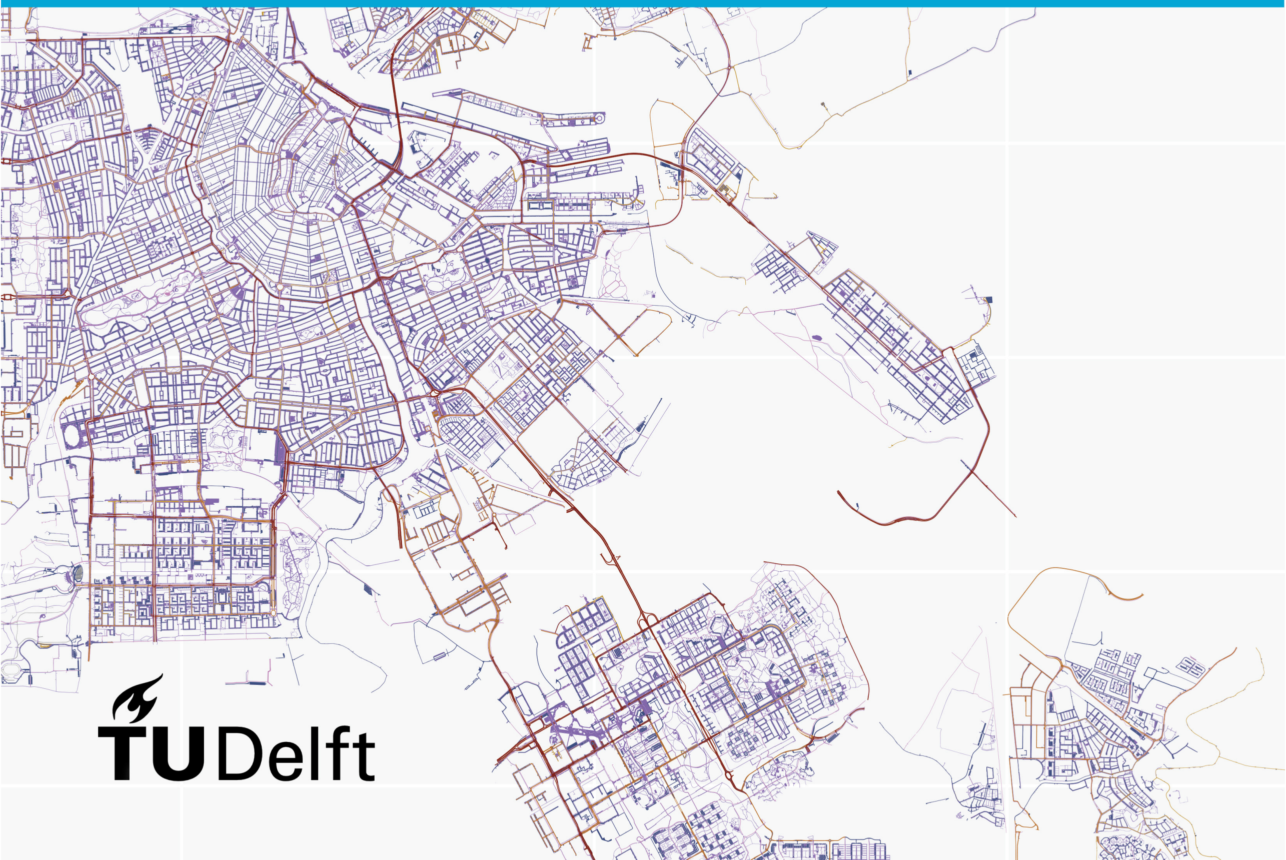
PhD Research Proposal

AI-Driven Decision-Making for Urban Infrastructure Life-Cycle Extension:
Towards real-world implementation

Amy Sterrenberg
2025



TU Delft

AI-Driven Decision-Making for Urban Infrastructure Life-Cycle Extension: Towards real-world implementation

PhD Proposal

Go/No-Go Date: 21/10/2025

PhD Candidate:

Amy Sterrenberg

Co-promotor:

Dr. Charalampos Andriotis

Promotor:

Prof. Dr. Jantien Stoter

Additional committee members:

Prof. Dr. Eleni Chatzi
Dr. Tong Wang

The work presented in this proposal was carried out at:

3D Geoinformation,
Department of Urbanism,
Faculty of Architecture and the Built Environment,
Delft University of Technology, the Netherlands.

The work presented in this proposal was carried out in collaboration with:

AiDAPT Lab,
Department of Architectural Engineering + Technology,
Faculty of Architecture and the Built Environment,
Delft University of Technology, the Netherlands.

Amsterdam Institute for Advanced Metropolitan Solutions,
Amsterdam, the Netherlands.

Funding for this work is provided by:

Nationaal Groeifonds,
Toekomstbestendige Leefomgeving: Infra.

Contents

List of Acronyms

AI Artificial Intelligence. Systems that can perform tasks typically requiring (human) intelligence, such as learning, reasoning, problem-solving, and decision-making. 2, 4, 15, 23, 24, 54, 55

GNN Graph Neural Network. A class of neural networks designed to operate on graph-structured data, where nodes, edges, and their features are used to learn representations that capture the relational and topological structure of the graph. 24, 26

HCRL Hierarchical Coordinated Reinforcement Learning. A reinforcement learning framework for multi-agent systems that combines hierarchical policy structures with coordination mechanisms between agents. 24–26

HRL Hierarchical Reinforcement Learning. A reinforcement learning approach that decomposes complex tasks into multiple levels of abstraction, where higher-level agents select goals, guiding lower-level agents in selecting actions or sub-policies. 24–26

IL Imitation Learning. A learning approach in which an agent learns to perform tasks by replicating expert demonstrations instead of using explicit reward signals. 24, 26, 35

IRL Inverse Reinforcement Learning. A reinforcement learning framework that infers the reward function underlying observed expert behaviour, rather than learning a policy directly from explicit rewards. 24–26, 43

MARL Multi-Agent Reinforcement Learning. A branch of reinforcement learning involving multiple agents that learn and interact within a shared environment, often requiring coordination, competition, or communication strategies. 20, 21, 25, 26

MDP Markov Decision Process. A stochastic decision-making process that uses a mathematical framework to model decision-making in dynamic systems. 13–16, 18, 20, 26

PdM Predictive Maintenance. Maintenance strategies that use data analysis, monitoring, and predictive modelling to estimate when an asset will require intervention, enabling maintenance to be performed before failure or significant deterioration occurs.. 1, 2, 22, 41, 42

POMDP Partially Observable Markov Decision Process. A generalisation of a Markov decision process in which the true state of the system is not fully observable, and decision-making relies on a belief state updated using probabilistic observations. 14, 15

RL Reinforcement Learning. A type of machine learning in which an agent learns to make decisions by interacting with an environment and receiving feedback in the form of rewards or penalties. 13, 15–21, 23–26, 31, 34, 44

1

INTRODUCTION

This introductory chapter presents a general introduction to the research, including the problem statement, a personal research motivation and an outline of the report structure.

1.1. PROBLEM STATEMENT

Timely maintenance of paved infrastructure is essential to ensure the serviceability and safety of urban transportation systems, which are essential to economic and social activity. However, cities today face growing challenges: vast and ageing infrastructure networks must be maintained under limited budgets, increasing traffic demand, and heightened sustainability pressures. These factors make maintenance planning increasingly complex and critical.

Current maintenance strategies are largely reactive; interventions are scheduled only after inspectors have observed damage. Such approaches limit long-term planning, increase repair costs, exacerbate traffic disruptions, and elevate safety risks, ultimately shortening the effective service life of paved infrastructure. As a result, reactive maintenance strategies may no longer suffice in the future. In contrast, Predictive Maintenance (PdM) offers a more proactive alternative, using condition monitoring and data analysis to anticipate degradation and optimise the timing of interventions. PdM enables not only more efficient asset-level maintenance but also strategic, long-term planning across the entire network.

Despite these advantages, the practical implementation of PdM remains limited. Two main challenges persist. First, PdM requires accurate prediction of infrastructure condition over time, for all infrastructure components. However, existing pavement degradation models are incomplete. They often fail to capture the diversity of urban networks, where differences in age, materials, and traffic load affact degradation trajectories. For example, tiled pavements, although common in Dutch and European cities, are rarely represented. Historical condition data are also limited, which further complicates degradation modelling.

The second limiting factor is the complexity of infrastructure maintenance planning. This complexity has three main sources. First, infrastructure systems are high-dimensional: they consist of thousands of interdependent road segments, each affecting the others. This results in a vast number of possible maintenance actions to choose from. Second, decisions must be made under uncertainty, as degradation processes are inherently stochastic. Third, maintenance planning is multi-objective: it must balance economic, societal, and environmental impacts (for example, minimising costs while reducing disruption and lowering emissions).

Artificial Intelligence (AI) techniques, particularly Deep Reinforcement Learning (DRL), show promise in addressing these challenges. DRL frames maintenance planning as a sequential decision-making problem: an agent takes actions, receives feedback, and learns to optimise intervention policies over long planning horizons. Unlike traditional optimisation or rule-based approaches, which rely on fixed degradation models and short-horizon decision rules, DRL can adapt its policies dynamically by learning from observed states and transition dynamics, thereby balancing degradation risks, intervention costs, and network performance more effectively.

Research Gaps

Still, current DRL approaches do not yet enable practical implementation of PdM. Existing models can optimise maintenance strategies, but only on a limited scale: for networks with up to a few hundred components. Scaling these methods to real urban networks, which contain thousands to tens of thousands of interdependent road segments, remains a major challenge. Most studies also rely on synthetic or simplified datasets, and few have been tested on real infrastructure networks. This, combined with incomplete degradation models that often fail to capture variations in surface type, traffic load, and other relevant characteristics, limits the ability of DRL methods to make reliable predictions for practical maintenance planning.

Another limitation is the focus of existing work on single-objective optimisation, typically minimising economic costs. Real-world maintenance planning, however, must balance multiple objectives, including societal and environmental impacts. Finally, DRL methods are not inherently interpretable: their decision-making processes are difficult to understand or justify to maintenance planners. This lack of transparency can limit trust and hinder adoption. Collectively, these limitations reveal three critical gaps that must be addressed for DRL to support practical PdM: (1) scalability, (2) interpretability and (3) real-scale validation.

Research Question

These gaps motivate the development of new methods capable of handling the complexity of urban-scale, multi-objective maintenance planning while providing interpretable decision-making and outputs. This has led to the following research question:

> ***"How can a data- and AI-driven decision-support framework be developed for multi-year, urban-scale inspection and maintenance planning that (1) scales to thousands of infrastructure components, (2) produces interpretable intervention policies for practitioners, and (3) can be validated on real-world heterogeneous infrastructure networks?"***

The goal of this research is to develop such a framework, producing practical PdM policies for urban infrastructure networks that maintenance planners can both interpret and trust. Achieving this requires three main steps. First, a predictive pavement degradation model will be developed. Second, a novel DRL architecture for maintenance decision-making will be designed, with a focus on decomposing and hierarchising the problem to manage complexity. Third, the framework will be validated on urban-scale networks to ensure both practical applicability and reliability. The research gaps, objectives, methods, and work plan are presented in more detail in the remainder of this proposal.

1.2. Personal Motivation

Maintenance planning is a high-dimensional, multi-objective decision-making problem in a stochastic setting. Problems with such a multitude of factors and uncertainties often appear intractable, even when approached with advanced algorithms. The formalisation of these problems, and the development of methods to address them, has long been a specific area of interest to me.

Since I am frequently asked how a background in architecture and design informatics led me to this research, I wish to briefly outline the strong parallels in the way these field address complex decision-making. Design processes can themselves be viewed as complex decision-making processes, involving a near-infinite set of possible solutions. While many options can be dismissed as suboptimal based on fundamental design principles, the remaining design space often remains too vast to manage, limiting the ability to identify a set of optimal solutions. A similar challenge arises in maintenance planning, with the added complexity of operating in a stochastic and uncertain environment.

It is precisely this challenge that motivates me: the pursuit of optimal solutions and, through them, a contribution to societal, environmental, and economic sustainability. This research represents a compelling example of such a challenge, as it combines the great societal value of effective maintenance planning with the high complexity of the underlying problem.

1.3. Report Structure

The remainder of this proposal is organised as follows. Chapter 2 includes the theoretical background and literature review of the PhD project. It ends by identifying three research gaps, which motivate the research question and objectives. These are presented in Chapter 3, along with the methods and workplan. This chapter concludes by outlining the wider impact and relevance of the research. Preliminary results are presented in Chapter 4. Finally, planning and practical aspects are addressed in Chapter 5, including a timetable, first year report, and information on the technical requirements for the PhD.

The following documents are included in the appendix:

A: PhD Agreement
B: Data Management Plan (DMP)
C: Overview of Graduate School Courses
D: ICOSSAR’25 Conference Paper

2

Literature Review

This chapter discusses foundational theoretical background and research done on the topic of inspection and maintenance strategy optimisation. In doing so, this chapter provides a comprehensive literature review for this PhD project.

2.1. Introduction

This chapter provides a theoretical and contextual foundation for the research. It reviews the key domains relevant to infrastructure life-cycle extension, and data-driven and AI modelling approaches required in this PhD. The objective is to situate the research within the broader landscape, highlight knowledge gaps, and establish the rationale for the proposed work. This chapter will follow the following structure:

- Section 2.2 provides definitions and presents the characteristics of transportation infrastructure systems to define the research scope.
- Section 2.3 introduces the foundational modelling framework for infrastructure systems and maintenance planning, including Markov Chains, Markov Decision Processes and (deep) reinforcement learning.
- Section 2.4 describes the current literature and introduces the gaps in detail.
- Section 2.5 introduces the emerging technologies that may be used to address these research gaps.

2.2. Transportation Infrastructure Systems: Definitions

This section outlines the characteristics of transportation infrastructure systems. It introduces and contextualises key terms and modelling concepts used in the PhD research. These concepts form the foundation upon which the research is built, providing both the basic technical language and conceptual framework necessary for understanding the scope, data, models, and algorithms employed in the remainder of this work.

First, *transportation infrastructure* systems consist of interconnected physical and functional elements that support the movement of people and goods. Within this research, the primary elements to be researched are *pavements*; the surface layers of transportation infrastructure elements. These pavements are analysed at the *road segment*-level, which, following the City of Amsterdam's convention, are defined as uninterrupted road stretches no longer than 100 metres. When studied collectively, these segments form an *infrastructure network*, a system of which the behaviour and dependencies are critical to understanding network-wide effects of maintenance on road segment-level.

Degradation describes the progressive decline in physical condition due to usage, weathering, and ageing. This is often called natural degradation. Degradation may also be caused by so-called rare events, such as accidents. To track the degradation process, *inspections* are performed. In inspection records, the condition of infrastructure elements is often expressed in discrete *condition states*. Within this framework, a failed state may be defined; a condition state which results in the loss of functionality of the component or system. Maintenance interventions directly influence degradation trajectories to keep the overall condition at a minimum level and help prevent failure, eventually extending the operational lifespan of infrastructure components, and thereby its *life-cycle*. The life-cycle of an infrastructure object is defined as all processes in-between and including the design and construction processes, operation and maintenance, as well as eventual decommissioning. Extending the operational phase of the infrastructure life-cycle contributes to reducing cost and emissions related to the pre- and post-operational stages of the infrastructure life-cycle.

Maintenance strategies or policies are multi-year plans including all inspection and maintenance actions to be performed. The decision-making process leading to these policies are a core aspect of *asset management*: a structured approach to managing infrastructure by scheduling inspection and maintenance actions to maximise performance and minimise costs and risk. A critical part of this process is defining and tracking *performance indicators*, which quantify performance measures (such as safety, condition, and cost) and support evidence-based decision-making. These performance indicators are elaborated on in the next subsection. Some of these performance indicators are influenced by the *interdependencies* between infrastructure elements; for instance, a maintenance intervention on one road segment might require a road closure, increasing traffic load on alternate routes, in turn accelerating their degradation.

PERFORMANCE INDICATORS

Designing inspection and maintenance strategies involves balancing multiple, often conflicting criteria such as cost, safety, and user disruption. To support this, a wide range of performance indicators can be identified. As shown in Figure 2.1, these indicators cover a broad spectrum of factors across systems and stakeholder perspectives. They align with the three pillars of sustainability: (1) societal systems, (2) economic systems, and (3) environmental systems [1]. Finally, they include both agency-level and user-level impacts.

The indicators span the full infrastructure life cycle and address not only direct impacts such as costs and emissions, but also broader effects like accessibility and land use. They can be organised around three core phenomena: (1) infrastructure condition, (2) inspection and maintenance actions, and (3) traffic disruptions. Research has been done on relevant models for calculating these indicators and outline their integration into a multi-criteria decision-making framework. A brief summary of this is presented in Section 4.4.

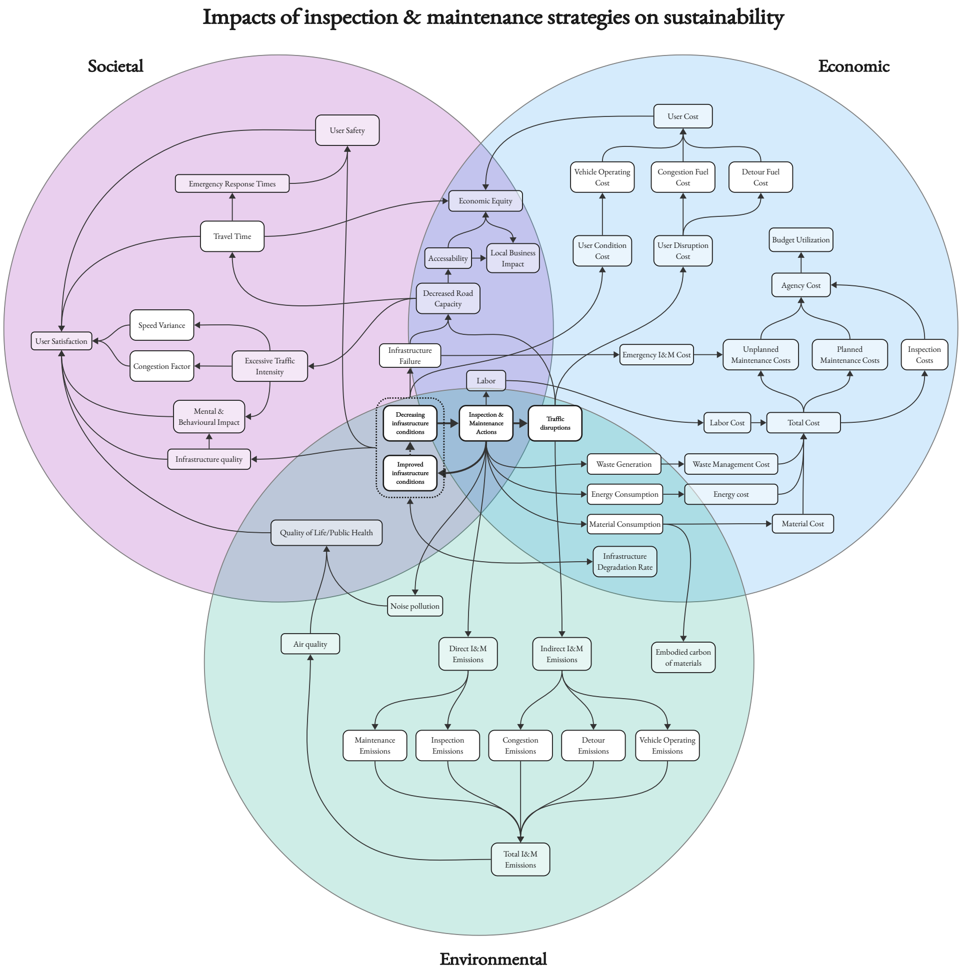


Figure 2.1: An overview of performance indicators to be considered in the I&M decision-making process for urban road networks. Arrows indicate causal relationships between indicators.

DEGRADATION PROCESSES

Road surfaces deteriorate due to traffic loading, environmental exposure (e.g., temperature, moisture), and decline of material properties. Two types of degradation affect infrastructure: (1) gradual degradation and (2) shock degradation [2, 3, 4]. Gradual degradation is a stochastic process that involves the continuous effects of fatigue and corrosion, whereas shock degradation includes the momentary or instantaneous effects of rare events such as earthquakes and floods. The effects of both processes on the condition of infrastructure objects over time is shown in Figure 2.2.

Degradation in pavements is multifaceted, emerging from the combined effects of several mechanisms, which are highly dependent on pavement construction and surface material (asphalt, concrete, tilled pavements, etc.). Typical degradation mecha-

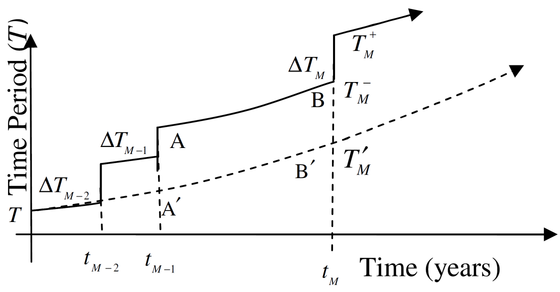


Figure 2.2: The effects of gradual degradation and shock degradation processes on structural condition over time, represented by the fundamental period T (which increases as stiffness decreases). Curve $A'B'$ represents gradual degradation only. Curve AB shows the combined effects of gradual and shock degradation. Derived from [2].

nisms include cracking, rutting, potholing, material fatigue, surface deformation, unevenness and edge damage [5, 6]. Most existing studies predominantly focus on asphalt pavements, with no relevant research identified concerning concrete or tiled pavements. Given this emphasis, the remainder of this section will also focus primarily on asphalt pavements. Nonetheless, the CROW guidelines additionally recognise joint width and joint filling between paving elements as additional degradation mechanisms for tiled and concrete pavements.

Cracking

Cracking encompasses several forms, including fatigue (or alligator), longitudinal, transverse, block, slippage, reflective, and edge cracking [6]. These arise from traffic loading, thermal and environmental stresses, construction deficiencies, moisture intrusion, inadequate support or aging. Cracks may initiate from the bottom (bottom-up) due to tensile and shear stresses within structural layers, or from the surface (top-down) due to surface tensile stresses, binder ageing, or reduced stiffness at high temperatures [5].

Surface deformations

Surface deformations include rutting, corrugations, shoving, depressions, and swell [6]. Rutting, the most common form, develops in wheel paths due to repeated loading, material densification, or shear displacement [5, 7]. Corrugations and shoving typically occur under braking or acceleration, while depressions indicate localised instability of underlying layers [6, 8]. Swell generally results from frost heave or moisture expansion in subgrade soils [6].

Other mechanisms

Chemical ageing of bitumen makes the material more brittle, while moisture infiltration weakens the bond between binder and aggregates [9]. In combination with traffic and thermal stresses, these processes accelerate material loss and pothole formation, with significant implications for road safety and serviceability [5].

2.3. BACKGROUND: FOUNDATIONAL MODELLING FRAMEWORKS

This section introduces the core modelling frameworks for data-driven decision-making. It provides an overview of key concepts: graph-based models, Markov chains, Markov decision processes (MDPs), and their extensions into reinforcement learning (RL) and multi-agent reinforcement learning (MARL). Understanding these foundations is essential for identifying research gaps and motivating the research directions developed in later sections.

2.3.1. INFRASTRUCTURE NETWORK MODELLING: GRAPHS

Road segments and their interconnections may be represented as graphs. A graph is defined as a tuple (V, E), where V is the set of vertices (or nodes) and E is the set of edges [10]. Each edge connects a pair of vertices, allowing graphs to model both objects and their relationships. $v_i \in V$ can be used to denote a node and $e_{ij} = (v_i, v_j) \in E$ for an edge between node v_i and v_j. An adjacency matrix A can be used to describe a graph. It is a binary $n \times n$ matrix (where n is the number of vertices) that indicates whether two nodes in a graph are connected by an edge. In other words, $A_{ij} = 1$ if $e_{ij} \in E$, and $A_{ij} = 0$ if $e_{ij} \notin E$ [11]. Edges in a graph can be directed or undirected. Directed edges start from a sender node and end in a receiver node [12]. When all edges in a graph are undirected, the adjacency matrix is symmetric over its diagonal.

Graph attributes

Graphs, nodes and edges may be assigned attributes. Node attributes can be stored in a node feature matrix $\mathbb{R}^{n \times d}$ with $\mathbf{x}_v \in \mathbb{R}^d$ denoting the feature vector for any single node. Graph edges can also be assigned attributes. Similarly, an edge feature matrix $\mathbb{R}^{m \times c}$ contains edge feature vectors $\mathbf{x}_{v,u} \in \mathbb{R}^c$ for each edge. An example of a graph with its attribute representations is shown in Figure 2.3.

Types of Graphs

Graphs can be designed as spatial-temporal graphs. In such graphs, node attributes that can change over time. Spatial–temporal graphs can be defined as $G(t) = (V, E, X(t))$, with $X(t) \in \mathbb{R}^{n \times d}$ [11]. Finally, in multi-graphs, multiple edges can be made between vertices, including edges between a single node [12].

Graphs for transportation networks

In the context of transportation infrastructure, vertices can represent road segments, while edges represent the connections between them. Attributes can be assigned to graphs, nodes and edges. Nodes, representing road segments, may, for example, be assigned an ID, condition label, material, last maintenance date and traffic intensity and capacity. Edges, representing connections between road segments, may be assigned directionality (one-way or bidirectional) and type (e.g., intersection, etc.).

Graphs can be efficiently created and analysed in Python using the NetworkX library[1]. This library provides tools for constructing graphs and assigning attributes to graphs, nodes, and edges. Additionally, it includes graph-based algorithms, such as

[1] https://networkx.org/

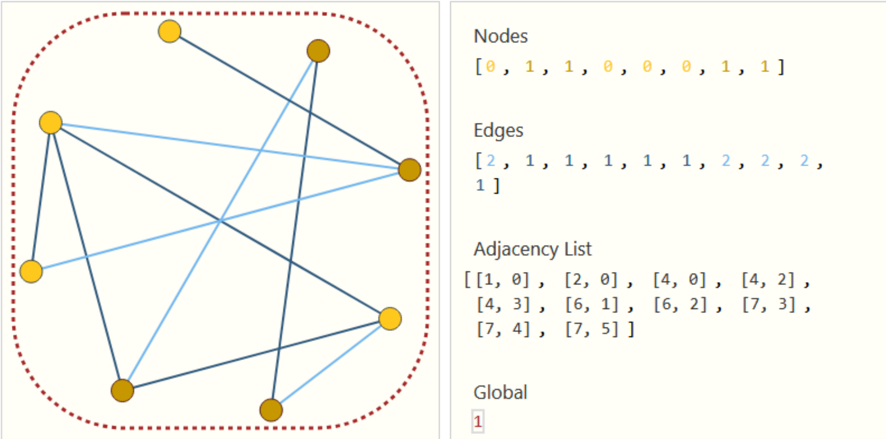


Figure 2.3: A graph and its attributes, derived from [13]

shortest path calculations and flow hierarchy analysis, which can be used in traffic modelling.

Traffic Modelling

Traffic models can provide information on the flow of traffic in an infrastructure network. For the applications in this PhD research, the modelling of individual vehicles is not required; information of predicted traffic load per road segment, average speed and speed variance are sufficient. The model to be developed for this PhD research model should also be able to simulate the effects of disruptions under various maintenance scenarios, as their influence on travel time and congestion are essential for informing decision-making. Therefore, microscopic models are not considered here [14]. Instead, the focus is on macroscoptic factors, such as traffic flow (number of vehicles N over time t), average speed, travel time, road occupancy (time a road is occupied) and density (vehicles per road segment) [15].

Traffic models can be further classified as stationary or temporal. In stationary models, properties such as traffic load do not change over time, for example throughout the day. In temporal models, the change of traffic due to time of day may be considered [16]. For the purpose of this PhD research, stationary models suffice, as maintenance actions typically take days to months to complete.

2.3.2. Degradation Modelling: Markov Chains

To predict the future condition of infrastructure components, a pavement degradation model is required. Such models can be deterministic or stochastic; deterministic models always produce the same output for a given set of inputs, whereas stochastic models explicitly incorporate randomness and variability. Stochastic approaches are particularly valuable when input parameters, environmental influences, or usage conditions are uncertain or highly variable. As these conditions that are common in infrastructure degradation, stochastic approaches are typically preferred for this context. In such models, degradation can be represented in continuous or discrete time, and with continuous or discrete states. The choice of variable to model also differs across approaches; many approaches directly represent the physical condition of components, while others reframe degradation: hazard models focus on time-to-failure, and reliability-based models describe the probability of failure. In recent years, much attention has been given to Markov Chains, which offer a flexible stochastic framework for capturing degradation dynamics.

Markov Chains, also called Markov Models or Markov Processes, are widely used to model the evolution of infrastructure condition as a sequence of probabilistic transitions between discrete condition states [17, 18]. This includes discrete-time Markov chains (DTMCs), continuous-time Markov chains (CTMCs), semi-Markov models (SMMs), and hidden Markov models (HMMs). Markov chain models represent transitions between a finite set of condition states in discrete or continuous time, which evolve over time depending on discrete actions. A Markov process is defined as $\{X(t), t \geq 0\}$ on a finite state space $S = \{1, 2, \ldots, n\}$. It can be defined as a tuple: $\langle S, P, \pi \rangle$, where:

- $S = \{1, 2, \ldots, n\}$ is the finite set of states;
- $P = [P_{ij}]$ is the transition probability matrix;
- $\pi = (\pi_1, \ldots, \pi_n)$ is the initial state distribution.

In a Markov Chain, the probability of transitioning to a future state adheres to the Markov property, meaning that the conditional distribution of a future state is dependent only on the present state [17]. Figure 2.4 shows a probabilistic graph of a Markov Chain. Based on this assumption, transitions from state i to any other state j can be defined by a probability P_{ij}. The model assumes stationary transition rates, meaning they remain constant over time. The probability of transitioning from state $i \in S$ to state $j \in S$ in exactly time Δt from an initial time t is denoted as [17]:

$$P_{ij}(t) = P(X(t + \Delta t) = j | X(\Delta t) = i) \tag{2.1}$$

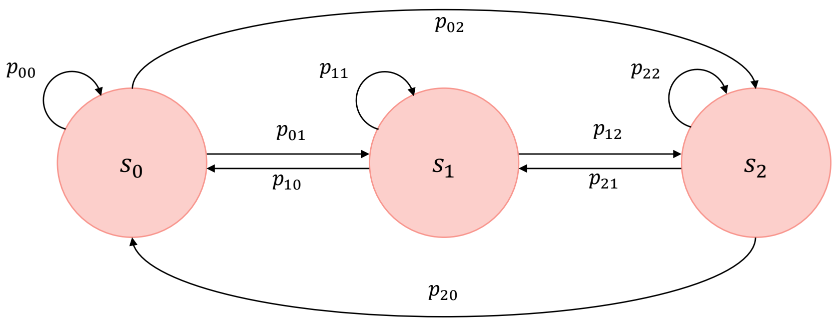


Figure 2.4: Probabilistic graph of a (discrete time) Markov chain model.

When these values for all origin and end state pairs are combined into a matrix, this is called the transition probability matrix $P(t)$ at time t:

$$P = \begin{bmatrix} P_{11}(t) & P_{12}(t) & \cdots & P_{1n}(t) \\ P_{21}(t) & P_{22}(t) & \cdots & P_{2n}(t) \\ \vdots & \vdots & \ddots & \vdots \\ P_{n1}(t) & P_{n2}(t) & \cdots & P_{nn}(t) \end{bmatrix} \tag{2.2}$$

Each row of P sums up to 1, and all entries satisfy $0 \leq P_{ij}(t) \leq 1$. The transition probability matrix for timestep t is used in discrete-time Markov processes, where $P_{ij}(t)$ can be estimated empirically by the observed transition frequencies [19, 20, 21]. Such matrices can be found using data when modelling a process as a Discrete-Time Markov Chain (DTMC). This means that the process is observed at discrete time points. In other words, the timestep between observations is constant.

Continuous-Time Markov Chains (CTMCs) on the other hand, can be used if transitions between states occur in continuous time. In a CTMC, the sojourn time (i.e. the amount of time the process remains in a given state i) follows an exponential distribution [17]. In this case, transition probability matrices can be derived from transition rate matrices. The transition rate matrix Q takes the form:

$$Q = \begin{bmatrix} -\lambda_1 & \lambda_{12} & \cdots & \lambda_{1n} \\ \lambda_{21} & -\lambda_2 & \cdots & \lambda_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{n1} & \lambda_{n2} & \cdots & -\lambda_n \end{bmatrix} \tag{2.3}$$

Following Eq 2.1, each off-diagonal entry λ_{ij} for $i \neq j$ is defined as the rate of transition from state i to state j [22, 23]:

$$\lambda_{ij} = \lim_{\Delta t \to 0^+} \frac{P(X(\Delta t) = j \mid X(0) = i)}{\Delta t} \tag{2.4}$$

The diagonal elements λ_i in the transition rate matrix Q reflect the total rate at which the segment leaves state i, and are defined such that each row of Q sums to zero.

Under the exponential distribution assumptions, the transition rates λ_{ij} are estimated from the observed transitions in the inspection dataset. The estimate for each λ_{ij} is given by [22, 24]:

$$\lambda_{ij} = \frac{n_{ij}}{\sum_{k=1}^{n_i} T_i^{(k)}} \tag{2.5}$$

where n_{ij} is the number of transitions from state i to state j over the observation period and $T_i^{(k)}$ is the observed sojourn time in state i before transitioning, for the k-th transition. To obtain the corresponding transition probability matrix P for a desired time horizon t, the matrix exponential can be computed [17]:

$$P = e^{Qt} \tag{2.6}$$

These discrete-time transition probability matrices, can subsequently be used in advanced decision-making algorithmic frameworks for predictive maintenance, such as a

Markov decision process or within (deep) reinforcement learning, which are discussed in section 2.3.3.

Semi-Markov Model (SMM)

A Semi-Markov Model (SMM) generalises the standard CTMC by allowing the time spent in each state, called the sojourn time, to follow an arbitrary probability distribution [25]. In a standard Markov model, transitions occur at fixed time steps (discrete-time) or according to exponential waiting times (continuous-time). SMMs relax this assumption while preserving the Markov property for the sequence of visited states. Therefore, a CTMC describes a specific case of SMM, where sojourn times are exponentially distributed.

Formally, a SMM can be constructed from a Markov Renewal Process (MRP). An MRP is defined by a renewal kernel (also known as a semi-Markov kernel) matrix $F(t) = [F_{ij}(t)]$ and an initial distribution π, where [26, 27]

$$F_{ij}(t) = \mathbb{P}(T_i \leq t, X_{n+1} = j \mid X_n = i), \quad i, j \in S, \ t \geq 0, \tag{2.7}$$

with T_i denoting the sojourn time in state i. The SMM is then obtained by embedding the MRP in continuous time, so that the process occupies state i for a random sojourn time governed by $F_{ij}(t)$ before moving to state j. The initial distribution $\pi = (\pi_1, \ldots, \pi_n)$ specifies the probability of starting in each state.

The tuple for such a model can be written as: $\langle S, P, G, \pi \rangle$, where S is the finite set of states, $P = [p_{ij}]$ is the embedded transition probability matrix of the underlying Markov chain, independent of time, $G = \{G_i(t)\}$ is the set of sojourn-time distributions, and $\pi = (\pi_1, \ldots, \pi_n)$ denotes the initial state distribution.

The connection between the transition probabilities and the sojourn time distributions is captured by the renewal kernel:

$$F_{ij}(t) = p_{ij} \, G_i(t),$$

which gives the joint probability that, given the process is currently in state i, the next state will be j and the transition will occur within time t.

Thus, the kernel $F_{ij}(t)$ combines both the embedded Markov chain structure through p_{ij}, and the sojourn-time distribution through $G_i(t)$. This generalises the standard Markov chain assumption of exponentially distributed (or fixed) holding times, thereby allowing for more realistic modelling of systems where waiting times are not memoryless.

Comparison of notation across models

To clarify the difference between DTMCs, CTMCs and SMMs, the key characteristics of each process is summerised in Table 2.1.

DTMCs, CTMCs and SMMs are widely used to model system that transition probabilistically between a finite set of states. However, these models assume that the current state of the system is fully observable at each time point. In many practical applications, however, the system's true state cannot be directly observed. Instead, only indirect or noisy observations are available. In such cases, Hidden Markov Models (HMMs) can be used. HMMs extend the Markov model framework to account for unobservable (hidden)

Table 2.1: Comparison of DTMC, CTMC, and SMM.

Model	Transition	Definition	Timing	Factorisation
DTMC	$P = [p_{ij}]$	$p_{ij} = \mathbb{P}(X_{n+1} = j \mid X_n = i)$	Step	–
CTMC	$P(t) = e^{Qt}$	$P_{ij}(t) = \mathbb{P}(X(t) = j \mid X(0) = i)$	Exp	–
SMM	$F(t) = [F_{ij}(t)]$	$F_{ij}(t) = \mathbb{P}(T_i \le t, X_{n+1} = j \mid X_n = i)$	$G_i(t)$	$F_{ij}(t) = p_{ij}\, G_i(t)$

states, using probabilistic observations.

Hidden Markov Model (HMM)

For a Markov Chain $\{X_t,\, t \ge 0\}$ on a finite state space $S = \{1, 2, \ldots, n\}$, a HMM is a doubly stochastic process defined by the tuple $\langle S, \mathcal{O}, P, B, \pi\rangle$[28, 29]. Here, S is the finite set of hidden states. The process produces observations from an observation space $\mathcal{O}$, which can either be a finite set of symbols $\{o_1, \ldots, o_m\}$ or a continuous space. The hidden states evolve according to a transition probability matrix $P = [P_{ij}(t)]$, which describes the probabilities of moving between hidden states over time. More precisely, each entry $P_{ij}(t)$ gives the probability that the process is in state j at time $s + t$, given that it was in state i at time s:

$$P_{ij}(t) = \mathbb{P}(X(s+t) = j \mid X(s) = i), \quad i, j \in S,\ t \ge 0. \tag{2.8}$$

At each time step, the model emits an observation according to a set of emission probabilities or densities $B = \{b_j(\cdot)\}$, such that

$$b_j(o) = \mathbb{P}(Y_t = o \mid X_t = j), \quad o \in \mathcal{O},\ j \in S. \tag{2.9}$$

Finally, the initial state distribution is given by $\pi = (\pi_1, \ldots, \pi_n)$, where

$$\pi_i = \mathbb{P}(X_0 = i), \quad i \in S. \tag{2.10}$$

2.3.3. MAINTENANCE DECISION-MAKING: MARKOV DECISION PROCESSES

Maintenance planning and decision-making are the main topic of research. In this section, the foundational mathematical framework used for this purpose is introduced: a Markov Decision Process.

Markov Decision Processes (MDPs) address sequential decision-making by extending Markov Chains. Formally, an MDP can be described as a tuple $\langle \mathcal{S}, \mathcal{A}, P, R\rangle$, where $\mathcal{S}$ is the set of states, $\mathcal{A}$ the set of actions, $P(s' \mid s, a)$ the transition probability function, and $R(s, a)$ the reward function. Often, a discount factor γ is added to this tuple. Some literature also mentions adding a factor H, representing the time horizon of the problem [30].

As MDPs are based on the concept of Markov Chains, they also assume the Markov property: the probability of reaching the next state depends only on the current state and action, not on the history of past states or actions. This property enables tractable modeling and learning of sequential decision problems. The MDP structure also provides the mathematical foundation for Reinforcement Learning (RL), where agents learn optimal policies that maximize expected returns. This is elaborated on in section 2.3.4.

APPLICATIONS IN MAINTENANCE PLANNING

In maintenance planning optimisation, MDPs have been widely applied at both component and network levels.

Component-Level Applications

At the component level, MDPs can be used to optimise maintenance decisions for individual facilities or infrastructure elements. The models typically incorporate stochastic degradation or reliability-based deterioration models, enabling the evaluation of different maintenance strategies and their expected long-term costs [31, 32].

Network-Level Applications

For network-level planning, MDPs can be applied in two main approaches:

- **Top-down approach:** Network components with similar structural, traffic, and environmental characteristics are grouped. This allows decision-makers to balance the trade-off between rehabilitating a small number of critical components versus maintaining a larger portion of the network [33].
- **Bottom-up approach:** Optimal or near-optimal maintenance sequences are first generated for individual facilities. These sequences are then combined to satisfy budget constraints while optimising one or more network-level objectives over a planning horizon [34]. These approaches typically result in increased computational complexity.

Dependency-Based Models

More recent studies have extended network-level MDPs to capture functional and economic interdependencies between components to further optimise maintenance strategies. For example, such strategies may consider simultaneous scheduling of maintenance actions on adjacent road segments to reduce traffic delays over the planning horizion. These dependency-based models introduce nonlinear constraints, resulting in larger state-action spaces and higher computational complexity [33].

EXTENSIONS OF THE MDP FRAMEWORK

MDPs are commonly used in maintenance planning strategy generation for transportation infrastructure systems (e.g., [35, 36, 37, 38, 39]). As such problems do not strictly follow fully observable, discrete-time MDP problems, extensions of the MDP framework are typically preferred in more recent literature. Examples include Semi-Markov Decision Processes (SMPs) for life-cycle optimisation (e.g.,[40, 32]), Partially Observable Markov Decision Processes (POMDPs) for integrating uncertain data (e.g.,[31, 41, 42, 43]), integrated frameworks combining network-level optimisation with component-level MDPs [44] and frameworks that combine MDPs with the concept of real option analyses (ROAs) to address cost uncertainties [45].

Semi-Markov Decision Processes (SMPs)

Semi-Markov Decision Processes (SMPs) extend MDPs by allowing the sojourn time (time spent in a state before transition) to follow a general probability distribution conditioned on the state–action pair, rather than assuming memoryless (exponentially distributed) transitions [25]. Formally, a SMP can be described as a tuple: $\langle \mathscr{S}, \mathscr{A}, P, R, F \rangle$, where $\mathscr{S}$, $\mathscr{A}$, P, and R are as in a standard MDP, and F, or $F(\tau \mid s, a))$, a distribution of holding times τ (or a waiting time matrix) [25, 46].

2

Partially Observable MDPs (POMPDs)
Partially Observable Markov Decision Processes (POMDPs) generalise MDPs to situations where the true system state is not fully observable. Observations provide indirect, noisy, or incomplete information. A POMDP is defined as the tuple $\langle \mathscr{S}, \mathscr{A}, P, R, \Omega, O \rangle$, where $\mathscr{S}$, $\mathscr{A}$, P, and R are again as in a standard MDP, Ω is the set of possible observations, and $O(o \mid s', a)$ is the observation probability function. POMDPs are solved by maintaining and updating a belief state (probability distribution over possible states) and optimising actions with respect to this belief. Beliefs about the system are updated using Bayes' rule after taking an action and receiving an observation. In maintenance planning, this allows integration of inspection data into decision-making when true degradation states are only partially observable [31]. In some systems, the agent may not be able to fully observe the state, but only some components of the state, resulting in mixed observability. For these problems, mixed observability MDPs have been developed [47][2].

Multi-objective MDPs
The multi-objective MDP can be defined as a similar tuple to standard MDPs: $\langle \mathscr{S}, \mathscr{A}, P, R \rangle$, where $\mathscr{S}$ is the set of states, $\mathscr{A}$ the set of actions, $P(s'|s,a)$ the transition probabilities and R is the reward function. Often, a discount factor γ is also included in this tuple.

The difference between the standard MDP definition and multi-objective MDPs lies primarily in the reward function and, subsequently, in the optimization goals. In multi-objective MDPs, the reward function is defined as a vector of m objectives, instead of a scalar:$\mathbf{R}(s,a) \in \mathbb{R}^m$ [48]. As a result, the value function is also vector-valued, meaning that optimal policies are defined as set of Pareto-optimal policies Π^*. [48]

2.3.4. MAINTENANCE POLICY OPTIMIZATION: REINFORCEMENT LEARNING

Reinforcement Learning (RL) extends the MDP framework by using data-driven learning to approximate optimal policies. It belongs to the broader field of Artificial Intelligence (AI), which concerns the ability of machines to perform tasks that typically require human intelligence [49, 50]. Within AI, Machine Learning (ML) focuses on systems that improve performance through experience rather than explicit programming [49].

Three main learning paradigms exist: supervised, unsupervised, and reinforcement learning [51]. Unlike supervised methods, which learn from labelled datasets, or unsupervised methods, which uncover patterns in unlabelled data, Reinforcement Learning (RL) learns through interaction [52, 51, 53]. An agent observes the state of its environment, selects an action, and receives a reward that reflects the desirability of the outcome. Over time, the agent learns a mapping from states to actions that maximises cumulative reward, called a policy (see Figure 2.5) [53]. To achieve this, RL problems are defined as a Markov Decsion Process (MDP), as introduced in section 2.3.3 [54, 53].

[2] The abbreviation 'MOMDP' is used to refer to mixed-observability MDPs, but may also refer to multi-objective MDPs [48].

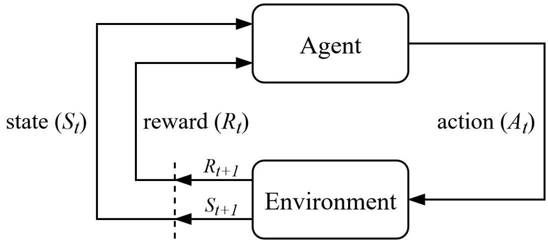


Figure 2.5: Schematic of a RL model, adapted from [53].

In infrastructure management, RL enables dynamic, policy-based decision-making under uncertainty. It can model interactions between maintenance actions, system conditions, and resulting impacts to learn strategies that optimise long-term performance. Deep RL (DRL) is an extension of RL that uses more complex Deep Learning (DL) architectures [55].

CATEGORISATION OF REINFORCEMENT LEARNING APPROACHES

RL models can be divided into several categories. Here, three categorisations will be introduced: (1) On- and -off policy RL, (2) model-based and model-free RL and (3) single-agent and multi-agent RL.

On-policy and off-policy reinforcement learning

In on-policy methods, a RL agent learns the value of the policy it is currently following, denoted π. To balance exploration and exploitation, on-policy methods often use ϵ-greedy policies, where the action with the highest estimated value is chosen with probability $1-\epsilon$, and a random action is selected with probability ϵ [53]. An example of an on-policy method is SARSA, which will be described in more detail later in this section.

In off-policy methods, the agent follows a behavior policy μ while learning the value of a different target policy π, often the optimal policy π^*. This allows off-policy methods to learn from data generated by other policies, including expert demonstrations or conventional controllers. Off-policy methods can be less stable than on-policy methods, but they offer greater flexibility in using existing data [53].

Model-based and model-free reinforcement learning

Model-based RL is used when the environment's transition probabilities and reward function are known or can be accurately specified [56]. When this information is available, the MDP can be solved using relatively simple planning algorithms (e.g., value iteration, policy iteration) to compute the optimal policy, provided the state and action spaces are not prohibitively large.

In contrast, model-free RL does not require prior knowledge of the transition dynamics or reward function. Here, the term 'model' refers specifically to the transition probability distribution and reward function within the MDP framework [56]. Instead of an explicit, analytical model, a model-free agent learns directly from observed transitions in empirical data.

Single-agent and multi-agent reinforcement learning

While single-agent RL models have one decision-making agent operating in an environment, multi-agent RL model scenarios where two or more agents share an environment, each taking decision to maximise individual return functions [57]. Agents in these models can have cooperative (working towards a shared goal) or competitive (working against each other) relationships. Mixed settings also exist.

FINDING OPTIMAL POLICIES

Policies in RL are defined as mappings of probabilities between states and actions. Therefore, $\pi(a|s)$ is the probability that the agent will select action a when in state s. RL approaches can be divided further based on the methods used to find these optimal policies: [55]

1. Value-based methods
2. Policy-based methods
3. Actor-critic methods

Policies can be either deterministic or stochastic. With a deterministic policy, only one action is mapped to each state. Stochastic policies instead map a probability distribution of all actions to each state. As a result, a deterministic policy will always select one action for a given state, where the stochastic policy uses sampling to select and action for a given state. Value-based methods usually derive deterministic optimal policies (though they often rely on stochastic exploration during training). By contrast, policy-based and actor-critic methods can represent stochastic policies, making them more flexible, especially in continuous or partially observable environments [58]. The three different approaches are described in more detail below.

Value-based methods

Information in this section is based on [53], chapter 3.

Value-based reinforcement learning methods find optimal policies by estimating value functions, which quantify how good it is (1) to be in a given state or (2) to take a particular action in that state. These functions are defined in terms of return: the cumulative discounted reward that an agent expects to receive from a given time step onward.

The two central value functions are the state-value function $v_\pi(s)$, giving the expected return when following a policy π from state s, and the action-value function $q_\pi(s,a)$, giving the expected return when taking action a in state s and thereafter following π. Both can be expressed recursively through the Bellman equations, which relate the value of a state (or state–action pair) to its immediate reward and the value of subsequent states.

The Bellman optimality equation defines the unique optimal value function v^*, describing the maximum achievable return from each state, independent of any policy. If the environment dynamics p are known, this system can in principle be solved directly to obtain the optimal policy, which is greedy with respect to the value function, as it makes locally optimal choices that are globally optimal due to the value function encoding long-term consequences.

In practice, directly solving the Bellman equations is rarely feasible because it requires full knowledge of transition probabilities, high computational capacity, and strict adherence to the Markov property. Even when the Markov property holds and the dynamics are known, the computational burden is often large. This is mainly due to the curse of dimensionality, which will be elaborated on in Section 2.4.1.

To address this, practical RL methods use approximate solutions that rely on sampled experience rather than full transition models. This allows learning to occur online, with the agent improving its policy through interaction with the environment. This enables the model to quickly and effectively learn the best actions for frequently visited states, at the cost of actions in rarely visited states. This property is not found in other approximation methods for solving MDPs.

Policy-based methods

Information in this section is based on [53], Chapter 13, unless stated otherwise.

Policy-based methods directly learn a parameterised policy $\pi_\theta(a|s)$ that defines the probability of taking action a in state s. Instead of deriving a policy from value functions, these methods optimise the policy parameters θ to maximise an objective function $J(\theta)$, which represents the expected return when following π_θ [55].

The parameters are adjusted through *gradient ascent*, gradually increasing the expected return by moving in the direction of the performance gradient $\nabla_\theta J(\theta)$. The REINFORCE algorithm [59] provides a practical way to estimate this gradient from sampled trajectories, updating the policy based on observed rewards. Each update increases the likelihood of actions that led to higher returns. In this update, each parameter increment is proportional to the product of the return G_t and the vector $\nabla_\theta \pi_\theta(A_t \mid S_t)/\pi_\theta(A_t \mid S_t)$, which points in the direction that increases the probability of selecting action A_t in future visits to state S_t. Scaling by G_t emphasizes actions that lead to higher rewards, while dividing by $\pi_\theta(A_t \mid S_t)$ ensures that the update reflects how changes in the policy affect the probability of that action; intuitively, this also prevents frequently chosen actions from dominating solely due to their higher selection rate.

Because REINFORCE relies on complete episodes to compute returns, it is best suited for episodic tasks. Nevertheless, a key advantage of policy-based methods is that they can naturally represent stochastic policies, allowing exploration and modeling of uncertain environments directly. The policy gradient theorem provides a tractable expression for estimating $\nabla_\theta J(\theta)$ from sampled trajectories, forming the basis for REINFORCE and related algorithms.

Actor-Critic Methods

Information in this section is based on [53], Chapter 13, unless stated otherwise.

Actor-critic methods combine the strengths of value-based and policy-based reinforcement learning. The *actor* selects actions according to a parameterised policy $\pi_\theta(a|s)$, while the *critic* evaluates those actions using a value function estimate. This separation allows more stable and efficient learning than using either approach alone.

The critic provides feedback to the actor based on the temporal-difference (TD) error, which measures the difference between predicted and actual returns. The actor then adjusts its policy in the direction that improves performance, while the critic updates its value estimates using the same TD signal. This interaction enables both components to learn continuously and mutually reinforce each other, as visualised in Figure 2.6.

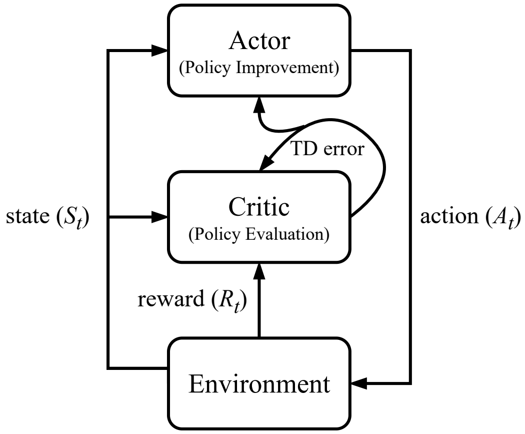


Figure 2.6: Schematic of an actor-critic algorithm, adapted from [60].

Actor-critic methods provide several advantages:

1. Reduced variance: Using a critic to estimate the value function reduces the variance of the policy gradient compared to policy-based methods like REINFORCE.
2. Online learning: TD-based updates allow learning from incomplete episodes, making actor-critic methods suitable for continuing tasks.
3. Stochastic policies: Like policy-based methods, the actor can naturally represent stochastic policies, which is important for exploration.

SINGLE-AGENT ARCHITECTURES

A wide range of RL architectures have been developed for single-agent settings. Some of the most common and foundational approaches are summarised here.

Early tabular methods, such as SARSA (State–Action–Reward–State–Action) [61] and Q-learning [62], update the action-value function after each step using the Bellman equation. SARSA is an iterative, on-policy algorithm that updates Q-values using the quintuple $(S_t, A_t, R_{t+1}, S_{t+1}, A_{t+1})$, while Q-learning is off-policy and directly learns the optimal value function, independent of the behaviour policy [53].

To handle high-dimensional state spaces, classical TD methods were extended via function approximation, most prominently through Deep Q-Networks (DQN), which parameterise the Q-function with a deep neural network [63]. In deep Q-learning, techniques such as experience replay and target networks are used to stabilise training [64]. The tendency of DQN to overestimate action values motivated the development of Double Deep Q-Networks (DDQN), which decouple action selection and evaluation by using the online network for selecting the next action and the target network for evaluating its value [65].

The Dueling (Deep) Q-Network architecture further refines value estimation by decomposing the Q-function into separate estimators for the state-value and state-dependent action advantage, improving learning in environments where many actions have similar value [66]. The Deep Reinforcement Relevance Network (DRRN) adapts the Q-learning framework by embedding both states and actions in separate vector spaces. This algorithm is particularly suited for environments with natural language action spaces [67].

Moving towards policy gradient and actor–critic frameworks, the previously introduced REINFORCE algorithm is considered foundational [59]. The Asynchronous Advantage Actor–Critic (A3C) algorithm extends this framework by employing multiple parallel agents that independently interact with the environment and asynchronously update a shared neural network using an advantage function [68].

The Deep Deterministic Policy Gradient (DDPG) algorithm introduces a deterministic policy update with an actor–critic structure suitable for continuous action domains [69]. Second-order trust-region methods such as Trust Region Policy Optimization (TRPO) constrain policy updates within a KL-divergence between old and new policies to achieve stable convergence [70]. Proximal Policy Optimization (PPO) is a more computationally efficient surrogate-based variant of this algorithm, introducing an objective function that allows for multiple epochs of minibatch updates [71]. Finally, the Soft Actor–Critic (SAC) algorithm extends the actor–critic framework with maximum entropy reinforcement learning, balancing reward maximisation with exploration by encouraging stochastic, high-entropy policies [72].

MULTI-AGENT REINFORCEMENT (DEEP) LEARNING (MA(D)RL)

Multi-Agent Reinforcement Learning (MARL) extends RL to environments with multiple interacting agents. Each agent learns a policy to maximise its own or a shared reward, often representing a system component in maintenance modelling. Together, the agents aim to maintain overall system performance. The framework builds on Markov Games, a generalisation of MDPs for multi-agent settings [73, 74]. Formally, a Markov Game for N agents is defined as a tuple [74]:

$$\langle \mathcal{S}, \{\mathcal{A}_i\}_{i=1}^N, P, \{R_i\}_{i=1}^N, \gamma \rangle$$

where:

- $\mathcal{S}$ is the set of environment states,
- $\mathcal{A}_i$ is the action set available to agent i, and $\mathcal{A} = \mathcal{A}_1 \times \cdots \times \mathcal{A}_N$ is the set of joint actions,
- $P(s'|s, a_1, \ldots, a_N)$ is the transition probability to state s' given s and joint action $(a_1, \ldots, a_N)$,
- $R_i(s, a_1, \ldots, a_N)$ is the reward function for agent i,
- $\gamma \in [0, 1)$ is the discount factor, often shared across agents.

In a Markov Game, each agent i aims to learn a policy that maximizes its expected return, which typically depends on the strategies of other agents in competitive, cooperative, or mixed settings. The Markov Game framework can be extended to partially observable settings (partially observable Markov Games, POMG) or, when agents share a reward function, to decentralized partially observable MDPs (Dec-POMDPs) [74].

Three training paradigms are typically followed in MARL models: (1) Centralized Training with Centralized Execution (CTCE), (2) Decentralized Training with Decentralized Execution (DTDE), and (3) Centralized Training with Decentralized Execution (CTDE) [74, 75]. In CTCE, both training and execution rely on global state and joint actions. In

DTDE, each agent trains and selects actions based solely on its local observations and rewards. In CTDE, training uses centralized information, but agents act based on local observations during execution. A summary of these paradigms is shown in Figure 2.7.

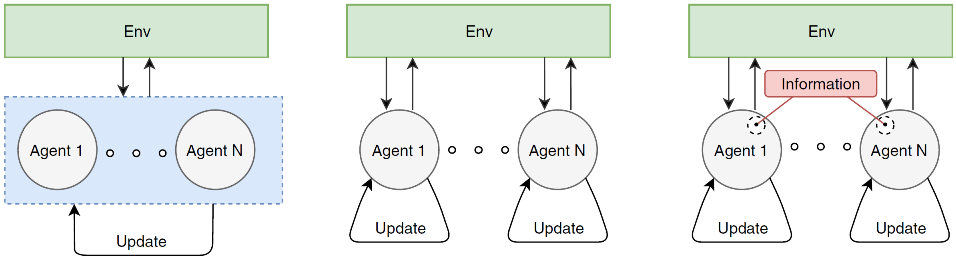


Figure 2.7: Schematic of training paradigms for MARL. In CTCE (left) a joint policy is applied to all agents. In contrast, DTDE (middle) applies individual policies to each agent. Finally, in CTDE (right), agents, while updating their own policy, exchange information. Derived from [74].

In maintenance decision-making, CTCE architectures such as Joint Actor–Critic (JAC), Deep Centralised Multi-Agent Actor–Critic (DCMAC), and Deep Decentralised Multi-Agent Actor–Critic (DDMAC) have shown high performance in coordinated tasks (e.g., a case study of a k-out-of-5 system) [75, 76]. DTDE approaches, such as Independent Actor–Critic (IAC) and its parameter-sharing variant (IAC-PS), offer better scalability but suffer from instability due to independent training. CTDE variants, including Independent Actor–Centralised Critic (IACC) and IACC-PS, achieve a balance by using a central critic during training while keeping policies decentralised at execution.

Despite their promise, MARL approaches remain challenged by high computational demands, convergence issues, and limited interpretability. These constraints currently limit their large-scale application in infrastructure maintenance systems [77, 75, 78].

2.4. CURRENT LIMITATIONS

In the previous section, existing frameworks for maintenance planning for urban infrastructure networks were presented. In this section, the limitations of these models are discussed. These limitations will be the foundation to the research gaps to be addressed in this PhD research. It should be noted that these gaps do overlap. In Section 2.5, emerging directions are presented that address one or more of these research gaps.

2.4.1. SCALABILITY: CURSE OF DIMENSIONALITY AND HISTORY

First, despite their promise, existing RL and MARL models for maintenance planning have limited scalability. This limitation is due to the *curse of dimensionality* and *curse of history*, which refer to the combinatorial growth of the complexity of maintenance policies with the number of infrastructure components, the number of available actions, and the number of decision epochs in the planning horizon.

Suppose there are N components, and for each component one out of A possible actions can be chosen (e.g., do nothing, minor maintenance, major maintenance). The total number of distinct action combinations at a single decision epoch is $|\mathscr{A}_{\text{joint}}| = A^N$.

This is the curse of dimensionality. Additionally, over a planning horizon of T epochs, a maintenance plan specifies one joint action at each epoch. The total number of distinct open-loop action plans is therefore $|\mathscr{A}_{\text{plans}}| = (A^N)^T$. This is the curse of history.

Due to this exponential growth, even with relatively modest values for an infrastructure network, the number of possible maintenance plans becomes very large. As a result, exhaustive enumeration of all possible plans becomes infeasible in practice. Current models are able to consider up to hundreds of components (e.g., [76, 75]). Some models are able to handle thousands of components. In such cases, however, these networks either do not consider interdependence (e.g., [79, 80]), or only a limited interdependency, by considering networks with limited interconnectivity, such as motorway networks (e.g, [81]). This introduces the first major limitation: limited *scalability*.

2.4.2. REAL-SCALE VALIDATION: DEGRADATION MODELS AND MULTI-OBJECTIVE OPTIMISATION

Further limitations exists within the degradation prediction models, restricting accurate degradation modelling, which is essential to effective PdM decision-making systems. While data quality and availability have been limiting factors in the past, data-driven degradation models have become more and more prevalent and have been extensively explored, using methods such as statistical inference, probabilistic graphs, including Markov models, and machine learning [82].

Such models have also been developed for the degradation of pavements (e.g., [83, 84, 85, 86, 87, 88, 89, 90, 91]). These models are typically calibrated using indicators of pavement condition, such as the pavement condition index (PCI), international roughness index (IRI), pavement serviceability index (PSI), or pavement condition rating (PCR). While these factors indicate pavement condition, they lack specification of exact damage modes. Additionally, most studies focus on asphalt and concrete pavements, for which these indicators are available, limiting their applicability to urban road networks characterised by heterogeneous attributes, such as diverse pavement materials, traffic loads, damage modes and other functional classifications [92].

Moreover, models explicitly quantifying the effect of maintenance activities on road condition are currently limited. Although some studies have explored post-maintenance condition changes (e.g., [93, 94]), there is limited integration between degradation and maintenance modelling in a unified probabilistic framework. This omission constrains the accuracy of long-term maintenance planning and limits the potential for rigorous predictive optimisation methods to be applied in practice (e.g., [95, 96, 97, 98, 33]).

Finally, the integration of predictive modelling techniques into real-world infrastructure maintenance planning has remained limited, particularly at scale and within complex urban networks. This is due not only to the computational and methodological challenges associated with high-dimensional, dynamic systems, but also to fragmented data landscapes and the limited interoperability of existing models. Many current models tend to focus on individual assets or small networks, often optimizing for cost alone, lacking support for multi-criteria evaluation and network-level reasoning. These limitations can be summarized as one gap: a lack of *real-scale validation*.

2.4.3. INTERPRETABILITY AND EXPLAINABILITY

Finally, a key consideration in applying AI-based models to infrastructure degradation modelling and maintenance planning is the interpretability and explainability of model behaviour. While AI models often achieve high predictive accuracy, they are typically black boxes whose internal reasoning is difficult to verify. Even when predictions are correct, there is a risk of "getting the right answer for the wrong reasons". Understanding when and why a model produces certain outcomes is therefore critical for assessing reliability.

Interpretability and explainability are related but distinct concepts [99, 100, 101]. Interpretability concerns how well a model's mechanisms or outputs can be understood by humans, while explainability focuses on attributing predictions to specific inputs and clarifying the reasoning behind decisions. These definitions may be implified to *how* an outcome is achieved (interpretability) and *why* it is achieved (explainability) [101]. Some authors place these terms on a spectrum, from inherently interpretable models to opaque models requiring post-hoc explanations [102, 103].

In current work on RL research for infrastructure maintenance planning, interpretability remains largely unaddressed. In broader contexts, however, several methods have been developed. Common post-hoc methods include SHAP (SHapley Additive exPlanations) and LIME (Local Interpretable Model-agnostic Explanations), which approximate a model's behaviour to indicate which features most influence predictions [104, 105, 106, 107]. These model-agnostic techniques have also been adapted for RL (e.g., [108, 109, 110, 111]).

More specialised extensions, such as Deep-SHAP [112] and SVERL (Shapley Values for Explaining Reinforcement Learning) [113], DL-LIME [114], ETeMoX (Event-driven TEmporal MOdels for eXplanations) [115], provide deeper insight into RL decision processes. However, such post-hoc methods reconstruct reasoning after training, potentially leading to incomplete or misleading explanations.

For this reason, intrinsic methods are often preferred. These embed interpretability within the learning process, producing models that are explainable by design. Examples include Programmatically Interpretable RL (PIRL) [116], which expresses policies as human-readable rules, and Symbolic Deep RL (SDRL) [117], which combines deep learning with symbolic reasoning. While these enhance transparency, they can be difficult to scale to complex, high-dimensional environments.

In summary, interpretability and explainability are essential for trustworthy, verifiable decision support in predictive maintenance. Post-hoc methods increase transparency but may oversimplify reasoning, whereas intrinsic approaches offer stronger guarantees at the cost of scalability. Both remain active research challenges. This is particularly important in predictive infrastructure maintenance, where complex decisions can have network-wide consequences; if the reasoning behind an AI-generated strategy is inaccessible, its practical applicability cannot be critically assessed. Therefore, this research prioritises interpretability over explainability, while acknowledging the usefulness of explainable tools in cases where interpretability is limited. Therefore, the final major research gap is defined as: limited *interpretability*.

2.5. EMERGING DIRECTIONS AND OPPORTUNITIES

Although underexplored in infrastructure maintenance planning, several RL approaches could be adapted to address these research gaps. Graph Neural Networks (GNNs), Hierarchical Reinforcement Learning (HRLs), Hierarchical Coordinated Reinforcement Learning (HCRLs), Inverse Reinforcement Learning (IRLs) and Imitation Learning (ILs) Methods are addressed in this section.

2.5.1. GRAPH NEURAL NETWORKS (GNNS)

Graph Neural Networks (GNNs) are AI architectures designed to operate on graph-structured data [11, 13, 118]. Unlike traditional neural networks that process fixed structures such as sequences or images, Graph Neural Networks (GNNs) can learn both the features of individual graph nodes and the structural relationships encoded in the graph topology. This makes them particularly suitable for relational data, such as transportation networks (e.g., [119, 120, 121, 16, 122, 123]). GNNs can be applied to the node, edge, or graph level, predicting properties of nodes, relationships between them, or overall graph characteristics [11]. Although typically trained in a supervised manner, GNNs can also be used for unsupervised or semi-supervised learning.

At their core, most GNNs use a *message passing* framework [119], where each node iteratively aggregates information from its neighbours and updates its representation. In each iteration (or layer) of message passing, a node u receives messages from its neighbours $\mathcal{N}(u)$, aggregates them using a permutation-invariant function such as `sum`, `mean`, or `max`, and combines the result with its own previous embedding through an update function, typically a neural network, such as a fully connected layer or MLP. Repeating this process over multiple layers allows information to propagate through the graph and enables nodes to learn context from increasingly distant neighbours. The final node embeddings can then support tasks such as node classification, link prediction, or graph-level prediction [13].

To make predictions at the level of an entire graph, *pooling* (or READOUT) functions aggregate node representations into a fixed-size vector [124]. These functions can be implemented as a simple summation or averaging, although more advanced pooling methods such as hierarchical, attention-based, or differentiable pooling [125, 126] are often used for complex graph-level tasks.

GNNs can be broadly grouped into four architectural families [11]:

- **Recurrent GNNs (RecGNNs)** iteratively update node states until convergence, forming the basis of early GNN formulations.
- **Convolutional GNNs (ConvGNNs)** generalise convolution operations to graphs, aggregating neighbour information across layers to learn higher-level representations.
- **Graph Autoencoders (GAEs)** encode the graph into a latent space and can reconstruct its structure, supporting tasks like embedding and generation.
- **Spatial–Temporal GNNs (STGNNs)** combine graph convolutions with temporal models (e.g., RNNs or attention mechanisms) to capture dynamic processes such as traffic forecasting.

2.5.2. Hierarchical Reinforcement Learning (HRL) & Hierarchical Coordinated Reinforcement Learning (HCRL)

In Hierarchical Reinforcement Learning (HRL), decision-making is structured hierarchically across at least two levels. Usually, higher-level agents define goals, which are then translated into specific actions by lower-level agents. This abstraction helps reduce the size of state and action spaces, improving scalability [78]. Coincidently, interpretability can be improved, as over-arching decisions made by high-level decision-makers can provide insight into more specific decisions made by lower-level agents. [78] demonstrate the applicability of an Hierarchical Reinforcement Learning (HRL) framework in a case-study. This work concludes that HRL is more scalable than existing frameworks.

Hierarchical Coordinated Reinforcement Learning (HCRL) combines HRL with Coordinated Reinforcement Learning (CRL). In CRL, a coordination mechanism is applied to the maintenance decision-making process, for example through a system reliability block diagram, which explicitly models interdependencies in a system [127]. This makes Hierarchical Coordinated Reinforcement Learning (HCRL) particularly suitable for systems where some components or interdependencies are considered more critical than others. One study demonstrates that HCRL can generate more cost-effective policies than a traditional RL method [127]. Furthermore, they suggest it may outperform other DRL methods due to its higher customizability and ability to model interdependencies directly.

The main difference between the described frameworks and MARL is how interdependencies are modelled. MARL methods rely on agents learning interdependencies implicitly through training, while HRL and HCRL approached use more explicit modelling in hierarchical or coordinated structures. The suitability of each framework likely depends on the characteristics of the system and its maintenance objectives. However, no comparative studies evaluating these methods on the same system have been found by the author at the time of writing.

2.5.3. Inverse Reinforcement Learning (IRL) & Imitation Learning (IL) Methods

In Inverse Reinforcement Learning (IRL), the goal is to infer the underlying reward function that explains observed expert behaviour [128]. Because this reward function directly guides agent behaviour, Inverse Reinforcement Learning (IRL) can provide intrinsic explanations for policy decisions (e.g., "The agent chose action a in state s because it maximizes reward for safety and efficiency"). However, reward-based explanations may still include some ambiguity; several reward functions can explain the same model behaviour. To reduce this ambiguity, IRL may be combined with H(C)RL methods that better structure the decision space. This is called Hierarchical Inverse Reinforcement Learning (HIRL) (e.g., [129]).

IRL can be extended to apprenticeship learning, where agents learn expert-like policies directly from demonstrations by optimising the inferred reward using algorithms such as Trust Region Policy Optimisation (TRPO) [130]. Yet, when the true reward structure is highly complex, these methods may fail to reproduce expert behaviour reliably [131].

A related family of approaches is Reinforcement Learning (RL), which trains agents to imitate expert behaviour without explicitly recovering the reward function. For example, Generative Adversarial Imitation Learning (GAIL) sets up a competition between two models, similar to a Generative Adversarial Network (GAN): a policy generator imitates the expert and a discriminator distinguishes between expert and agent actions. Training continues until the discriminator can no longer tell them apart, allowing the agent to reproduce expert-like behaviour more efficiently than traditional IRL-based methods [132].

While Imitation Learning (IL) methods improve scalability, IRL-based frameworks remain more interpretable because they recover the underlying reward function. Adversarial Inverse Reinforcement Learning (AIRL) combines both advantages by using a GAN-like setup to learn not only expert-like policies but also the reward function that explains them [132]. As with IRL, this may be combined with HRL in Hierarchical Adversarial Inverse Reinforcement Learning (H-AIRL) (e.g., [133, 134]).

2.6. CONCLUSION

This chapter has reviewed theoretical foundations and state-of-the-art methods in infrastructure inspection, degradation modelling, and RL-based maintenance planning. While significant progress has been made, there remain gaps in:

1. Scalability;
2. Interpretability; and
3. Real-scale Validation.

These gaps form the basis of the research contributions in this project. (Partially observable) MDPs, (deep) RL, MARL, HRL, HCRL, GNN, IRL and IL provide promising methods in these contexts and will form the foundational framework on which the work of this PhD will be built.

3

Proposed PhD Research

This chapter comprises all foundational aspects of the proposed research. It includes the research question & objectives (as introduced in Chapter 1), as well as the methods, and a workplan. Next, a brief introduction to the collaborative framework of the PhD is given. This chapter concludes with a section on the wider research impact and relevance.

3.1. Research Question & Objectives

The three research gaps, identified in Section 2.4 as scalability, interpretability and real-scale validation, together have led to one main research question:

> ***"How can a data- and AI-driven decision-support framework be developed for multi-year, urban-scale inspection and maintenance planning that (1) scales to thousands of infrastructure components, (2) produces interpretable intervention policies for practitioners, and (3) can be validated on real-world heterogeneous infrastructure networks?"***

To address this question, the following research objectives are considered, breaking down what needs to be understood, developed, and validated:

RO-1. *To analyse the requirements for urban-scale maintenance planning systems.*
This includes identifying the data, computational, and organisational requirements, as well as the optimisation objectives, for decision-support frameworks operating across large, heterogeneous infrastructure networks under uncertainties and stochastic degradation. This objective clarifies what "urban-scale" implies in practice and ensures the framework is grounded in real-world constraints.

RO-2. *To develop data-driven models for predicting infrastructure condition evolution.*
This includes formulating and testing predictive degradation models that can efficiently predict future conditions for roads with diverse characteristics and with varying inspection frequencies, data availability, and uncertainty levels. This enables accurate, real-scale prediction of infrastructure conditions.

RO-3. *To design scalable decision-making mechanisms for maintenance planning.* This includes developing algorithms that generate maintenance or intervention policies that can efficiently handle thousands of interconnected components. This supports scalability for urban-scale systems.

RO-4. *To design interpretable decision-making mechanisms for maintenance planning.* This includes developing algorithms that generate maintenance or intervention policies that are actionable, reliable and understandable for practitioners. This ensures transparency and trust in model decision-making and outputs.

RO-5. *To develop and validate a unified decision-support framework using real-world infrastructure data.* This includes combining all components (the degradation model and interpretable, scalable decision-making framework) into a coherent, modular framework that supports multi-year maintenance planning at the network level, and applying this to real inspection and maintenance datasets to evaluate its predictive accuracy, scalability, interpretability, and practical value. This objective bridges research and practice through empirical validation.

3.2. METHODS

To achieve the objectives outlined above, the research will follow a data-driven approach supported by both empirical analysis. Each method described below addresses a specific research objective. Collectively they contribute to developing and validating a scalable, interpretable decision-support framework for urban infrastructure maintenance planning:

MET-1. *Analysing the requirements for urban-scale maintenance planning systems* (RO-1) involves a review of current practices and datasets used in municipal maintenance planning. This is already completed and presented in Chapter 2 and Chapter 4.

MET-2. *Developing data-driven models for predicting infrastructure condition evolution* (RO-2) involves exploring stochastic and probabilistic models, including Markov models, for use with real-world inspections data.

MET-3. *Designing scalable decision-making mechanisms for maintenance planning* (RO-3) involves extending existing MARL architectures. One approach is to use the inherent graph-like structure of infrastructure systems with GNN-based architectures. another approach is to use HRL-based architectures to decompose the maintenance decision-making into smaller sub-problems.

MET-4. *Designing interpretable decision-making mechanisms for maintenance planning* (RO-4) is focused on developing algorithms that generate transparent decision-making and explainable maintenance policies. Potential approaches include the use of prior policies or other expert inputs, using IRL- and IL-based approaches. While post hoc methods may be considered, they are not preferred for reasons outlined in Section 2.4.3.

MET-5. *To develop and validate a unified decision-support framework using real-world infrastructure data* (RO-5) involves connecting the identified model requirements and optimisation objectives (1) with the developed predictive degradation model (2), the scalable decision-making model architecture (3), and the developed interpretability methods (4). This step aligns model inputs and outputs within a consistent structure and establishes the data flow required for coordinated operation. Then, this framework is tested using real-world datasets and infrastructure networks. Performance metrics include scalability, computational efficiency, predictive accuracy, interpretability, and practical usability.

3.3. WORKPLAN

Based on the research question posed in Section 3.1, the final deliverable can be formulated as:

> ***A validated, data- and AI-driven decision-support framework for multi-year, urban-scale maintenance planning that (1) scales to thousands of infrastructure components, (2) produces interpretable intervention policies for practitioners, and (3) can be validated on real-world heterogeneous infrastructure networks.***

To deliver this main outcome, the research is organised into a multi-project structure with one main project and four defined subprojects, which are further divided into subtasks. This section first introduces the general structure of these projects. Then, each (sub)project is linked to research objectives and described in more detail.

The main project is:
M: *Maintenance Decision-Support Framework*

The four subprojects are:
S1: *Predictive Infrastructure Condition Model*
S2: *Infrastructure Network Model*
S3: *Interpretability Framework*
S4: *Maintenance Planning Framework*

This structure, illustrated in Figure 3.1, is hierarchical; the outcomes of subtasks and -projects feed into their respective parent projects. The structure of the project is therefore roughly chronological (Section 5.1 presents a timeline). Both the main and subprojects each include a set of research tasks, which contribute to filling a research gap:

1. **Scalability**: S2, S4 & M
2. **Interpretability**: S2, S3, S4 & M
3. **Real-scale validation**: S1, S2 & M

SUBPROJECT OBJECTIVES

In the design of the subproject research objectives, emphasis is placed on aligning research efforts with current practice, while also incorporating forward-looking perspectives that address emerging challenges and evolving performance criteria, such as shifts in available budgets, the growing impacts of climate change, and municipal policies. The objectives are formulated with the intend to be specific enough to effectively guide detailed research and provide a clear research direction, yet sufficiently broad to cover the complexity and variability inherent in the research topics. The main project M directly results in the final deliverable as defined at the start of this section. The research objectives for subprojects S1- S4 are shown below.

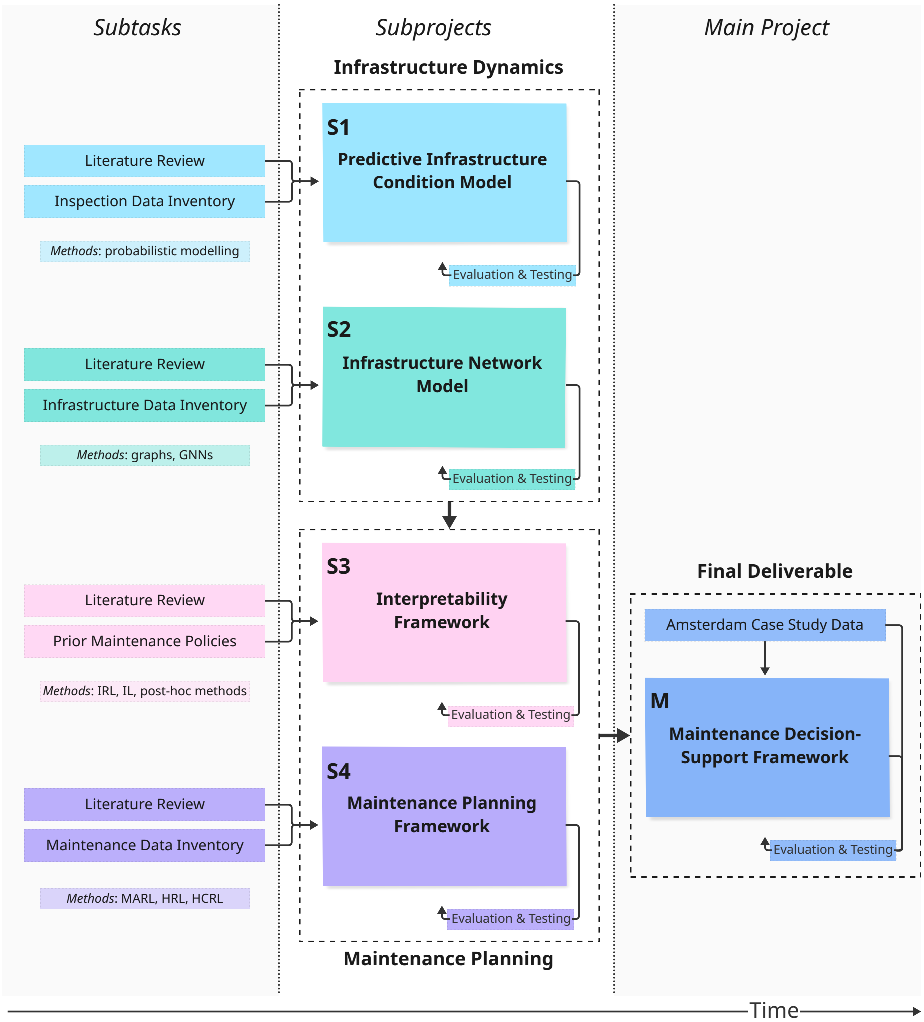


Figure 3.1: An overview of main and sub-projects. Projects are organised in a hierarchical way, where each subtask and -project is necessary for their respective parent project.

***S1**: Predictive Infrastructure Condition Model*
Research Objectives: 1 & 2
This subproject involves developing a model for stochastic degradation process and maintenance effects to probabilistically predict the future condition of infrastructure assets, using historical data and probabilistic deterioration modelling. It should account for uncertainty in pavement degradation and maintenance outcomes. It also includes a data analysis, answering questions such as "How complete, accurate, and suitable are these data for supporting predictive modelling, and how might this influence model structure,

assumptions, and predictive performance?"

***S2**: Infrastructure Network Model*
Research Objectives: 1, 3 & 4
This subproject involves developing a model capable of representing urban-scale infrastructure networks, capturing interdependencies between road segments, traffic dynamics, and capacity constraints. This includes researching how characteristics of infrastructure networks can be represented so that they are well-suited for integration with predictive maintenance planning frameworks.

***S3**: Interpretability Framework*
Research Objectives: 4
This subproject focusses on developing interpretable frameworks that can be integrated into the predictive inspection & maintenance planning frameworks, and specifically the models to be developed in ***S4**: Maintenance Planning Framework*. First, questions such as "How are maintenance decisions made in current practice, and what data and tools are used to support them?" are considered to understand the degree of interpretability in existing practical models. Second, we research existing methods for interpretability within RL frameworks, and consider their applicability to our case. Then, these models are adapted to the case of maintenance planning. In such frameworks, prior maintenance policies, developed by human experts, may be used.

***S4**: Maintenance Planning Framework*
Research Objectives: 3
This subprojects involves developing a model for maintenance planning that addresses the scalability research gap and up-scales existing methods. To do this, we first answer questions such as "What are the key strengths and limitations of existing models?". Then, we focus on how these limitations can be lifted within the context of infrastructure maintenance planning. As part of the RL structure, this project also involves development of a reward model (i.e., an evaluative function quantifying the desirability of maintenance strategies in terms of system performance, economic cost, societal value, and environmental impact) capable of comparing maintenance policies through multi-criteria evaluation. This, in turn, involves answering questions such as "What performance indicators are currently considered in infrastructure maintenance planning?" and "What performance indicators could inform the development of future reward models?".

***M**: Maintenance Decision-Support Framework*
Research Objectives: 5
This projects focusses on merging all developed frameworks together into one unified decision-support framework. Finally, it is validated using real-world infrastructure data, such as the data provided by the municipality of Amsterdam. This data is provided to this research by the TBL-Infra consortium, which is explained in Section 3.5.

3.4. POTENTIAL RISKS AND RISK MITIGATION MEASURES

While this research aims to complete all research objectives (as described in Section 3.1) and the final deliverable (as described in Section 3.3), several risks may arise during its execution. These risks relate to data availability and quality, methodological challenges, and practical applicability of the developed framework. Identifying these risks at an early stage enables the design of appropriate mitigation strategies to maintain the project's feasibility, scientific validity, and relevance to real-world practice. The following section outlines the key potential risks and the measures planned to address them.

Risk 1: *Data Availability and Quality*
Limited access to inspection or maintenance data, incomplete historical records, or inconsistent data formats across assets and years.
This risk may impact S1, S2 and M.

Mitigation Measures: Although inspection and maintenance data have been provided by the municipality of Amsterdam, these datasets were not originally collected for the purpose of predictive maintenance modelling. As a result, they do not fully accommodate the requirements of such models. Preliminary results, presented in Chapter 4, indicate that the available data are sufficient to develop a predictive condition model (S1) and infrastructure model (S2). However, several limitations remain and have been identified (see Section 4.2). These are currently addressed through the selected modelling methods, which can handle features of the dataset, such as irregular time intervals between observations. If, during the remainder of the project, it becomes necessary further develop the model on additional data, this will be done using complementary datasets (from other providers) or synthetic data generated through simulation.

Risk 2: *Computational Challenges*
The selected modelling methods (e.g., reinforcement learning) may be computationally demanding, particularly when applied to large-scale infrastructure networks.
This risk may impact S3, S4 and M.

Mitigation Measures: In the early phases of model development, smaller test networks will be used to assess scalability and performance. Model simplification strategies, such as reducing the state–action space, may also be applied where appropriate. If computational demands exceed local capacity, simulations will be executed on DelftBlue, TU Delft's high-performance computing facility.

Risk 3: *Validation and Generalisation*
The final model (M) may perform well on the City of Amsterdam dataset but fail to generalise to other cities, pavement types, or maintenance strategies.
This risk may impact all subprojects and M.

Mitigation Measures: To ensure quality and generalisability, the model will be validated across different subsets of assets. In addition, the framework will be tested on at least one external dataset or hypothetical network to assess transferability. All modelling assumptions will be clearly documented to support reproducibility and adaptation to other contexts.

3.5. COLLABORATIVE FRAMEWORK

This PhD project is funded by the *Nationaal Groeifonds* (Dutch National Growth Fund) and is part of the broader programme *Toekomstbestendige Leefomgeving* (Future-Proof Living Environment), conducted in collaboration with the TBL-Infra consortium. Within this programme, the PhD is situated in Project 2: *Levensduurverlenging* (Life-Cycle Extension), and more specifically in subproject (SP) 1.2.2 and work package WP12: *Decision Support System Levensduurverlenging* (Decision Support System for Life-Cycle Extension). An overview of the project structure within the TBL-Infra consortium is provided in Figure 3.2. This section outlines the collaborative framework and situates the PhD project within the wider programme context.

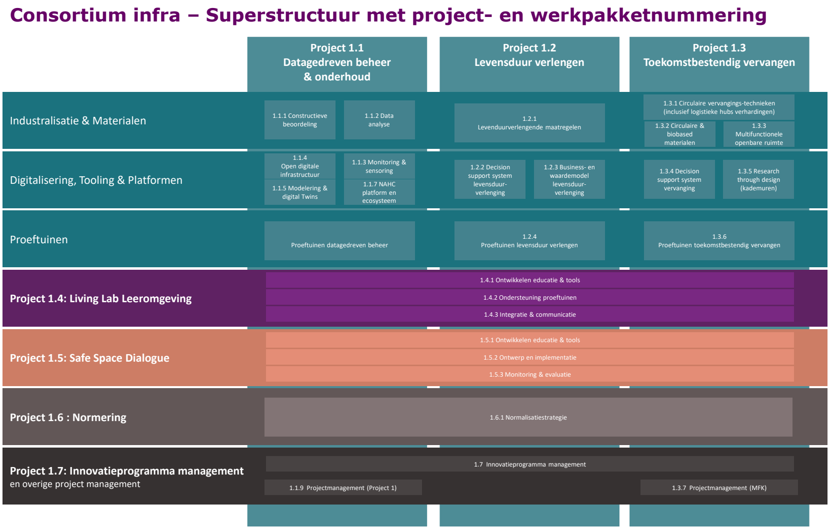


Figure 3.2: An overview of projects in textitToekomstbestendige Leefomgeving (Future-Proof Living Environment): Project 2 *Levensduurverlenging* (Life-Cycle Extension). This PhD project is part of SP 1.2.2: *Decision Support System Levensduurverlenging* (Decision Support System for Life-Cycle Extension)

3.5.1. TBL-INFRA CONSORTIUM

The TBL-Infra consortium is a collaborative framework comprising public infrastructure clients, contractors, engineering and consultancy firms, academic institutions, and emerging enterprises such as start-ups and scale-ups. It adopts a programmatic and project-based approach to innovation, with clearly defined roles and responsibilities for all partners. By representing the full infrastructure value chain, the consortium facilitates the integration of complementary expertise and supports the development of innovations with practical applicability. Its open structure allows for the inclusion of additional partners over time. Together, the consortium focuses on the development and testing of new design strategies and working methods aimed at extending the functional lifespan of existing infrastructure, including road networks. As a part of this goal, this PhD project contributes to the development of data- and AI-driven methods to support decision-making for life-cycle extension of road infrastructure. The research outputs are expected to support the broader aims of the TBL-Infra programme by contributing to a shift towards long-term, predictive asset management practices. In turn, the consortium supports the PhD by providing historical inspection, maintenance, and infrastructure data, which can be used to model deterioration and to predict infrastructure performance under different maintenance interventions. It also allows us to address the need for real-scale validation of the developed frameworks, which is one of the key research gaps.

3.6. Research Impact & Relevance

This section outlines the expected scientific, practical, and societal contributions of the proposed PhD research.

At the scientific level, the research addresses current gaps in applying machine learning, particularly RL, to large-scale, stochastic, and multi-objective infrastructure systems. It develops novel RL-based modelling approaches for urban networks that integrate data-driven optimisation and policy generation within infrastructure asset management. The resulting methods are designed to be modular and adaptable for other infrastructure types or policy contexts.

Practically, the research supports maintenance decision-making in real-world settings. By linking predictive condition and network models, the developed framework helps practitioners design long-term maintenance strategies consistent with current data, practices, and constraints. Collaboration with the municipality of Amsterdam ensures the methods address actual management needs, while the model outputs are designed for interpretability and practical use (for example through visualisation and interpretable policy comparisons, though not as a user interface).

Societally, the research advances sustainable urban development by enabling infrastructure managers to evaluate environmental, economic, and social trade-offs in long-term planning. It promotes transparent, data-driven, and equitable decision-making that supports climate-resilient and resource-efficient infrastructure systems.

Summary of contributions:

- **To science:** Novel modelling approaches for predictive infrastructure maintenance, advancing RL for large-scale, multi-objective planning.
- **To practice:** An interpretable decision-support framework for evaluating and comparing long-term maintenance strategies under realistic constraints.
- **To society:** Methods that promote sustainable, cost-efficient, and policy-aligned infrastructure systems.

4

PRELIMINARY RESULTS

This chapter covers the preliminary research results, including a data inventory, maintenance cost analysis, an overview of traditional maintenance strategies, a degradation model and an infrastructure graph model.

4.1. INTRODUCTION

This chapter provides some foundational preliminary research results, including subtasks of S1- S4 and initial versions of S1 and S2. This chapter will follow the following structure:

- Section 4.2 presents a data inventory and analysis on the inspection and maintenance data provided by the municipality of Amsterdam. This forms the basis for the predictive model of S1.
- Section 4.3 includes a brief overview of existing maintenance strategies, which may be used to inform IL models in S3.
- Section 4.4 presents a summary of the research on the performance indicators and their relationships. It forms a basis for the reward function for S3 and S4.
- Section 4.5 includes an analysis of costs per maintenance intervention, which can inform the reward function for S3 and S4.
- Section 4.6 forms a basis for S2, creating infrastructure graphs that allow for traffic modelling.
- Section 4.7 and Section 4.8 include the predictive infrastructure condition model of S1, modelling degradation and maintenance effects on road segments respectively, based on the data analysed in Section 4.2

4.2. DATA INVENTORY & ANALYSIS

In this section, an overview of available inspection and maintenance data is presented. This includes all data that is currently made available to the research by consortium partners (see Section 3.5). For all datasets, data gaps and uncertainties are also discussed.

4.2.1. INSPECTION DATA

Inspection data is used to give insight into the historical conditions of roads. This information is essential to model road degradation. For the purposes of this research, the City of Amsterdam has provided historical data on road conditions. This dataset consists of road inspection data from the Amsterdam infrastructure network, collected according to the CROW 146b protocol, per road segment. This data is used by the City of Amsterdam for planning asset maintenance, administrative reporting and complying with legal requirements (City of Amsterdam, personal communication, March 5, 2025). In the dataset, road segments are defined as any section of road without crossroads or intersections, with a maximum of 100 meters in length. A selection of road segments is shown in Figure 4.1. For each road segment, feature data is also available. Features include location information (i.e., neighbourhood, municipality, region), road construction category (i.e., asphalt, concrete, tiled pavement, semi-paved, synthetic, and unpaved), exact road surface material, year of construction or year of most recent full reconstruction of road surface, year of most recent full conservation measures (surface treatments), surface area (m^2) and road traffic intensity and function labels (i.e., heavy, average, light, residential, commercial/pedestrian, and cycle paths). It should be noted that the data may be incomplete for some road segments.

Figure 4.1: A section of Amsterdam's road segment map, in Amersfoort / RD New coorinate system (espg:28992, unit:meters)

The inspection data is available from 2015 onward and is collected roughly every two to three years. It consists of labels, indicating condition states of the road segment as a whole, reflecting the label to describe the worst damage case in a road segment. Labels include: $G, L1, L2, L3, M1, M2, M3, E1, E2$, and $E3$. In these labels, the letter represents damage severity (G = good/no damage, L = minor, M = moderate, E = serious), while the number indicates the extent of the affected area (1 = small, 2 = average, 3 = large). Eight categories of damage (damage modes) are considered: (1) transverse unevenness, (2) irregularities, (3) ravelling, (4) edge damage, (5) crack formation, (6) joint filling, (7) joint width and (8) settling. The relevance of specific damage modes is determined by the pavement construction category; up to six distinct damage types are relevant per road segment. A snippet of this dataset is presented in Table 4.1.

Table 4.1: Inspection records of road segments with various types of damage.

ID	Date	**Transverse Unevenness** *(Dwarsonvlakheid)*	**Irregularities** *(Oneffenheden)*	**Ravelling** *(Rafeling)*	**Edge damage** *(Randschade)*	**Crack formation** *(Scheurvorming)*	**Joint filling** *(Voegvulling)*	**Joint width** *(Voegwijdte)*	**Settling** *(Zetting)*
36885	20-09-16	G	L3	G	G	G	G	G	G
36885	29-05-18	G	G					G	G
36885	16-04-21	G	M3					G	G
36885	27-03-23	G	M3					G	
36886	20-09-16	G	L3	G	G	G	G	G	G
36886	29-05-18	G	G					G	G
36886	16-04-21	G	G					G	G
36887	20-09-16	G	M2	G	G	G	G	G	G
36887	29-05-18	G	G					G	G
36887	16-04-21	G	M1					G	G

DATA GAPS AND UNCERTAINTIES

Data collection methodology

Inspection data is collected through manual inspection, following CROW 146b protocol. This protocol describes the boundary conditions per label for each damage mode. However, experts from consortium partners believe that a certain human error persists in this data (City of Amsterdam, personal communication, March 5, 2025 & Heijmans, personal communication, March 10, 2025). This uncertainty complicates modelling of the degradation process and maintenance effects.

Geospatial resolution

In the dataset, data is recorded per road segment, defined as a subsection of road of maximum 100 meters in length, with no crossroads. Within this resolution there is a risk for traceability issues, as there may be several damaged areas within one road segment, while the condition label refers only to the most significantly damaged area (as described in the CROW 146b protocol). As a result, the total extent of damage on the road segment is not recorded. For example, if the road segment damage is labelled $M2$, there may be one damaged area labelled $M2$, or several damaged areas that would be labelled, e.g., $L1$, $L2$ and $M2$ if considered separately. However, as these more minor damages are not recorded, damage development cannot be traced per occurrence. Additionally, for any transition that shows an improvement to any state other than G, such as $M2 \rightarrow L1$, it is unclear whether partial or imperfect maintenance has taken place on one damaged area, or whether the $M2$-level damage has been completely resolved, making another unaddressed damaged area labelled $L1$ the most significant damage on this road segment during the next inspection. Alternatively, such an 'imperfect increase' transition could be caused by human inconsistencies, as described prior. These traceability issues could be reduced or removed by utilizing a more detailed spatial resolution focussed on each individual damaged area. Examples include a damage bounding box, or a fine grid overlay for which the damage state can be indicated.

Temporal resolution
The inspection data has been collected since approximately 2015, and is recorded roughly every two to three years per road segment. The histogram in Figure 4.2 shows the distribution of time gaps between inspections. As road degradation is slow and maintenance is generally performed before severe damage states are reached, it is unlikely that the data captures the full degradation process for road segments. However, individual transitions can be used to calculate transition probabilities for road segments with similar features. Hence, the temporal resolution of the data is considered sufficient.

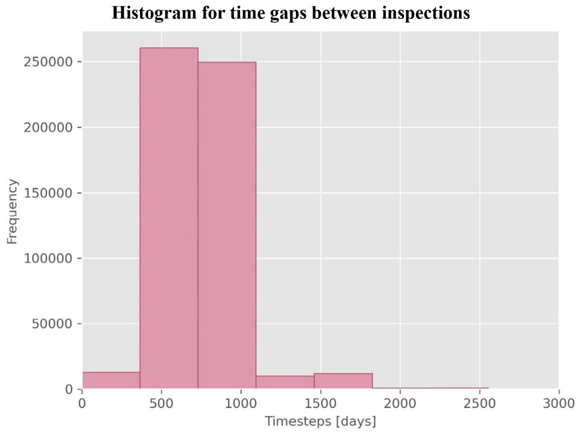


Figure 4.2: Frequency of time gaps between inspections, bar size = 365 days

Data type
As mentioned, the road conditions in this dataset are classified in four categories for damage extent ($G/0,1,2,3$), four categories for damage severity (G,L,M,E) and eight damage modes. No further details, such as exact locations, length or depth of damage, are collected. More detailed information would facilitate more accurate modelling of damage development.

Future Inspection Data Collection Methods
Various novel technologies enable automated data collection, reducing inconsistencies found in manual inspections. These methods typically involve depth cameras and object-recognition algorithms to detect and classify damage. Damage classification can follow any chosen scale, depending on how the classification algorithms are designed and trained. This flexibility would allow overcoming the limitations of the CROW damage

classification scale, as previously described. However, at this moment, historic inspection datasets using such methods are not available to the research.

4.2.2. MAINTENANCE DATA

Maintenance data inform the modelling of the natural degradation process and maintenance effects. By combining maintenance and inspection datasets, individual changes in road conditions can be correctly attributed to these processes. The municipality of Amsterdam has provided maintenance data to this PhD research, in which maintenance actions are attributed one of the following categories:

- Minor Maintenance (Dutch: *Klein Onderhoud, KO*)
- Planned Minor Maintenance (Dutch: *Planmatig Klein Onderhoud, PKO*)
- Major Maintenance/Replacement (Dutch: *Groot Onderhoud/Vervanging, GOVV*)
- Damage Detection Reports (Dutch: *Signaleringsmeldingen, SIG*)
- Daily Maintenance (Dutch: *Dagelijks Onderhoud, DO*)
- Regional Developments (Dutch: *Herinrichtingen/Gebiedsontwikkelingen, HI*)

Daily Maintenance (DO) contains a combination of Minor Maintenance (KO), Planned Minor Maintenance (PKO) and Damage Detection Reports (SIG). Major Maintenance (GO) described cases where maintenance is performed on pavement surfaces. Replacement (VV) includes full replacement of the pavement. Finally, Regional Developments (HI) includes replacement of pavement, including modifications of the profile, layout and/or functionality of a road (City of Amsterdam, personal communication, March 5, 2025).

Two maintenance datasets were made available: (1) Minor Maintenance (KO) and (2) Major Maintenance/Replacement (GOVV) + Planned Minor Maintenance (PKO). Data is collected on Damage Detection Reports (SIG), but this data is currently unavailable to the research. No data is collected and/or available to the City of Amsterdam regarding Regional Developments. The KO dataset includes planned intervention locations (point coordinates), identified by road inspectors following the CROW 146b protocol. It contains 25044 datapoints in total, which can be linked to 17813 unique road segments. The GOVV dataset contains 228 planned maintenance for larger areas or neighbourhoods, represented as polygons or multi-polygons. Each of these (multi)polygons contains multiple road segments. The geometric information from both datasets is shown in Figure 4.3. Additionally, the KO dataset includes the data collection date, pavement category (general and specific), damage category (general and specific), damage size, and its unit. The GOVV dataset includes the data collection date, project name, category, budget, asset category and group, source, project status, and planned maintenance phases: preparation (Dutch: *Geplande Voorbereiding Uitvoering, GVU*), start (Dutch: *Geplande Start Uitvoering, GSU*), and completion (Dutch: *Geplande Einde Uitvoering, GEU*), each specified by year. It also records the total area (m^2).

DATA GAPS AND UNCERTAINTIES

Data completeness

As mentioned, the KO and GOVV datasets do not cover all maintenance interventions performed in Amsterdam. Other maintenance categories include Damage Detection Re-

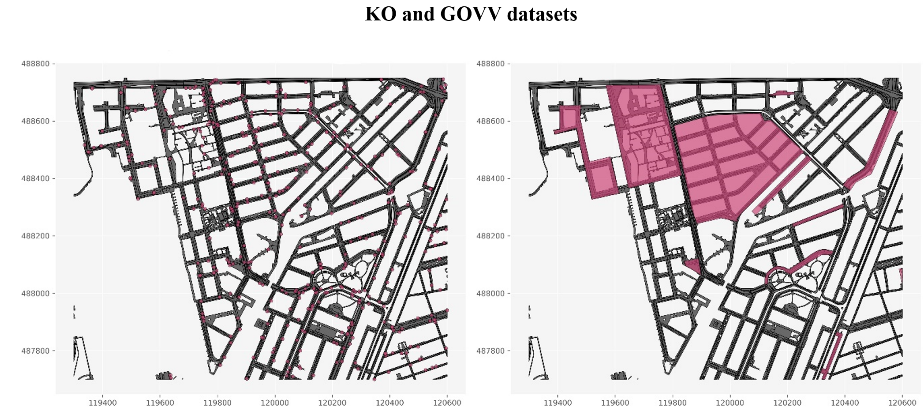


Figure 4.3: A section of Amsterdam's KO (left) and GOVV (right) maintenance datasets, in Amersfoort / RD New coordinate system (espg:28992, unit:meter)

ports (Dutch: *Signaleringsmeldingen, SIG*) and Regional Developments (Dutch: *Herinrichtingen/Gebiedsontwikkelingen*). At the time of writing, data on these maintenance categories are either unavailable or have not been provided for this research project.

Geospatial Resolution
The data in the KO dataset is geospatially described by points that indicate the location of damage, which can be linked to road segments, which in turn are described as (multi-)polygons in provided data. In the GOVV dataset, locations are given as (multi-)polygons reflecting areas that indicate neighbourhoods/regions in which maintenance is planned. These areas often includes multiple road segments. Experts confirmed (City of Amsterdam, personal communication, March 5, 2025) that these areas are maintained with the intention of repairing all significant damage.

Temporal Resolution
For the KO dataset, data entries range from 2018-12-17 to 2024-07-17 (yyyy-mm-dd). For the GOVV data ranges from 2019 for 2024 (planned maintenance start/end dates).

Data Accuracy
Both datasets are not records of historically perfomed maintenance, but instead includes planned maintenance, without confirmation of whether the maintenance has been performed. More specifically, the KO dataset only includes the date on which damage that required maintenance was noted. The GOVV dataset includes planned dates of maintenance projects. For both datasets, it is unknown whether these maintenance projects were executed, nor whether they were completed in the planned time period. These uncertainties make it difficult to match the changes in road condition found in the inspection data to maintenance actions from the maintenance datasets. This complicates degradation and maintenance effect modelling.

Additionally, maintenance information does not include details on the exact maintenance measures taken. Neither does it state if maintenance has been performed on all damage present in a road segment, or only select damage modes. Confirmation and exact dates of maintenance, as well as more detailed information on which damage mode is addressed would allow for more accurate degradation modelling.

4.3. MAINTENANCE STRATEGIES

A maintenance strategy describes how asset managers preserve the quality and safety of assets through planned maintenance interventions. Its goal is to optimise operational efficiency and extend the operational lifespan. Maintenance may restore an asset to its original condition (perfect maintenance) or partially improve its state (imperfect maintenance) [135, 136, 137]. Maintenance strategies are typically organised into two main categories: (1) reactive and (2) proactive [136, 138, 139, 140], though some literature defined four categories [141, 142]: (1) reactive, (2) preventive, (3) predictive, (4) proactive. In practice, multiple maintenance strategies may also be combined [139]. The latter categorisation will be used here.

Reactive Maintenance

Reactive maintenance refers to interventions after damage or failure has occurred [139, 140]. While cost-effective in the short term, it increases risks such as unplanned downtime, collateral damage, and reduced asset life [139]. Some authors distinguish between corrective maintenance (repairs after observed damage) and failure-based maintenance (repairs only after breakdown) [143, 142], though the terms are often used interchangeably. Specific forms include run-to-fail (RTF) strategies, where failure is deliberately allowed, and emergency maintenance, performed immediately after breakdown to minimise consequences [136, 143].

Preventive Maintenance

Preventive maintenance aims to avoid failures and retain asset condition by scheduling interventions in advance [141, 136, 82, 139]. Strategies include time-based (TBM), age-based, usage-based, and condition-based maintenance (CBM). These approaches rely on historical and real-time data to set maintenance intervals [82]. Time-based maintenance strategies are based on the in-service time of an asset, where maintenance is performed with a predetermined frequency [144, 136, 139]. This is sometimes also referred to as block replacement [135]. In age-based or age-dependent strategies, the maintenance intervals are determined depending on asset age [145, 135]. Similarly, in usage-based maintenance, interventions are scheduled based on usage frequency, using operational data [139]. Finally, condition-based maintenance use damage detection and condition diagnostics to plan maintenance before failure [146].

Additionally, work- and opportunity-based maintenance can be seen as examples of preventive maintenance [141]. Opportunity-based maintenance refers to maintenance strategies that focus on planning interventions at times at which components can be maintained without causing system failure [147]. No formal definitions of less common strategies, such as work-based maintenance could be found by the author, however.

Predictive Maintenance

Predictive Maintenance (PdM) extends preventive strategies by using prognostics or predictive models to forecast degradation and optimise timing [141, 138, 140]. Innovations in Industry 4.0 (I4.0), Internet of Things (IoT) and Machine Learning (ML) can increase PdM capabilities [140]. Preventitive strategies such as condition-based maintenance can become predictive when they incorporates degradation modelling [142, 136, 139]. Risk-based maintenance (RBM) is another PdM type, where interventions minimise the expected consequences of failure [139]. Compared to preventive maintenance, PdM offers more flexible scheduling and better balancing of risk, reliability, and cost. Finally, reliability-centred maintenance (RCM), though primarily considered predictive, integrates multiple maintenance strategies into one framework (reactive, preventive, and predictive), prioritising system reliability over restoring all components to ideal condition [136, 139].

Proactive Maintenance

Proactive maintenance strategies can be defined as improvements of the preset construction of components to prevent failures [141]. In other sources, it is defined more broadly as strategies aimed to prevent a system or its components from reaching a failure state though regular inspections and pre-emptive actions. This definition overlaps largely with preventive maintenance and sources define preventive maintenance as a sub-category of proactive maintenance.[143, 139]. Though it often involves higher initial costs, it reduces unexpected failures and enables scheduled downtime, which is generally more cost-effective in the long term. Additionally, this method allows for downtime to be scheduled in advance, minimizing operational disruptions [139].

Multi-component Strategies

Beyond individual components, maintenance may be planned at system level. Such multi-component strategies include (1) batch maintenance, (2) opportunity maintenance and (3) group maintenance [137].

In batch maintenance, multiple components are serviced according to a shared maintenance cycle [137, 148]. Opportunity maintenance combines several jobs into one intervention; when one component requires maintenance, others are evaluated, and those expected to need attention soon are repaired simultaneously, reducing overall downtime [149, 150]. Group maintenance coordinates components that require the same resources. This may involve allowing redundant components to fail before intervention (corrective group maintenance) or aligning schedules so that several components are serviced within the same interval (preventive group maintenance) [135, 137]. These approaches are examples of static group maintenance. Static models can be defined by: (1) T-age policies (age-based replacement), (2) m-failure policies (replacement after m failures), ((3) (m, T)-policies (whichever occurs first), and (4) T-policies (time-based replacement) [135].

In dynamic group maintenance, real-time condition data are used to optimise intervention timing [151, 152, 137]. Prognostic models inform when maintenance is most effective, making the strategy more adaptive. Two planning methods exist: the finite time axis method, which optimises interventions within a fixed horizon and suits systems with short service life. [153]; and the rolling time axis method, which uses a continuous horizon, updating schedules as new condition data become available [137].

Maintenance Strategies for infrastructure systems

Infrastructure networks can be considered complex systems consisting of many components, as discussed in section 2.3.1. For infrastructure systems, ensuring long-term reliability and operational efficiency is essential. At the same time, due to resource constraints, sustainability concerns and the need for all urban areas in the network to be accessible at all times, unnecessary maintenance actions should be avoided. Consequently, maintenance strategies for transportation infrastructure systems should aim to achieve a balance between system reliability and component downtime. Each of the strategies presented in this section may be applied to infrastructure systems. To ensure interpretability, these policies may be used as a basis for the decision-making models, as done in IRL (see Section 2.5.3)

4.4. REWARD MODEL & PERFORMANCE INDICATORS

In Section 2.2, a set of performance indicators for infrastructure maintenance strategies was presented. These have been researched to identify simulation methods as well as relationships between them. For brevity, the resulting causal model is presented here (Figure 4.4). Research consistently links overall network condition to key indicators such as vehicle operating costs, accident rates, and noise pollution levels. Other performance metrics, such as maintenance costs and emissions, are more directly related to the frequency and distribution of maintenance activities. However, the impact on traffic flow is highly dependent on the specific location and extent of the road segments affected by maintenance. Consequently, such effects should be considered separately. The interrelationships among these indicators are illustrated in Figure 4.4.

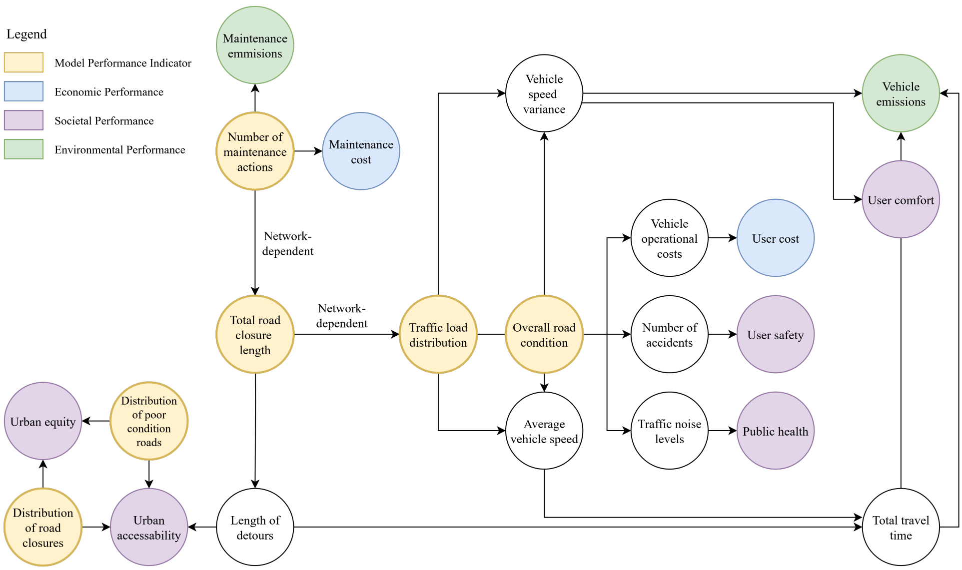


Figure 4.4: Causal model including primary performance indicators.

From this figure, we conclude that a reward function can be defined based on the following key indicators:

- Number of maintenance actions
- Total road closure length
- Distribution of road closures
- Traffic load distributions
- Overall road condition
- Distribution of poor condition roads

To include these indicators in a multi-objective optimisation framework, trade-offs must be defined to find a balance between all objectives, for example using the scalarisation approach or the Pareto approach [154]. This forms the basis for the reward model in M.

4.5. Maintenance Cost Analysis

In this section, an analysis of maintenance cost is presented, based on data on standard costs for various maintenance actions provided by the municipality of Amsterdam. This reflects the cost of maintenance actions by two road characteristics: (1) road construction material and (2) road functionality label (following CROW 146b protocol), which is essential to modelling realistic cost of actions in the RL framework. For each road construction material category, several maintenance actions are included in the dataset. In this analysis, road constructions are simplified to three categories: (1) Tiled pavements, (2) Asphalt pavements and (3) Concrete pavements. Additionally, the following functionality labels are considered:

- (2–) Roads with heavy traffic (*Zwaar belaste weg*)
- (3–) Roads with average traffic (*Gemiddeld belaste weg*)
- (4–) Roads with light traffic (*Licht belaste weg*)
- (5–) Residential roads (*Wegen in woongebied*)
- (6–) Roads in commercial and pedestrian areas (*Wegen in verblijfsgebied*)
- (7–) Cycle paths (*Fietspaden*)

Results

An initial analysis was conducted on maintenance costs across all road construction types and functionality categories. Further analyses were carried out for each combination of construction material and usage category. However, for the sake of brevity, only the results for the main categories are presented here. However, results for all combinations of these categories are available.

Results across all roads are shown in Figure 4.5. These include the average standard cost in EUR per maintenance action for all road construction and usage labels. This includes a total of 459 data entries, distributed as follows:

- **Rehabilitation**: 93 entries, relevant to all construction types
- **Evenness improvement**: 93 entries, relevant to all construction types
- **Elevation**: 41 entries, relevant to asphalt and concrete pavements
- **Preservation**: 41 entries, relevant to asphalt and concrete pavements
- **Strengthening**: 29 entries, relevant to asphalt pavements
- **Partial major maintenance**: 29 entries, relevant to asphalt pavements

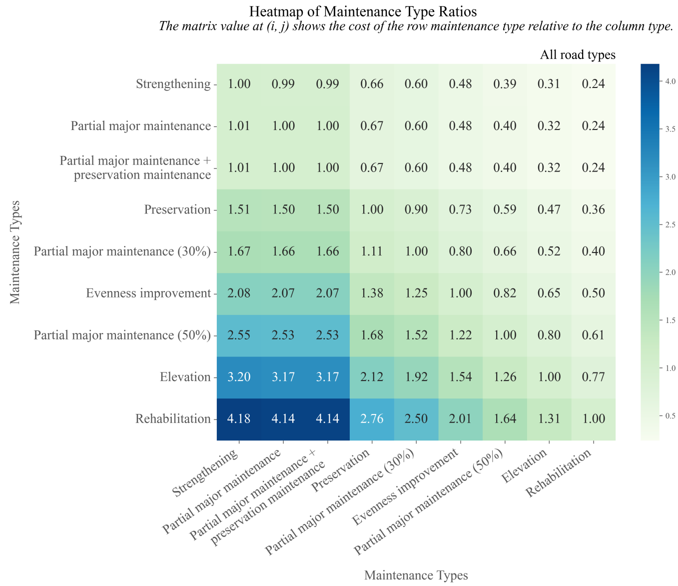


Figure 4.5: Ratios between standard costs of maintenance actions over all roads in the provided dataset.

- **Partial major maintenance 30%**: 52 entries, relevant to tiled pavements
- **Partial major maintenance 50%**: 52 entries, relevant to tiled pavements
- **Partial major maintenance + preservation maintenance**: 29 entries, relevant to asphalt pavements

Results show that rehabilitation is the most costly maintenance action. It is approximately 4.18 times as expensive as strengthening; the cheapest maintenance action. It should be noted that the actions labelled 'Partial major maintenance' are only used for asphalt pavements specifically. Partial major maintenance 30% and 50% apply to tiled pavements and are more expensive comparatively. Elevation is also more expensive compared to most partial major maintenance action. Preservation actions are also comparatively cheap.

4.6. INFRASTRUCTURE MODEL

*This section includes preliminary results of descriptive subproject **S2**: Infrastructure Network Model.*

The municipality of Amsterdam has provided a geoJSON file containing geometry information for the paved surfaces under their management. An algorithm was developed to convert this file into (benchmark) graph environments. Here, three example graphs that could function as an initial testing set for the main model M.

These examples graphs are:

1. network 1: a small network;
2. network 2: a medium-sized network; and
3. network 3: a large network

The networks are shown in Figure 4.6a-4.6c.

(a) Network 1: Smallest network.　(b) Network 2: Medium network.　(c) Network 3: Largest network.

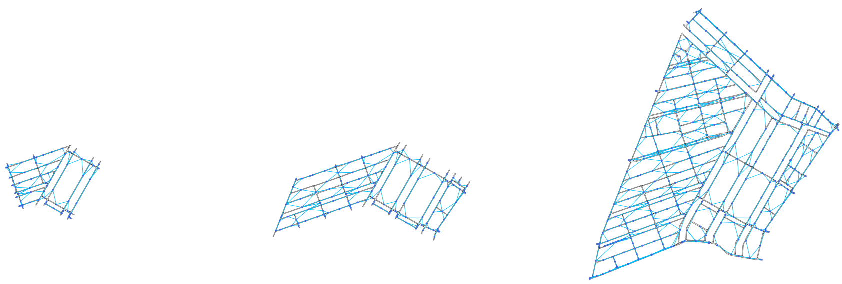
Figure 4.6: Selected networks from the Amsterdam road segment dataset, shown in blue.

The developed algorithm can convert these areas to graphs. Due to minor inaccuracies in the geoJSON file, the algorithm also provides an interface for minor manual refinement. These results are shown in Figure 4.7. Network 1 required no adjustments. In network 2, two edges were added. In network 3, three edges were added.

Next, a simple traffic model is applied to these graphs. Each node represents a road segment. Nodes are assigned the a condition and size (m^2). Edges represent the connections between the road segments. Each node $v \in V$ has an associated length l_v, capacity c_v, traffic load q_v, and condition label $s_v \in \mathscr{S}$, where $\mathscr{S}$ is the set of CROW-defined condition states.

Traffic is modelled using Origin-Destination (OD) pairs. Once these pairs are identified, the shortest path is calculated. Next, the number of times that each edge is used is counted. This number reflects the traffic load. An example of traffic load distribution is shown in Figure 4.8, for network 3. This method can also be used to calculate the travel time between OD pairs (using the road segment lengths).

This method forms the basis for traffic modelling. When maintenance actions limit road capacity, the simulation results change. The resulting traffic load values and travel times can be used to quantify network-wide effects, as described in Section 2.3.1.



(a) Network 1: Smallest network.　(b) Network 2: Medium network.　(c) Network 3: Largest network.

Figure 4.7: Manually refined graphs from networks from the Amsterdam road segment dataset.

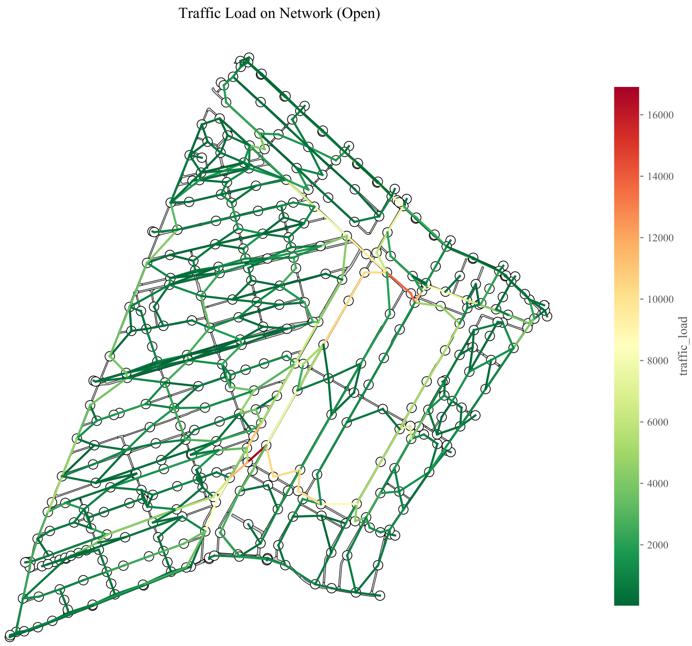


Figure 4.8: The traffic loads, modelled using OD pairs between all nodes.

4.7. DEGRADATION MODEL

*This section includes preliminary results of descriptive subproject **S1**: Predictive Infrastructure Condition Model. Results presented here are also included in the paper in Appendix D.*

The data described in Section 4.2 is used to model road degradation. As previously states, the inspection data correspond to eight unique damage modes with up to six distinct damage types per road segment, in addition to feature data concerning road construction materials and traffic pattern classifications. This level of detail facilitates the specification of degradation and maintenance effects across various combinations of material and traffic pattern attributes. However, the dataset does not provide sufficient coverage to enable robust modelling of all such combinations.

With regard to construction materials, tiled pavements and asphalt pavements constitute the majority of the network. Specifically, 80.1% of roads in Amsterdam are classified as tiled pavements, and 15.9% as asphalt pavements. All remaining material categories individually represent less than 3% of the road network. While further disaggregation of the transition model (e.g., by traffic pattern classification) is possible, for the sake of brevity, aggregated results for tiled pavements across all traffic pattern classifications and for damage modes ’transverse unevenness’, ’irregularities’ and ’joint width’ are presented here.

Degradation processes are modelled by estimating transition probability matrices derived from observed condition transitions. Figure 4.9 presents the 8-year transition

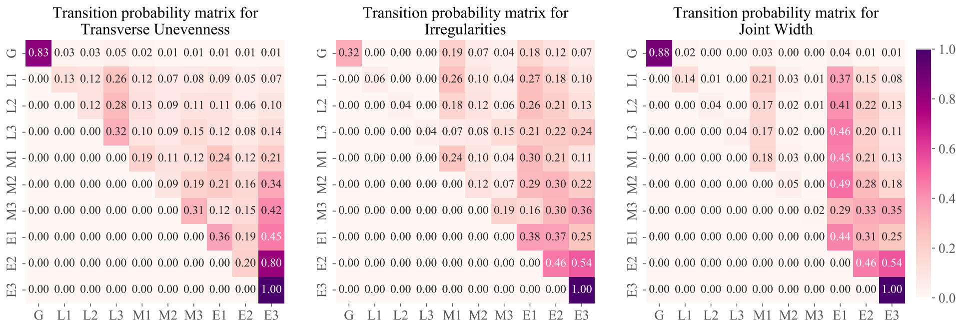


Figure 4.9: Eight-year transition probability matrices for the three relevant damage modes in tiled pavements across the Amsterdam road network: transverse unevenness, irregularities, and joint width. Condition states range from G (good/no damage) to E3 (severe damage affecting a large area), with severity increasing from top to bottom and left to right.

4

probability matrices for the three dominant damage modes observed in tiled pavements within the Amsterdam road network: transverse unevenness, irregularities, and joint width. Each matrix displays the starting states along the vertical axis and the corresponding ending states along the horizontal axis. States are coded as follows: G (good/no damage), $L1 - L3$ (minor damage), $M1 - M3$ (moderate damage), and $E1 - E3$ (serious damage), with numerical indices denoting the spatial extent of damage.

Table 4.2 summarises the total number of recorded transitions for each damage mode, as well as the proportion of observations in which no change in condition occurred ($G \rightarrow G$). High proportions of such stable states, particularly in joint width (94.5%) and transverse unevenness (91.6%), suggest relatively slow deterioration processes for those damage types. In contrast, the proportion of unchanged states is considerably lower for irregularities (38.1%), indicating more dynamic or variable degradation behaviour.

Analysis of the dataset shows that some condition states, particularly those representing more severe or extensive damage, are sparsely populated. As such, the reliability of transition probability estimates from these states, specifically $M3$, $E2$, and $E3$, is limited. Notably, for irregularities, the starting states $L2$ and $L3$ are also especially underrepresented. To isolate degradation processes, transitions reflecting stable or worsening conditions are included in the estimation. The most severe state ($E3$) is treated as an absorbing or terminal state in this model.

The values along the diagonal of each matrix indicate the probability that, over an

Table 4.2: Logged transitions and proportion of 'G' → 'G' transitions by damage mode

Damage Mode	**Number of logged transitions**	**Number of 'G' → 'G' transitions**
Transverse unevenness	73,249	67,087 (91.6%)
Irregularities	60,086	22,878 (38.1%)
Joint width	78,577	74,220 (94.5%)

eight-year period, segments remain in their current condition state. These values are especially high for $G \rightarrow G$ transitions for the damage modes 'transverse unevenness' and 'joint width'. Diagonal values are also higher for states indicating high damage severity ($E1$–$E3$), with an exception for $E2$ in transverse unevenness. For transverse unevenness, however, state transitions $L3 \rightarrow L3$ and $M3 \rightarrow M3$ are also relatively high. Non-zero values in the upper triangle suggest a gradual risk of deterioration. In particular, transitions from any state except G to $M1$ and $E1$ appear relatively likely for 'irregularities', and 'joint width' damage types. This trend is especially pronounced for joint width, suggesting that once deterioration initiates, it escalates more rapidly in this damage mode. Zero entries in the upper triangle of the matrix imply the absence of observed transitions to these states. However, although certain states are rounded down to zero in the matrices presented in Figure 4.9, the associated transition rates indicate existence of small probabilities. Finally, for transverse unevenness and irregularities, some transitions that indicate an increase in damage severity, while damage extent decreases or vice versa, such as $M3 \rightarrow E1$, are found to have slightly lower values compared to other transitions in the same row.

4.8. MAINTENANCE EFFECT MODEL

*This section includes preliminary results of descriptive subproject **S1**: Predictive Infrastructure Condition Model. Results presented here are also included in the paper in Appendix D.* As for the degradation model, results for the maintenance effects are presented here for titled pavements across all traffic pattern classifications and for damage mode 'transverse unevenness', 'irregularities' and 'joint width'.

Maintenance effects were modelled through the estimation of instantaneous transition probability matrices, derived from observed condition transitions that could be directly associated with either minor or major maintenance interventions. The instantaneous transition for each damage mode assumes the following form:

$$P_m = w_1 \cdot P_m^1 + w_2 \cdot P_m^2, \quad w_1 + w_2 = 1, \tag{4.1}$$

where P_m^1 represents perfect maintenance and P_m^2 no maintenance. Therefore, P_m^1 is a transition probability matrix where $p(s' = G \mid s) = 1$ for all $s \in S$. This indicates that, regardless of the current state s, the system deterministically transitions to state G, effectively resetting the system to the best condition. P_m^2 is the identity matrix I over the state space S, implying that no transition occurs and the system remains in its current state.

The weights of these matrices, estimated using observed transitions combined with

Table 4.3: Learned weights for maintenance impacts by damage mode and maintenance type.

Weight	Transverse Unevenness		Irregularities		Joint Width	
	Minor	Major	Minor	Major	Minor	Major
w_1	0.597	0.701	0.284	0.442	0.585	0.766
w_2	0.403	0.299	0.716	0.558	0.415	0.234

a set of prototype matrices based on Eq. 4.1, are represented in Table 4.3. These describe the prototype matrices with the lowest cross-entropy relative to the empirical transition data, as outlined in the methodology section.

As dictated in Eq 4.1, w_1 indicates how often maintenance interventions result in perfect repair (transition to state G) and w_2 indicates no observable change in condition. Results are shown in Table 4.10. Major maintenance consistently yields higher probabilities of perfect repair compared to minor maintenance. When comparing across damage modes, joint width exhibits the highest likelihood of perfect repair, followed closely by transverse unevenness, whereas irregularities show substantially lower perfect repair rates. Additionally, for damage modes 'transverse unevenness' and 'joint width', the probability for perfect repair is always higher than the probability for no change in condition, indicating that maintenance interventions more commonly result in a perfect repair than no repair. For irregularities, maintenance actions have a smaller effect.

4.8.1. APPLICATION

Figure 4.10 illustrates a section of the Amsterdam road infrastructure network, to which the previously developed degradation and maintenance effect models have been applied. These comparative visualisations highlight the varying impact of maintenance strategies on the long-term condition of the network. The initial state of the network is

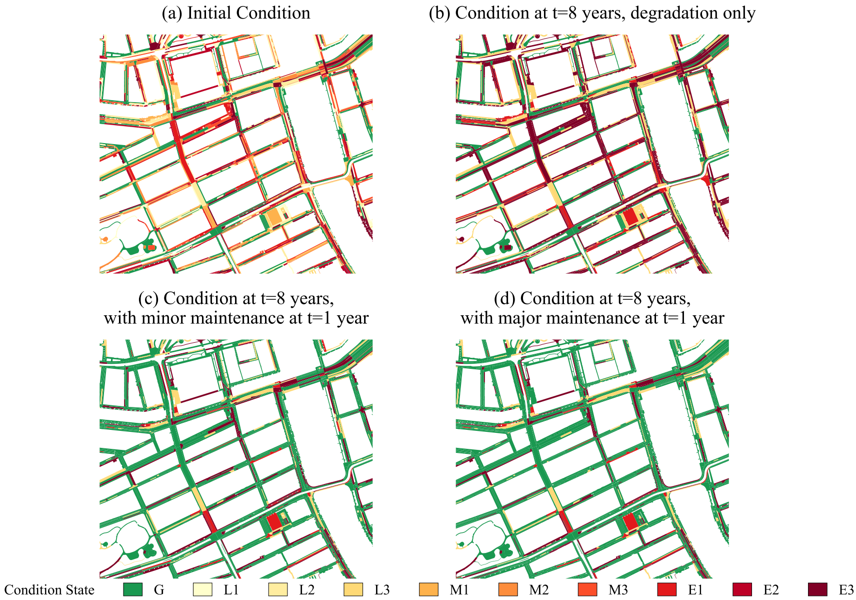


Figure 4.10: Visualisation of the condition evolution of a section of Amsterdam's road network under different maintenance scenarios, assuming tiled pavements for all segments: (a) Initial condition state; (b) Projected condition after eight years of natural degradation without intervention; (c) Condition after eight years with minor maintenance applied uniformly after one year; (d) Condition after eight years with major maintenance applied uniformly after one year. The comparison illustrates the differential impact of maintenance strategies on long-term infrastructure condition.

shown in Figure 4.10a. Figure 4.10b illustrates the condition development over a eight-year period under natural degradation, without any intervention, indicating the condition state as the highest probability in the distribution over states for transverse unevenness. While some segments remain in their initial condition category, others exhibit progressive deterioration, particularly those that began in more severe damage states. Segments initially in perfect or lightly deteriorated states generally show slower rates of degradation. Figure 4.10c shows the same scenario with minor maintenance applied uniformly to all road segments after year one, while Figure 4.10d reflects the scenario in which major maintenance is implemented at the same time point. In both cases, a substantial number of segments are restored to the optimal condition state; however, a subset of road segments shows no improvement following intervention. This proportion is notably higher under the minor maintenance scenario than under major maintenance, indicating the limited effectiveness of minor interventions in reversing deterioration in certain cases.

5

Planning and Practical Aspects

This chapter includes a planning, timeline, and first-year report, as well as other practical aspects, such as an overview of completed and planned Graduate School courses and the Data Management Plan. It also includes a list of potential conferences to attend and journal to publish in.

5.1. Timetable

A timeline of completed and planned events is presented in Figure 5.1. It should be noted that this timeline is indicative. The progression of individual projects will be determined by their specific requirements and the evolving needs of the research. As such, the schedule should be interpreted as a flexible guideline rather than a fixed plan. Moreover, the scope and scale of the various projects differ considerably.

Year 1 included mostly subtasks for subprojects S1, S2, S3 and S4. Therefore, they are primarily preparatory in nature. Most of these contribute to the literature review presented in Chapter 2 and have therefore been completed. Other tasks, such as the data inventory required for S1, S2 and S4, are also completed. These can be found in Chapter 4.

The core modelling activities, represented by the main- and subprojects, are structured in a phased, sequential manner. The overarching project (M), which involves the development of a maintenance strategy generation tool, is designed to integrate and build upon outputs from the preceding models. A preliminary implementation of project M can begin once foundational versions of all subprojects are available. It should be noted that this might reveal limitations with the subprojects, which should then be revisited and re-evaluated until performance is optimal. Initial versions of S1 and S2 have already been developed, as documented in Chapter 4. S1 is also presented in more detail in the publication in Appendix C. Development of the models S3 and S4 will begin in the start of year 2. The integration of all subproject results in M will start alongside the development of these models, and will become the main focus in the latter part of year 3.

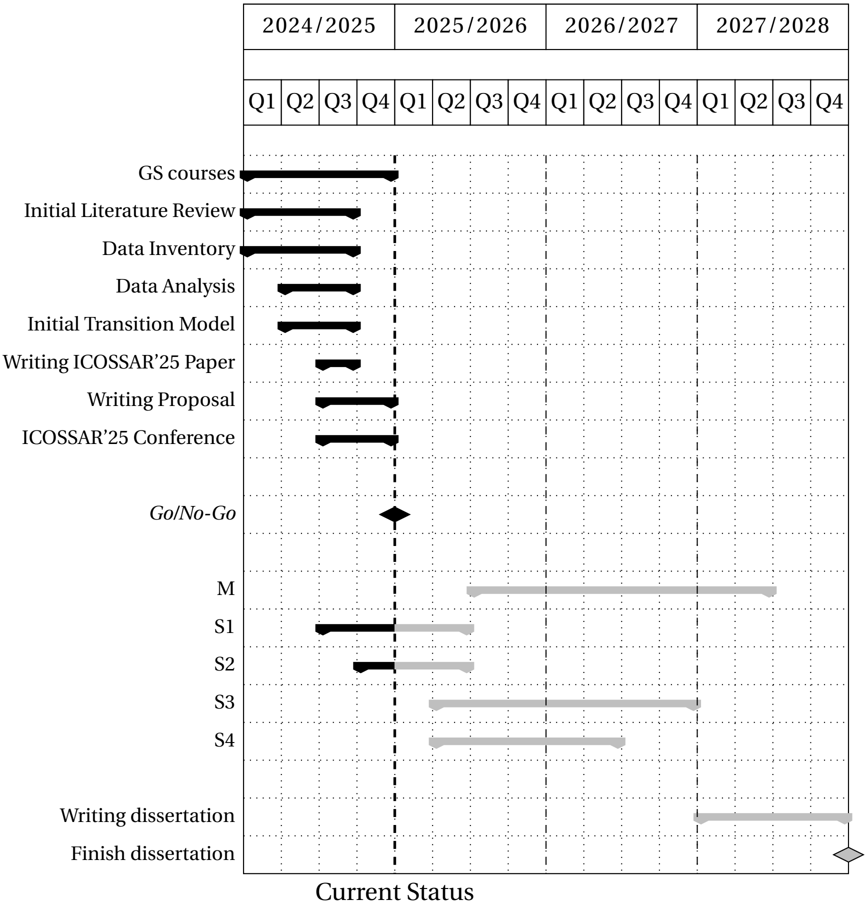


Figure 5.1: Timetable of the PhD Research, per year, per quarter.

5.2. First Year Report

In addition to the tasks outlined in Figure 5.1, the literature review and background presented in Chapter 2 and the preliminary results presented in Chapter 4, the first year of my PhD project included close collaboration with TBL-Infra consortium partners (see Section 3.5), the development of a foundational understanding of current infrastructure management practices in the Netherlands, as well as stakeholder perspectives from all relevant parties.

I worked closely with consortium partners, specifically the Amsterdam Institute for Advanced Metropolitan Solutions (AMS), the City of Amsterdam, Heijmans, Unihorn and Velotech, through regular meetings, interviews, and joint workshops. These engagements enabled mapping of current workflows in road inspection and maintenance planning, highlighting key challenges, available data, and opportunities for innovation. This exchange of knowledge was crucial for aligning my research questions and objectives, as presented in Section 3.1, with the needs and practical realities of infrastructure stakeholders. Additionally, this included a responsibility to answer the *Innovation Tracker* questions defined in the consortium meeting of 26 September 2024. I had full responsibility of answering the following question through the PhD research:

- Which sources of information are required and available to support the development of a predictive model for objects in the built environment?

As of August 2025, the consortium had finalized responses to all *Innovation Tracker* questions. This Innovation Tracker will be concluded in September of 2025.

In parallel, I collaborated with AMS and the municipality of Amsterdam on the development of a Digital Twin Dashboard Prototype, intended to provide an open, easily interpretable interface for infrastructure monitoring, to be used by all parties involved in asset management. My support included attending and helping facilitate interviews with key stakeholders across different departments within the City of Amsterdam and industry partners. This activity not only provided critical user feedback for dashboard development but also deepened my understanding of the design of digital tools to support strategic decision-making.

Within the research group 3D Geoinformation I have taken on the responsibility of organising and hosting weekly presentations, called 3DTea@UDS. Furthermore, I have become a member of the 3D Geoinformation IDEA task force; a group that organises events surrounding relevant social topics in academia, such as social safety. Additionally, I attend the biweekly seminars organised by the AiDAPT Lab research group, where we discuss AI-related research papers and research done by members of the research group.

5.2.1. ICOSSAR25

During my first year, I attended the 14th International Conference on Structural Safety and Reliability (ICOSSAR25). I also had the opportunity to present the first results, as outlined in the paper in Appendix D, of the PhD project at this conference.

The conference was exceptionally well-organised and intellectually stimulating, making it an ideal environment to learn, reflect, and connect. Additionally, it was a pleasure to meet other first-time conference attendees and exchange ideas with fellow researchers. I was genuinely impressed by the work being done across the community. While I attended many presentations, I particularly enjoyed sessions on AI, Graph Neural Networks, and Deep or Multi-Agent Reinforcement Learning, life-cycle performance optimisation, and decision-making under uncertainty. Many of these also related to maintenance planning specifically. Presentations included many different approaches, deepening my knowledge in my own research area while also broadening my perspective on methods I was less familiar with.

5.2.2. Other Events

During the first year of the PhD, I have also spoken about PhD research and related topics at various events. A brief overview of these events is shared here.

***Hoe AI de architectuur verandert (How AI transforms Architecture)* at Pakhuis De Zwijger, Amsterdam**
In November of 2024 I had the opportunity to speak about the effects of AI on the fields and architecture and urbanism in a broadcast by Pakhuis de Zwijger[1]. I had the opportuny to introduce the PhD research during this event. A recording of this broadcast is available on YouTube[2].

PhD Research Interview for the municipality of Amsterdam
This event included a promotional video for the wider consortium project, initiated by the municipality of Amsterdam. This video was published in December of 2024 [3]

***Building the Future* Symposium at Windesheim College, Zwolle**
In March of 2025, I was invited to speak at the Windesheim College in Zwolle, the Netherlands[4]. During this event, I spoke about the technical side of AI and its potential uses within the fields of architecture, urbanism, construction and engineering. I also introduced the PhD project as one an example.

5

***Architect van je eigen werk* event at SFA | Stichting Fonds Architectenbureaus**
In September of 2025, I organised a workshop on the basics of AI for the *Architect van je eigen werk* event, held in Mauritskazerne in Ede, the Netherlands by SFA[5]. The workshop was specifically designed for architects with no prior programming experience and focused on gaining a basic understanding of AI architectures.

5.3. TOOLS AND TECHNICAL ASPECTS

Table 5.1 provides an overview of tools, programming languages and libraries to be used during the PhD Research.

This PhD project will include the development of one main tool and three subtools. All tools developed during this research will be published according to the TBL-Infra consortium agreement. Any data that I am able to share publicly will be uploaded to the TUResearchData repository. A Data Management Plan is included in this report as Appendix B.

5.4. GRADUATE SCHOOL COURSES

The Graduate School requires PhD candidates to obtain 45 Graduate School credits (GSc), comprised of 15 GSc for *Discipline Related Skills*, 15 GSc for *Research Skills* and 15 GSc for *Transferable Skills*. Currently, I have obtained a total of 22.5 GSc, with courses totalling 5 additional GSc currently in progress. An overview of how the remaining GSc are distributed over year 2-4 is shown in Figure 5.2. Figure 5.3 shows a timeline of completed

[1] https://dezwijger.nl/programma/hoe-ai-de-architectuur-verandert
[2] https://www.youtube.com/watch?v=r-AQkftorh8
[3] https://toekomstbestendigeleefomgeving.nl/nieuws/infra-nieuws/amsterdam-voorspelt-onderhoud-met-digitale-informatie/
[4] https://www.windesheim.nl/agenda/2025/maart/symposium-building-the-future?uc=1
[5] https://www.sfa-architecten.nl/geen-categorie/architect-van-je-eigen-werk-2025-2ede-editie-24-september-2025/

Table 5.1: Overview of tools and technical aspects to be used in the PhD Research.

Tool	Purpose
Computing Infrastructure	
TUD HPC clusters, DelftBlue[6]	For computationally intensive simulations and machine learning model training.
Programming Languages	
Python	Primary programming language for data processing, analysis, and modeling.
Key Libraries	
`pandas`, `numpy`, `scipy`	For data manipulation and numerical computations.
`geopandas`	For data manipulation and numerical computations on geospatial data.
`tensorflow`, `pytorch`	For machine learning modelling & training.
`matplotlib`	For visualisation.
Version Control and Collaboration	
Git, GitHub, GitLab	For source code versioning, collaboration & publishing.

and planned courses. Each course, with full course names and GSc value, is provided in Appendix C.

5.5. DATA MANAGEMENT PLAN

The full Data Management Plan (DMP) is included in this report as Appendix B. This DMP has been reviewed by the ABE Data Steward. Data from external partners from the TBL-Infra consortium is used in this project. As a result, data and code developed during this research will be published in accordance with the consortium agreements. Whenever possible, data and code will be published as open-source, through git repositories. Any publicly shareable data will be uploaded to the 4TU.ResearchData repository.

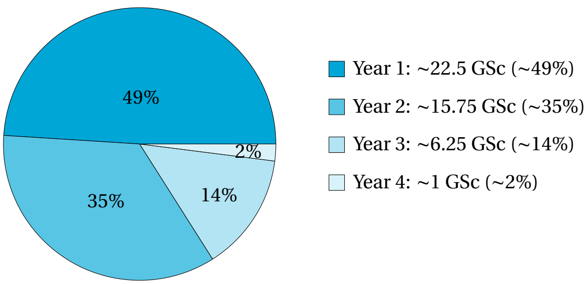


Figure 5.2: Overview of planned GSc's to be obtained per year.

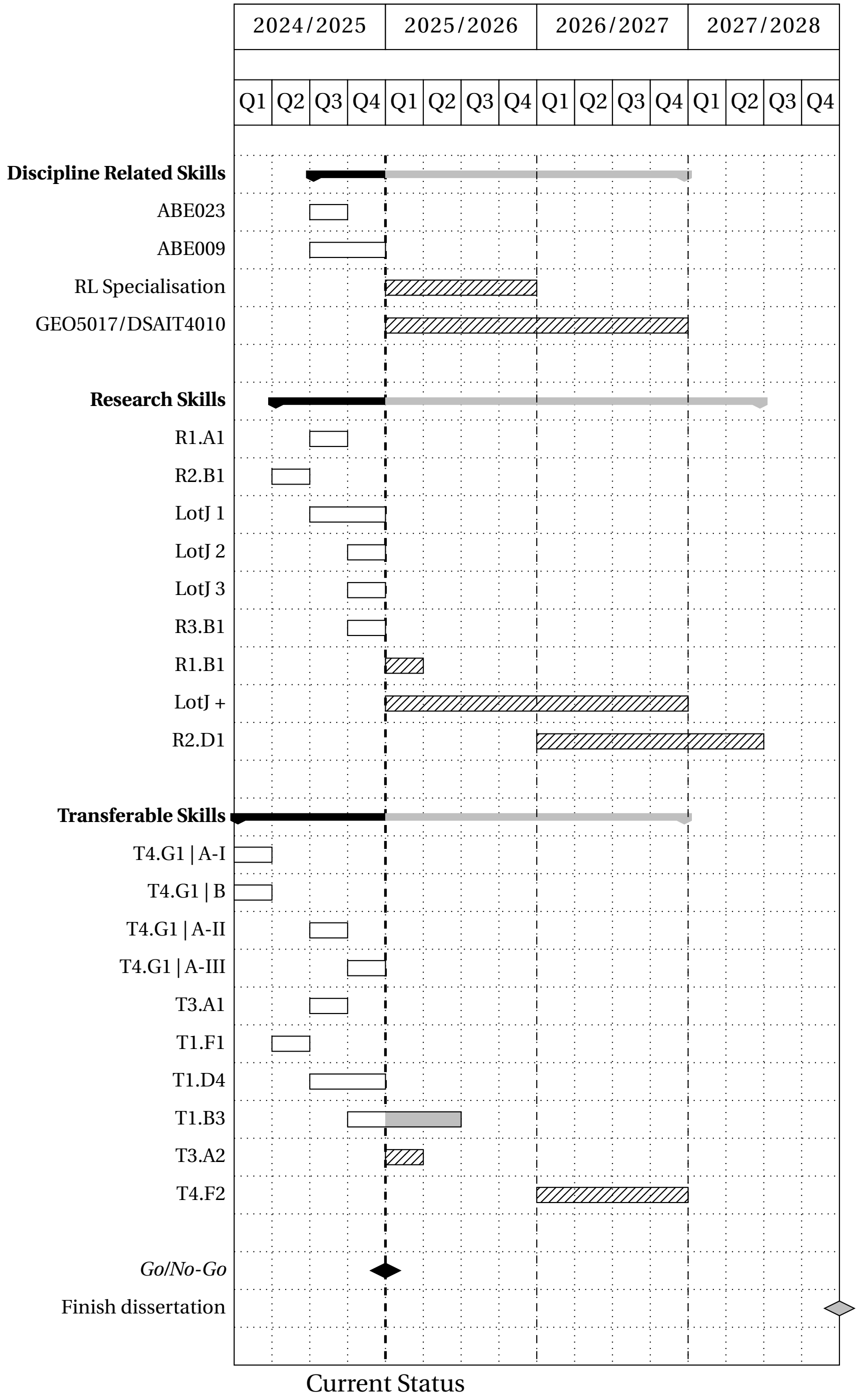


Figure 5.3: Timeline of the completed and planned Graduate School courses, per year, per quarter, starting from September 2024.

5

5.6. PLANNED PUBLICATIONS

Planned publications include conference and journal papers. As the research is multidisciplinary, conferences and journals in multiple fields, including Artificial Intelligence, Machine Learning, Infrastructure Engineering, System Reliability and Geo-information, are considered. I aim to publish one to two papers per year, throughout the 4-year duration of the PhD. This will include both conference and journal papers.

5.6.1. JOURNALS

Journals to be considered include, but are not limited to:

- *Reliability Engineering & System Safety*, IF: 9.4
- *Structural Safety*, IF: 5.7
- *Probabilistic Engineering Mechanics*, IF: 3.0
- *IEEE Transactions on Reliability*, IF: 11.6
- *Structure and Infrastructure Engineering*, IF: 2.6
- *Journal of Infrastructure Systems*, IF: 2.2
- *Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, IF: 2.7
- *Journal of Transportation Engineering, Part A: Systems*, IF: 2.1
- *Computer-Aided Civil and Infrastructure Engineering (CACAIE)*, IF: 8.5
- *Automation in Construction*, IF: 11.5
- *Neural Computing and Applications*, IF: 5.1
- *Journal of Manufacturing Systems*, IF: 12.3

Table 5.2: Mapping of subprojects to relevant journals.

(Sub)project	Reliability Eng. & Sys. Safety	Structural Safety	Probabilistic Eng. Mechanics	Trans. on Reliability	Structure & Infrastructure Eng.	Infrastructure Systems	Risk & Uncertainty in Eng. Systems	Transportation Eng.	Computer-Aided Civil & Infra Eng.	Automation in Construction	Neural Computing & Applications	Manufacturing Systems
Main Project M	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Subproject 1	✓	✓	✓	✓	✓	✓	✓	✓				
Subproject 2					✓	✓		✓	✓			
Subproject 3	✓			✓	✓	✓	✓	✓	✓	✓	✓	
Subproject 4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

The reviewed journals cover complementary aspects of reliability, maintenance, and data-driven decision support in infrastructure systems. Several journals focus on reliability and risk analysis across complex engineering systems, including *Reliability Engineering & System Safety*, *Structural Safety*, *Probabilistic Engineering Mechanics*, and *IEEE Transactions on Reliability*. These journals emphasise risk quantification, reliability modelling, and system safety, often combining theoretical and applied approaches.

Journals specialising in infrastructure management and life-cycle performance include *Structure and Infrastructure Engineering*, *Journal of Infrastructure Systems*, *Journal of Risk and Uncertainty in Engineering Systems, Part A: Civil Engineering*, and *Journal of Transportation Engineering, Part A: Systems*. Their scopes explicitly include maintenance, asset management, and decision support for civil and transportation infrastructure.

A third group of journals focuses on computational and AI-based methods applicable to infrastructure engineering. *Computer-Aided Civil and Infrastructure Engineering (CACAIE)* and *Automation in Construction* publish advances in digital and decision-support tools across the infrastructure life cycle. *Neural Computing and Applications* and *Journal of Manufacturing Systems* address broader developments in artificial intelligence, optimisation, and reliability that are transferable to infrastructure contexts.

Based on these findings and the detailed aims and scopes listed on the journal websites, the relevant subproject outputs and their potential target journals are summarised in Table 5.2.

5.6.2. CONFERENCES

Conferences to be considered include, but are not limited to:

- *International Conference on Structural Safety and Reliability (ICOSSAR)*
- *European Safety and Reliability (ESREL)*
- *International Symposium on Life-Cycle Civil Engineering (IALCCE)*
- *International Symposium on Reliability Engineering and Risk Management (ISR-ERM)*
- *International Conference On Application Of Statistics And Probability In Civil Engineering (ICASP)*
- *Engineering Mechanics Institute Conference and Probabilistic Mechanics & Reliability Conference (EMI/PMC)*
- *International Association for Bridge Maintenance and Safety (IABMAS)*
- *Conference on Neural Information Processing Systems (NeurIPS)*
- *International Conference on Learning Representations (ICLR)*
- *European Conference on Artificial Intelligence (ECAI)*
- *International Conference on Geographic Information Science (GIScience)*
- *Association of Geographic Information Laboratories in Europe Conference (AGILE)*
- *International Conference on Geographical Information Systems Theory, Applications and Management (GISTAM)*

Conferences focusing on reliability, risk, and maintenance of engineering systems include *ICOSSAR*, *ESREL*, *IALCCE*, *ISRERM*, *ICASP*, and *EMI/PMC*. These events regularly feature work on safety, uncertainty modelling, and maintenance decision-making in civil and structural engineering. *IABMAS* focuses specifically on bridge management

Table 5.3: Mapping of subprojects to relevant conferences.

(Sub)project	ICOSSAR	ESREL	IALCCE	ISRERM	ICASP	EMI/PMC	IABMAS	NeurIPS	ICLR	ECAI	GIScience	AGILE	GISTAM
Main Project M	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Subproject 1	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Subproject 2											✓	✓	✓
Subproject 3	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓			
Subproject 4	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

and maintenance and can be considered if bridge assets are included in the research.

The AI and machine learning sector is represented by conferences such as *NeurIPS*, *ICLR*, and *ECAI*, which cover theoretical and applied advances in deep learning, data-driven decision support, and sustainable AI. Such topics directly relevant to intelligent maintenance modelling.

Conferences with a geospatial and computational focus include *GIScience*, *AGILE*, and *GISTAM*. These venues welcome research on geographic information systems, geo-computation, and data-driven modelling for spatially distributed infrastructure.

Based on their thematic scope, potential subproject submissions are indicated in Table 5.3. An overview of the most recent or upcoming editions of these conferences, including their frequency, location, and submission details, is provided in Table 5.4.

Table 5.4: Overview of relevant conferences and their upcoming editions.

Conference	Frequency	Next Edition (Location)	Submission Deadlines
ICOSSAR	Every 4 years	Next TBA	TBA
ESREL	Annual	14–19 Jun 2026 (Portugal)	Abstract: 19 Oct 2025
IALCCE	Biennial	Next TBA	TBA
ISRERM	Biennial	28 Jun–1 Jul 2026 (Japan)	Abstract: 30 Nov 2025
ICASP	Every 4 years	Last: 2023; next expected 2027	TBA
EMI/PMC	Annual	2–5 Jun 2026 (USA)	TBA
IABMAS	Biennial	6–10 Jul 2026 (USA)	TBA
NeurIPS	Annual	Dec 2025 (USA)	TBA
ICLR	Annual	Next TBA	TBA
ECAI	Biennial	25–30 Oct 2025 (Italy)	TBA
GIScience	Biennial	Next TBA	TBA
AGILE	Annual	16–19 Jun 2026 (Estonia)	TBA
GISTAM	Annual	21–23 May 2026 (Spain)	Paper: 5 Jan 2026

Bibliography

[1] Edward B. Barbier. "The concept of sustainable economic development". In: *The Economics of Sustainability* 14.2 (1987), pp. 87–96.

[2] Ramesh Kumar, P Gardoni, and M Sanchez-Silva. "Effect of cumulative seismic damage and corrosion on the life-cycle cost of reinforced concrete bridges". In: *Earthquake Engineering & Structural Dynamics* 38.7 (2009), pp. 887–905.

[3] Leandro Iannacone and Paolo Gardoni. "Stochastic differential equations for the deterioration processes of engineering systems". In: (2019).

[4] Md Saeed Hasan. "Deterioration prediction of concrete bridge components using artificial intelligence and stochastic methods". In: March (2015).

[5] Giovanni Leonardi, Vincenzo Barrile, Rocco Palamara, Federica Suraci, and Gabriele Candela. "Road Degradation Survey Through Images by Drone". In: *New Metropolitan Perspectives*. Ed. by Francesco Calabrò, Lucia Della Spina, and Carmelina Bevilacqua. Cham: Springer International Publishing, 2019, pp. 222–228.

[6] Sharad S Adlinge and Prof a K Gupta. "Pavement Deterioration and its Causes". In: *Mechanical Civil Engineering* (2009), pp. 9–15.

[7] Mohammad Reza Mirzahosseini, Alireza Aghaeifar, Amir Hossein Alavi, Amir Hossein Gandomi, and Reza Seyednour. "Permanent deformation analysis of asphalt mixtures using soft computing techniques". In: *Expert Systems with Applications* 38.5 (2011), pp. 6081–6100.

[8] T Henning and GJ Morrow. *Transition from visual condition rating of cracking , shoving and ravelling to automatic data collection May 2017*. May. 2017.

[9] John S Miller, William Y Bellinger, and Others. *Distress identification manual for the long-term pavement performance program*. Tech. rep. United States. Department of Transportation. Federal Highway Administration~. . ., 2003.

[10] Robert A Wilson. *Basic graph theory*. 2023, pp. 6–18.

[11] Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. "A Comprehensive Survey on Graph Neural Networks". In: *IEEE Transactions on Neural Networks and Learning Systems* 32.1 (2021), pp. 4–24.

[12] Peter W. Battaglia, Jessica B. Hamrick, Victor Bapst, Alvaro Sanchez-Gonzalez, Vinicius Zambaldi, Mateusz Malinowski, Andrea Tacchetti, David Raposo, Adam Santoro, Ryan Faulkner, Caglar Gulcehre, Francis Song, Andrew Ballard, Justin Gilmer, George Dahl, Ashish Vaswani, Kelsey Allen, Charles Nash, Victoria Langston, Chris Dyer, Nicolas Heess, Daan Wierstra, Pushmeet Kohli, Matt Botvinick, Oriol Vinyals, Yujia Li, and Razvan Pascanu. "Relational inductive biases, deep learning, and graph networks". In: (2018), pp. 1–40.

[13] Benjamin Sanchez-Lengeling, Emily Reif, Adam Pearce, and Alexander B Wiltschko. "A gentle introduction to graph neural networks". In: *Distill* 6.9 (2021), e33.

[14] Stefan Irnich, Paolo Toth, and Daniele Vigo. “Chapter 1: The family of vehicle routing problems”. In: *Vehicle Routing: Problems, Methods, and Applications, Second Edition*. SIAM, 2014, pp. 1–33.

[15] Fred L. Hall. “Traffic stream characteristics”. In: *Revised Monograph on Traffic Flow Theory* 165 (1992), pp. 2.1 –2.36.

[16] Renyi Chen and Huaxiong Yao. “Hybrid Graph Models for Traffic Prediction”. In: *Applied Sciences (Switzerland)* 13.15 (2023), pp. 1–17.

[17] Sheldon Ross. *Introduction to Probability Models*. 11th ed. Elsevier, 2014.

[18] Nour Faris, Tarek Zayed, and Ali Fares. “Review of Condition Rating and Deterioration Modeling Approaches for Concrete Bridges”. In: *Buildings* 15.2 (2025).

[19] Yi Jiang and Kumares C Sinha. “Bridge service life prediction model using the Markov chain”. In: *Transp. Res. Rec.* 1223 (1989), pp. 24–30.

[20] Kamalesh Panthi. “A Methodological Framework for Modeling Pavement Maintenance Costs for Projects with Performance-based Contracts”. In: (2009), pp. 1–191.

[21] Ángela Alonso-Solorzano, Heriberto Pérez-Acebo, Daniel J Findley, and Hernán Gonzalo-Orden. “Transition probability matrices for pavement deterioration modelling with variable duty cycle times”. In: *Int. J. Pavement Eng.* 24.2 (2023).

[22] Yasunari Inamura. “Estimating Continuous Time Transition Matrices From Discretely Observed Data”. In: *Bank Japan Work. Pap. Ser.* E07.6 (2006), pp. 1–40.

[23] Ward Whitt. “Continuous-time Markov chains”. In: *Dept. of Industrial Engineering and Operations Research, Columbia University, New York* (2006), p. 164.

[24] Toualith Jean Marc Meango and Mohamed Salah Ouali. “Failure interaction model based on extreme shock and Markov processes”. In: *Reliab. Eng. Syst. Saf.* 197 (2020).

[25] Franciszek Grabski. *Semi-Markov processes: applications in system reliability and maintenance*. Elsevier, 2014.

[26] Franciszek Grabski. “Concept of semi-Markov process”. In: *Zeszyty Naukowe Akademii Marynarki Wojennej* 57 (2016).

[27] Eliuvish Cuicizion. “Markov Renewal Proportional Hazards is All You Need”. In: *arXiv preprint arXiv:2502.03479* (2025).

[28] Lawrence R Rabiner, Chin-Hui Lee, Biing-Hwang Juang, and Jay G Wilpon. “HMM clustering for connected word recognition”. In: *International Conference on Acoustics, Speech, and Signal Processing,* IEEE. 1989, pp. 405–408.

[29] Mariette Awad and Rahul Khanna. “Hidden markov model”. In: *Efficient Learning Machines: Theories, Concepts, and Applications for Engineers and System Designers*. Springer, 2015, pp. 81–104.

[30] Pieter Abbeel. *Markov Decision Processes and Exact Solution Methods*. 2012.

[31] K. G. Papakonstantinou and M. Shinozuka. “Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part I: Theory”. In: *Reliability Engineering and System Safety* 130 (2014), pp. 202–213.

[32] S Yazdanparast, A Sadegheih, M Fallahnezhad, and M Abooie. “Modelling and Decision-making on Deteriorating Production Systems using Stochastic Dynamic Programming Approach”. In: *International Journal of Engineering* 31.12 (2018), pp. 2052–2058.

[33] Yun Bai, Saeed Babanajad, and Zheyong Bian. “Transportation infrastructure asset management modeling using Markov decision process under epistemic uncertainties”. In: *Smart and Resilient Transportation* 3.3 (2021), pp. 249–265.

[34] Hwasoo Yeo, Yoonjin Yoon, and Samer Madanat. “Algorithms for bottom-up maintenance optimisation for heterogeneous infrastructure systems”. In: *Structure and Infrastructure Engineering* 9.4 (2013), pp. 317–328.

[35] Cyrus Derman. “On sequential decisions and Markov chains”. In: *Management Science* 9.1 (1962), pp. 16–24.

[36] Morton Klein. “Inspection—maintenance—replacement schedules under markovian deterioration”. In: *Management Science* 9.1 (1962), pp. 25–32.

[37] Kamal Golabi, Ram B Kulkarni, and George B Way. “A statewide pavement management system”. In: *Interfaces* 12.6 (1982), pp. 5–21.

[38] Kamal Golabi and Richard Shepard. “Pontis: A system for maintenance optimization and improvement of US bridge networks”. In: *Interfaces* 27.1 (1997), pp. 71–88.

[39] Pablo L. Durango-Cohen and Samer M. Madanat. “Optimization of inspection and maintenance decisions for infrastructure facilities under performance model uncertainty: A quasi-Bayes approach”. In: *Transportation Research Part A: Policy and Practice* 42.8 (2008), pp. 1074–1085.

[40] Dayong Wu, Changwei Yuan, Wesley Kumfer, and Hongchao Liu. “A life-cycle optimization model using semi-markov process for highway bridge maintenance”. In: *Applied Mathematical Modelling* 43 (2017), pp. 45–60.

[41] K. G. Papakonstantinou and M. Shinozuka. “Planning structural inspection and maintenance policies via dynamic programming and Markov processes. Part II: POMDP implementation”. In: *Reliability Engineering and System Safety* 130 (2014), pp. 214–224.

[42] Jong Woo Kim, Go Bong Choi, and Jong Min Lee. “A POMDP framework for integrated scheduling of infrastructure maintenance and inspection”. In: *Computers and Chemical Engineering* 112 (2018), pp. 239–252.

[43] R. Ghandali, M. H. Abooie, and M. S.Fallah Nezhad. “A POMDP framework to find optimal policy in sustainable maintenance”. In: *Scientia Iranica* 27.3 E (2020), pp. 1544–1561.

[44] Fatemeh Nazari, Mohamadhossein Noruzoliaee, Bo Zou, and Abolfazl (Kouros) Mohammadian. “Optimal Facility-Specific Inspection and Maintenance Decisions under Measurement Uncertainty: Unifying Framework”. In: *Journal of Infrastructure Systems* 23.4 (2017), pp. 1–10.

[45] Martine van den Boomen, Matthijs T.J. Spaan, Yue Shang, and A. R.M. Wolfert. “Infrastructure maintenance and replacement optimization under multiple uncertainties and managerial flexibility”. In: *Construction Management and Economics* 38.1 (2020), pp. 91–107.

[46] Anna Kalenkova, Lewis Mitchell, and Matthew Roughan. “Performance Analysis: Discovering Semi-Markov Models From Event Logs”. In: February (2025), pp. 1–20.

[47] Sylvie C W Ong, Shao Wei Png, David Hsu, and Wee Sun Lee. “Planning under uncertainty for robotic tasks with mixed observability”. In: *The International Journal of Robotics Research* 29.8 (2010), pp. 1053–1068.

[48] Diederik M Roijers, Peter Vamplew, Shimon Whiteson, and Richard Dazeley. “A survey of multi-objective sequential decision-making”. In: *Journal of Artificial Intelligence Research* 48 (2013), pp. 67–113.

[49] J Matthew Helm, Andrew M Swiergosz, Heather S Haeberle, Jaret M Karnuta, Jonathan L Schaffer, Viktor E Krebs, Andrew I Spitzer, and Prem N Ramkumar. "Machine learning and artificial intelligence: definitions, applications, and future directions". In: *Current reviews in musculoskeletal medicine* 13.1 (2020), pp. 69–76.

[50] Dalvinder Singh Grewal. "A critical conceptual analysis of definitions of artificial intelligence as applicable to computer engineering". In: *IOSR Journal of Computer Engineering* 16.2 (2014), pp. 9–13.

[51] Eduardo F. Morales and Hugo Jair Escalante. *A brief introduction to supervised, unsupervised, and reinforcement learning.* Elsevier Inc., 2021, pp. 111–129.

[52] Pádraig Cunningham, Matthieu Cord, and Sarah Jane Delany. "Supervised learning". In: *Machine learning techniques for multimedia: case studies on organization and retrieval.* Springer, 2008, pp. 21–49.

[53] Richard S Sutton and Andrew G Barto. *Reinforcement learning : an introduction.* English. Cambridge, Massachusetts, 2018.

[54] Marco Wiering and Martijn van Otterlo. *Reinforcement learning : state-of-the-art.* English. Berlin, 2012.

[55] Surjeet Balhara, Nishu Gupta, Ahmed Alkhayyat, Isha Bharti, Rami Q Malik, Sarmad Nozad Mahmood, and Firas Abedi. "A survey on deep reinforcement learning architectures, applications and emerging trends". In: *IET Communications* 19.1 (2025), e12447.

[56] Qingyan Huang. "Model-based or model-free, a review of approaches in reinforcement learning". In: *2020 International Conference on Computing and Data Science (CDS).* IEEE. 2020, pp. 219–221.

[57] Ardi Tampuu, Tambet Matiisen, Dorian Kodelja, Ilya Kuzovkin, Kristjan Korjus, Juhan Aru, Jaan Aru, and Raul Vicente. "Multiagent cooperation and competition with deep reinforcement learning". In: *PloS one* 12.4 (2017), e0172395.

[58] Zhen "Leo" Liu. "Policy-Based Reinforcement Learning". In: *Artificial Intelligence for Engineers: Basics and Implementations.* Springer, 2025, pp. 357–378.

[59] Ronald J Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning". In: *Machine learning* 8.3 (1992), pp. 229–256.

[60] Xin Xu, Lei Zuo, and Zhenhua Huang. "Reinforcement learning algorithms with function approximation: Recent advances and applications". In: *Information sciences* 261 (2014), pp. 1–31.

[61] Gavin A Rummery and Mahesan Niranjan. *On-line Q-learning using connectionist systems.* Vol. 37. University of Cambridge, Department of Engineering Cambridge, UK, 1994.

[62] Christopher J C H Watkins and Peter Dayan. "Q-learning". In: *Machine learning* 8.3 (1992), pp. 279–292.

[63] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, and Others. "Human-level control through deep reinforcement learning". In: *nature* 518.7540 (2015), pp. 529–533.

[64] Deepanshu Mehta. "State-of-the-art reinforcement learning algorithms". In: *International Journal of Engineering Research and Technology* 8.1 (2020), pp. 717–722.

[65] Hado Van Hasselt, Arthur Guez, and David Silver. "Deep reinforcement learning with double q-learning". In: *Proceedings of the AAAI conference on artificial intelligence.* Vol. 30. 1. 2016.

[66] Ziyu Wang, Tom Schaul, Matteo Hessel, Hado Hasselt, Marc Lanctot, and Nando Freitas. “Dueling network architectures for deep reinforcement learning”. In: *International conference on machine learning*. PMLR. 2016, pp. 1995–2003.

[67] Ji He, Jianshu Chen, Xiaodong He, Jianfeng Gao, Lihong Li, Li Deng, and Mari Ostendorf. “Deep reinforcement learning with a natural language action space”. In: *arXiv preprint arXiv:1511.04636* (2015).

[68] Volodymyr Mnih, Adria Puigdomenech Badia, Mehdi Mirza, Alex Graves, Timothy Lillicrap, Tim Harley, David Silver, and Koray Kavukcuoglu. “Asynchronous methods for deep reinforcement learning”. In: *International conference on machine learning*. PmLR. 2016, pp. 1928–1937.

[69] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. “Continuous control with deep reinforcement learning”. In: *arXiv preprint arXiv:1509.02971* (2015).

[70] John Schulman, Sergey Levine, Pieter Abbeel, Michael Jordan, and Philipp Moritz. “Trust region policy optimization”. In: *International conference on machine learning*. PMLR. 2015, pp. 1889–1897.

[71] John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. “Proximal policy optimization algorithms”. In: *arXiv preprint arXiv:1707.06347* (2017).

[72] Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. “Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor”. In: *International conference on machine learning*. Pmlr. 2018, pp. 1861–1870.

[73] Michael L Littman. “Markov games as a framework for multi-agent reinforcement learning”. In: *Machine learning proceedings 1994*. Elsevier, 1994, pp. 157–163.

[74] Sven Gronauer and Klaus Diepold. “Multi-agent deep reinforcement learning: a survey”. In: *Artificial Intelligence Review* 55.2 (2022), pp. 895–943.

[75] Prateek Bhustali and Charalampos P Andriotis. “Assessing the optimality of decentralized inspection and maintenance policies for stochastically degrading engineering systems”. In: *Benelux Conference on Artificial Intelligence*. Springer. 2023, pp. 236–254.

[76] Charalampos P Andriotis and Konstantinos G Papakonstantinou. “Deep reinforcement learning driven inspection and maintenance planning under incomplete information and constraints”. In: *Reliability Engineering & System Safety* 212 (2021), p. 107551.

[77] Pascal Leroy, Pablo G Morato, Jonathan Pisane, Athanasios Kolios, and Damien Ernst. “IMP-MARL: a suite of environments for large-scale infrastructure management planning via MARL”. In: *Advances in neural information processing systems* 36 (2023), pp. 53522–53551.

[78] Zachary Hamida and James-A Goulet. “Hierarchical reinforcement learning for transportation infrastructure maintenance planning”. In: *Reliability Engineering & System Safety* 235 (2023), p. 109214.

[79] Linyi Yao, Qiao Dong, Jiwang Jiang, and Fujian Ni. “Deep reinforcement learning for long-term pavement maintenance planning”. In: *Computer-Aided Civil and Infrastructure Engineering* 35.11 (2020), pp. 1230–1245.

[80] Reza Mohammadi and Qing He. “A deep reinforcement learning approach for rail renewal and maintenance planning”. In: *Reliability Engineering & System Safety* 225 (2022), p. 108615.

[81] Linyi Yao, Zhen Leng, Jiwang Jiang, and Fujian Ni. “A multi-agent reinforcement learning model for maintenance optimization of interdependent highway pavement networks”. In: *Computer-Aided Civil and Infrastructure Engineering* 39.19 (2024), pp. 2951–2970.

[82] Anna Jarosz, Marta Zagorowska, and Jerzy Baranowski. “Recent advances in data-driven methods for degradation modelling across applications”. In: (2025).

[83] Rabi G Mishalani and Samer M Madanat. “Computation of Infrastructure Transition Probabilities Using Stochastic Duration Models”. In: *J. Infrastruct. Syst.* 8.4 (2002), pp. 139–148.

[84] K P George, A S Rajagopal, and L K Lim. “Models for predicting pavement deterioration”. In: *Transp. Res. Rec.* 1215 (1989), pp. 1–7.

[85] Turki I Al-Suleiman and Adnan M S Shiyab. “Prediction of Pavement Remaining Service Life Using Roughness Data - Case Study in Dubai”. In: *Int. J. Pavement Eng.* 4.2 (2003), pp. 121–129.

[86] Nader Abdelaziz, Ragaa T Abd El-Hakim, Sherif M El-Badawy, and Hafez A Afify. “International Roughness Index prediction model for flexible pavements”. In: *Int. J. Pavement Eng.* 21.1 (2020), pp. 88–99.

[87] Saeid Alimoradi, Amir Golroo, and Seyed Mohammad Asgharzadeh. “Development of pavement roughness master curves using Markov Chain”. In: *Int. J. Pavement Eng.* 23.2 (2022), pp. 453–463.

[88] S Madeh Piryonesi and Tamer E El-Diraby. “Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition Index”. In: *J. Infrastruct. Syst.* 26.1 (2020).

[89] Ningyuan Li, Wei Chau Xie, and Ralph Haas. “Reliability-based processing of markov chains for modeling pavement network deterioration”. In: *Transp. Res. Rec.* 1524 (1996), pp. 203–213.

[90] Kiyoshi Kobayashi, Kiyoyuki Kaito, and Nam Lethanh. “A Bayesian Estimation Method to Improve Deterioration Prediction for Infrastructure System with Markov Chain Model”. In: *Int. J. Archit. Eng. Constr.* 1.1 (2012), pp. 1–13.

[91] Jhenyffer Lorrany Matias de Oliveira, Gary Davis, Alireza Khani, and Mihai Marasteanu. “Heterogeneous Markov Chain Model to Predict Pavement Performance and Deterioration”. In: *Transp. Res. Rec.* 2676.9 (2022), pp. 568–581.

[92] Amir Shtayat, Sara Moridpour, Berthold Best, and Shahriar Rumi. “An Overview of Pavement Degradation Prediction Models”. In: *J. Adv. Transp.* 2022 (2022).

[93] Putu Mandiartha, Colin F Duffield, Russell G Thompson, and Marcus R Wigan. “Measuring pavement maintenance effectiveness using Markov Chains analysis”. In: *Struct. Infrastruct. Eng.* 13.7 (2017), pp. 844–854.

[94] Yaning Qiao, Andrew Dawson, Tony Parry, and Gerardo W Flintsch. “Immediate effects of some corrective maintenance interventions on flexible pavements”. In: *Int. J. Pavement Eng.* 19.6 (2018), pp. 502–508.

[95] Samer Madanat. “Optimal infrastructure management decisions under uncertainty”. In: *Transp. Res. Part C* 1.1 (1993), pp. 77–88.

[96] M Saifullah, K G Papakonstantinou, C P Andriotis, and S M Stoffels. “Multi-agent deep reinforcement learning with centralized training and decentralized execution for transportation infrastructure management”. In: i (2024), pp. 1–42.

[97] C P Andriotis and K G Papakonstantinou. “Managing engineering systems with large state and action spaces through deep reinforcement learning”. In: *Reliab. Eng. Syst. Saf.* 191.April (2019), p. 106483.

[98] Konstantinos G Papakonstantinou, Charalampos P Andriotis, and Masanobu Shinozuka. “POMDP and MOMDP solutions for structural life-cycle cost minimization under partial and mixed observability”. In: *Struct. Infrastruct. Eng.* 14.7 (2018), pp. 869–882.

5

[99] Sajid Ali, Tamer Abuhmed, Shaker El-Sappagh, Khan Muhammad, Jose M. Alonso-Moral, Roberto Confalonieri, Riccardo Guidotti, Javier Del Ser, Natalia Díaz-Rodríguez, and Francisco Herrera. “Explainable Artificial Intelligence (XAI): What we know and what is left to attain Trustworthy Artificial Intelligence”. In: *Information Fusion* 99.April (2023), p. 101805.

[100] David A Broniatowski. “Psychological foundations of explainability and interpretability in artificial intelligence”. In: *Natl. Inst. Stand. Technol. Interag. Intern. Rep. 8367* (2021), p. 56.

[101] Muhammad Raza. *Explainable vs. Interpretable Artificial Intelligence*. 2024.

[102] Conor O’Sullivan. “Interpretable vs Explainable Machine Learning”. In: (2023).

[103] Cynthia Rudin. “Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead”. In: *Nature Machine Intelligence* 1.5 (2019), pp. 206–215.

[104] Scott M. Lundberg and Su In Lee. “A unified approach to interpreting model predictions”. In: *Advances in Neural Information Processing Systems* 2017-Decem.Section 2 (2017), pp. 4766–4775.

[105] Idit Cohen. *Explainable AI (XAI) with SHAP - regression problem*. 2021.

[106] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. “" Why should i trust you?" Explaining the predictions of any classifier”. In: *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016, pp. 1135–1144.

[107] Aleix Nieto Juscafresa. “An introduction to explainable artificial intelligence with LIME and SHAP”. In: *Treballs Finals de Grau (TFG) - Matemàtiques* (2022).

[108] Ke Zhang, Peidong Xu, and Jun Zhang. “Explainable AI in deep reinforcement learning models: A shap method applied in power system emergency control”. In: *2020 IEEE 4th conference on energy internet and energy system integration (EI2)*. IEEE. 2020, pp. 711–716.

[109] Jakob Løver, Vilde B Gjærum, and Anastasios M Lekkas. “Explainable AI methods on a deep reinforcement learning agent for automatic docking”. In: *IFAC-PapersOnLine* 54.16 (2021), pp. 146–152.

[110] Sindre Benjamin Remman, Inga Strümke, and Anastasios M Lekkas. “Causal versus marginal shapley values for robotic lever manipulation controlled using deep reinforcement learning”. In: *2022 American Control Conference (ACC)*. IEEE. 2022, pp. 2683–2690.

[111] Mohammad Hossein Nejati Amiri, Fawaz Annaz, Mario De Oliveira, and Florimond Gueniat. “Deep Reinforcement Learning with Local Interpretability for Transparent Microgrid Resilience Energy Management”. In: *arXiv preprint arXiv:2508.08132* (2025).

[112] Ke Zhang, Jun Zhang, Pei-Dong Xu, Tianlu Gao, and David Wenzhong Gao. “Explainable AI in deep reinforcement learning models for power system emergency control”. In: *IEEE Transactions on Computational Social Systems* 9.2 (2021), pp. 419–427.

[113] Daniel Beechey, Thomas M S Smith, and Özgür. “Explaining reinforcement learning with shapley values”. In: *International Conference on Machine Learning*. PMLR. 2023, pp. 2003–2014.

[114] Ziyang Lu, M Cenk Gursoy, Chilukuri K Mohan, and Pramod K Varshney. “Explainable AI for Radar Resource Management: Modified LIME in Deep Reinforcement Learning”. In: *arXiv preprint arXiv:2506.20916* (2025).

[115] Juan Marcelo Parra-Ullauri, Antonio Garc´-Dom´, Nelly Bencomo, Changgang Zheng, Chen Zhen, Juan Boubeta-Puig, Guadalupe Ortiz, and Shufan Yang. “Event-driven temporal models for explanations-ETeMoX: explaining reinforcement learning”. In: *Software and Systems Modeling* 21.3 (2022), pp. 1091–1113.

[116] Abhinav Verma, Vijayaraghavan Murali, Rishabh Singh, Pushmeet Kohli, and Swarat Chaudhuri. "Programmatically interpretable reinforcement learning". In: *International conference on machine learning*. PMLR. 2018, pp. 5045–5054.

[117] Daoming Lyu, Fangkai Yang, Bo Liu, and Steven Gustafson. "SDRL: interpretable and data-efficient deep reinforcement learning leveraging symbolic planning". In: *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 33. 01. 2019, pp. 2970–2977.

[118] Lingfei Wu, Peng Cui, Jian Pei, Liang Zhao, and Le Song. "Graph Neural Networks". In: *Graph Neural Networks: Foundations, Frontiers, and Applications* (2022), pp. 27–37.

[119] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. "Neural message passing for quantum chemistry". In: *34th International Conference on Machine Learning, ICML 2017* 3 (2017), pp. 2053–2070.

[120] Azzedine Boukerche and Jiahao Wang. "Machine Learning-based traffic prediction models for Intelligent Transportation Systems". In: *Computer Networks* 181.May (2020), p. 107530.

[121] Yun Ge, Jian F. Zhai, and Pei C. Su. "Traffic Flow Prediction Based on Multi-Spatiotemporal Attention Gated Graph Convolution Network". In: *Journal of Advanced Transportation* 2022 (2022).

[122] Mohammed Khairy, Hoda M.O. Mokhtar, and Mohammed Abdalla. "Adaptive traffic prediction model using Graph Neural Networks optimized by reinforcement learning". In: *International Journal of Cognitive Computing in Engineering* 6.February (2025), pp. 431–440.

[123] Giacomo Arcieri, Gregory Duthé, Christophe Muller, David Haener, Konstantinos G Papakonstantinou, Daniel Straub, and Eleni Chatzi. "Graph-based multi-agent reinforcement learning for railway infrastructure decision support". In: *14th International Conference on Structural Safety and Reliability-ICOSSAR'25*. Scipedia. 2025.

[124] Keyulu Xu, Stefanie Jegelka, Weihua Hu, and Jure Leskovec. "How powerful are graph neural networks?" In: *7th International Conference on Learning Representations, ICLR 2019* (2019), pp. 1–17.

[125] Rex Ying, Christopher Morris, William L. Hamilton, Jiaxuan You, Xiang Ren, and Jure Leskovec. "Hierarchical graph representation learning with differentiable pooling". In: *Advances in Neural Information Processing Systems* 2018-Decem (2018), pp. 4800–4810.

[126] Junhyun Lee, Inyeop Lee, and Jaewoo Kang. "Self-attention graph pooling". In: *36th International Conference on Machine Learning, ICML 2019* 2019-June (2019), pp. 6661–6670.

[127] Yifan Zhou, Bangcheng Li, and Tian Ran Lin. "Maintenance optimisation of multicomponent systems using hierarchical coordinated reinforcement learning". In: *Reliability Engineering & System Safety* 217 (2022), p. 108078.

[128] Andrew Y Ng, Stuart Russell, and Others. "Algorithms for inverse reinforcement learning." In: *Icml*. Vol. 1. 2. 2000, p. 2.

[129] Sanjay Krishnan, Animesh Garg, Richard Liaw, Lauren Miller, Florian T Pokorny, and Ken Goldberg. "Hirl: Hierarchical inverse reinforcement learning for long-horizon tasks with delayed rewards". In: *arXiv preprint arXiv:1604.06508* (2016).

[130] Pieter Abbeel and Andrew Y Ng. "Apprenticeship learning via inverse reinforcement learning". In: *Proceedings of the twenty-first international conference on Machine learning*. 2004, p. 1.

[131] Jonathan Ho and Stefano Ermon. "Generative adversarial imitation learning". In: *Advances in neural information processing systems* 29 (2016).

[132] Justin Fu, Katie Luo, and Sergey Levine. "Learning robust rewards with adversarial inverse reinforcement learning". In: *arXiv preprint arXiv:1710.11248* (2017).

[133] Jiayu Chen, Tian Lan, and Vaneet Aggarwal. "Hierarchical Adversarial Inverse Reinforcement Learning". In: *IEEE Transactions on Neural Networks and Learning Systems* 35.12 (2024), pp. 17549–17558.

[134] Jiayu Chen, Dipesh Tamboli, Tian Lan, and Vaneet Aggarwal. "Multi-task hierarchical adversarial inverse reinforcement learning". In: *International Conference on Machine Learning*. PMLR. 2023, pp. 4895–4920.

[135] Tomasz Nowakowski and Sylwia Werbińka. "On problems of multicomponent system maintenance modelling". In: *International Journal of Automation and Computing* 6.4 (2009), pp. 364–378.

[136] Amitkumar Patil, Gunjan Soni, Anuj Prakash, and Kritika Karwasra. "Maintenance strategy selection: a comprehensive review of current paradigms and solution approaches". In: *International Journal of Quality and Reliability Management* 39.3 (2022), pp. 675–703.

[137] Jingyi Zhao, Chunhai Gao, and Tao Tang. "A Review of Sustainable Maintenance Strategies for Single Component and Multicomponent Equipment". In: *Sustainability (Switzerland)* 14.5 (2022), pp. 1–22.

[138] Pascal Vrignat, Frédéric Kratz, and Manuel Avila. "Sustainable manufacturing, maintenance policies, prognostics and health management: A literature review". In: *Reliability Engineering and System Safety* 218 (2022).

[139] Mohammad Yazdi. *Springer Series in Reliability Engineering Mohammad Yazdi Advances in Computational Mathematics for Industrial System Reliability and Springer Series in Reliability Engineering*. 2024.

[140] Yousof Gholipour, Mohsen Zare, Majid vaziri sereshk, and Yasser Gholipour. "A Comprehensive Review of Maintenance Strategies: From Reactive to Proactive Approaches". In: *Central Asia and the Caucasus* 26.1 (2025), pp. 70–83.

[141] Antero Ollila and Markku Malmipuro. "Maintenance has a role in quality". In: *The TQM magazine* 11.1 (1999), pp. 17–21.

[142] Paulina Gackowiec. "General overview of maintenance strategies – concepts and approaches". In: *Multidisciplinary Aspects of Production Engineering* 2.1 (2019), pp. 126–139.

[143] Elizabeth Voss. *The Essential Guide to Choosing a Maintenance Strategy for Your Assets*. https://fiixsoftware.com/blog/essential-guide-to-comparing-types-of-maintenance-strategies/. 2023.

[144] Bram de Jonge, Ruud Teunter, and Tiedo Tinga. "The influence of practical factors on the benefits of condition-based maintenance over time-based maintenance". In: *Reliability Engineering and System Safety* 158 (2017), pp. 21–30.

[145] Hongzhou Wang. "A survey of maintenance policies of deteriorating systems". In: *European Journal of Operational Research* 139.3 (2002), pp. 469–489.

[146] Jong Ho Shin and Hong Bae Jun. "On condition based maintenance policy". In: *Journal of Computational Design and Engineering* 2.2 (2015), pp. 119–127.

[147] Rommert Dekker and Cyp van Rijn. "PROMPT, a decision support system for opportunity-based preventive maintenance". In: *Reliability and Maintenance of Complex Systems*. Springer, 1996, pp. 530–549.

[148] Wentai Zhang, E. Haihong, Haoran Luo, and Mingzhi Sun. “FulBM: Fast Fully Batch Maintenance for Landmark-based 3-hop Cover Labeling”. In: *ACM Transactions on Knowledge Discovery from Data* 18.6 (2024).

[149] Mingxin Li, Xiaoli Jiang, and Rudy R. Negenborn. “Opportunistic maintenance for offshore wind farms with multiple-component age-based preventive dispatch”. In: *Ocean Engineering* 231 (2021).

[150] Hasnida Ab-Samat and Shahrul Kamaruddin. “Opportunistic maintenance (OM) as a new advancement in maintenance approaches: A review”. In: *Journal of Quality in Maintenance Engineering* 20.2 (2014), pp. 98–121.

[151] Adriaan Van Horenbeek and Liliane Pintelon. “A dynamic predictive maintenance policy for complex multi-component systems”. In: *Reliability Engineering and System Safety* 120 (2013), pp. 39–50.

[152] J. I. Aizpurua, V. M. Catterson, Y. Papadopoulos, F. Chiacchio, and D. D’Urso. “Supporting group maintenance through prognostics-enhanced dynamic dependability prediction”. In: *Reliability Engineering and System Safety* 168.April (2017), pp. 171–188.

[153] Mayank Pandey, Ming J. Zuo, and Ramin Moghaddass. “Selective maintenance scheduling over a finite planning horizon”. In: *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* 230.2 (2016), pp. 162–177.

[154] Nyoman Gunantara. “A review of multi-objective optimization: Methods and its applications”. In: *Cogent Engineering* 5.1 (2018), pp. 1–16.

A

APPENDIX A

The PhD agreement was defined in the first three months of the PhD trajectory. It includes arrangements with the supervisory team and defined goals for the development of competences and skills.

PhD Agreement

Full name doctoral candidate:	TUD ID Number:
Amy (A.G.E.) Sterrenberg	934895

Research Information	
Working title dissertation:	Data- and AI-driven Transportation Infrastructure Life-Cycle Extension
Start date:	01-09-2024
Research description, publication goals and planning (you can upload an attachment):	The research integrates geoinformation, digital twins, AI, and algorithmic decision-making to develop optimal strategies for life-cycle extension of infrastructure for various performance indicators, including sustainable goals. This work intends to employ Deep Reinforcement Learning (DRL) for generating inspection and maintenance strategies at real scales, with emphasis on the transportation network of Amsterdam.
Attachment:	N/A

Arrangements with supervisory team

Please describe the way in which supervision will take place and how the supervision hours (between 100-200 hours) will be divided among the supervisors:

A weekly meeting, either at the Faculty of Architecture or online, of circa 45-60 minutes is scheduled with Charalampos Andriotis (copromoter, daily supervisor). Once a month, Jantien Stoter (promoter) and, if available, others involved in the project will join this meeting (Lukas Beuster and Ken Arroyo Ohori).

As a result, there will be roughly 52x45 min = 39-52 hours of weekly meeting time per year, of which the promoter (and others) will join 12x45 min = 9-12 hours.

If the weekly meetings can not take place due to illness, leave, etc., they may be skipped. Monthly meetings will preferably be rescheduled in such cases.

Additional one-on-one meetings with team members will be arranged as needed to provide specific help or support between scheduled sessions.

Other arrangements

Data Management Plan - **MANDATORY** (e.g. training on Data Management, consultation of faculty data steward, creating a DMP with DMPonline, discussion about DMP with supervisors):

To ensure proper training on Data Management, Amy is enrolled for the spring 2025 edition of the ABE 023 Research Data Management course. Additionally, the 'Online info session on publishing requirements for data and code' has been attended. A consultation with the faculty data steward will be requested if necessary.

Educational activities (i.e. teaching, supervising, assisting in courses etc.) Specify tasks and allocated time, max 10-15 percent of your time (only mandatory for doctoral candidates employed by TU Delft):

Educational activities to be decided. This may include MSc student group supervision, potentially to benefit the PhD research (i.e. based on proposed topics relevant for this research). Education (in the form of Graduate School courses) is planned to develop teaching/supervising skills.

Other arrangements (i.e. conference visits, budget, etc):

Several conferences on probabilistic modeling in engineering systems, machine learning and infrastructure lifecycle decision making are considered, such as ICASP, ESREL, ICLR, NeurIPS, ECAI, IALCCE, IABMAS, ECCOMAS, UNCECOMP. Geo-information related conferences that could be of interest are GIScience and AGILE.

Development of competences and skills

The doctoral candidate briefly identifies his or her performance in relation to these competences and last year's agreements and indicates where and how improvements can be made. Discuss each competence during the meeting.

The supervisor(s) give(s) feedback and indicate(s) how improvements can be made. Record any new agreement on the last page of this form.

Definition of skills' levels

1 = needs further development; 2 = at requested professional level; 3 = exceeds requested professional level

Competences	Level	Goals and actions
Discipline related skills		
D.1 Scientific Knowledge	1 ☐ 2 ☑ 3 ☐	To develop skills on AI/ML, in particular DRL. For this, GEO5017: Machine Learning for the Built Environment and DSAIT4010: Probabilistic AI and Reasoning are considered (1st/2nd year). Additionally, an online course on Reinforcement Learning is planned.
D.2 Engineering & Design	1 ☐ 2 ☑ 3 ☐	Supervision in early stages is preferred to effectively plan the research in collaboration with stakeholders & to ensure practical relevance. ABE009: Research Proposal for Architecture & the Built Environment planned to further support such discipline related research/engineering skills. (1st year).
Research skills		
R.1 Research Management Designing, Project management, Problem solving, Valorization	1 ☐ 2 ☑ 3 ☐	Planning to follow R1.A1 | Research Design (1st year) for further development of research design & management skill. Also planning to follow R1.B1 | Navigating Academic Publishing (2nd/3rd/4th year).
R.2 Academic Thinking Conceptual thinking, Analytical thinking, Synthetic skills, Critical thinking, Creativity & Innovation	1 ☐ 2 ☑ 3 ☐	Improvement is mainly needed on effective reading of academic literature, as a result, R2.B1 | Speedreading and Mindmapping (1st/2nd year) will be followed. At a later stage R2.D1 How to Formulate Successful Propositions for your PhD Defence will be followed (in 3rd/4th year).
R.3 Academic Attitude Societal context, Ethics	1 ☐ 2 ☑ 3 ☐	Planning to follow R3.B1 | Engineering Ethics to identify ethical challenges of the research (1st/2nd/3rd year).
Skills development by learning-on-the-job activities (LOJ)	1 ☑ 2 ☐ 3 ☐	Highly motivated to join learning-on-the-job activities. Particularly interested in addressing audiences, writing papers, supervising MSc student(s) and join peer-review meetings (all years).
Transferable skills		
T.1 Effective communication Presenting, Writing skills, Story telling, Language skills	1 ☐ 2 ☑ 3 ☐	To improve presenting and writing. Planning to follow a suggested English writing course based on a planned 'placement test', T1.F1 | LinkedIn for Researchers (1st/2nd/3rd year) and T1.B2 | Writing a Dissertation (3rd/4th year).
T.2 Working with others Networking, Collaboration, Negotiation, Leadership	1 ☐ 2 ☑ 3 ☐	To improve networking skills by following T1.F1 | LinkedIn for Researchers (1st/2nd/3rd year). Further improve collaboration skills by attending events organized by research groups (on the job learning activities)
T.3 Teaching, Supervising & Coaching Teaching, Supervising students / Coaching	1 ☑ 2 ☐ 3 ☐	Planning to take T3.A1 | Foundations of Educational Design (1st/2nd year) and one or more of the follow-up courses. Highly motivated to take on responsibilities in education.
T.4 Self-management Autonomy, Time management, Flexibility, Perseverance, Dealing with risk and uncertainty, Entrepreneurship, Personal development	1 ☐ 2 ☐ 3 ☑	Developed this skill previously in both a private and academic setting. Planning to follow T4.F2 | Career Development - Making Impact with your Thesis (3rd/4th year).

☑ Doctoral Education planning in DMA or attached to this form has been seen and approved by supervisors and doctoral candidate

	Doctoral candidate:	Intended promotor(s):	Daily supervisor(s):
Name:	Amy Sterrenberg	Jantien Stoter	Charalampos Andriotis
Date:	25-11-2024	10-12-2024	11-12-2024
Signature:	Amy Sterrenberg — Digitally signed by Amy Sterrenberg Date: 2024.11.25 17:09:10 +01'00'	Jantien Stoter — Digitally signed by Jantien Stoter Date: 2024.12.10 10:13:56 +01'00'	Digitally signed by Charalampos Andriotis Date: 2024.12.11 15:08:42 +01'00'

Please fill in this form and email it to your Faculty Graduate School within 3 months after start date, as you cannot upload the form in DMA yourself.

B

Appendix B

The following pages contain the Data Managment Plan, following the TU Delft Data Management Plan template (2025). This DMP has been reviewed by Architecture and the Built Environment Data Steward Janine Strandberg on 21 May 2025.

B

Plan Overview

A Data Management Plan created using DMPonline

Title: Urban Data Science PhD: Data- and AI-driven Infrastructure Life-cycle Exension

Creator: Amy Sterrenberg

Affiliation: Delft University of Technology

Template: TU Delft Data Management Plan template (2025)

Project abstract:

This PhD project, conducted within the 3D Geoinformation Group and AiDAPT Lab, focuses on AI-driven inspection and maintenance planning for infrastructure, particularly within the City of Amsterdam. The research integrates geoinformation, digital twins, and algorithmic decision-making to develop optimal strategies for extending the life of aging infrastructure, such as roads and bridges. By utilizing geo-data, AI, and engineering models, the goal is to create predictive models and optimize maintenance interventions, contributing to sustainability and infrastructure resilience. This work is carried out in the context of the National Growth Fund project “Future proof living environment”, in collaboration with the City of Amsterdam and the Amsterdam Institute of Advanced Metropolitan Solutions.

ID: 170518

Start date: 01-09-2024

End date: 01-09-2028

Last modified: 14-10-2025

B

Urban Data Science PhD: Data- and AI-driven Infrastructure Life-cycle Exension

0. Adminstrative questions

I have consulted the Architecture and the Built Environment Data Steward Janine Strandberg to review the first full draft of this Data Management Plan on 21 May 2025. I received this feedback on 16 June 2025.

Notes on (temporarily) cancelled studies

I have consulted the Architecture and the Built Environment Contract Manager Ernst Harting on the Data Use Agreement with Heijmans regarding Schiphol Infrastructure Inspection Data and recieved advice on 28 January 2025. This data is currently not in use for the PhD project.

I have consulted the Architecture and the Built Environment Data Steward Janine Strandberg on Ethical Approval for a potential Survey Study among Dutch Municipalities and recieved advice on 26 February 2024. This study has now been canceled.

- No – please provide details of the lead institution below and TU Delft's role in the project

In this project, funded by Nationaal Groeifonds: Toekomstbestendige Leefomgeving Infra, TU Delft is leading the research design and execution. Amsterdam Institure for Advanced Metropolitan Solutions (AMS) is leading the consortium organisation. For the purposes of this project, the City of Amsterdam, Heijmans, Unihorn and Velotech are members of the consortium and share commercial data on road infrastructure, including road feature data, inspection records and maintenance records. This research uses the infrastructure road network as a case study. This PhD is part of 'Project 1: Datagedreven Beheer & Onderhoud'. Within this project, TU Delft is responsible for work packages WP1: Constructieve beoordeling and WP11: Levensduur verlengende maatregelen.

I. Data/code description and collection or re-use

Type of data/code	**File format(s)**	**How will data/code be collected/generated?** *For re-used data/code: what are the sources and terms of use?*	**Purpose of processing**	**Storage location**	**Who will have access to the data/code?**

B

Amsterdam Overview & Features Pavement Infrastructure	.GeoJSON	Re-use of existing data from the City of Amsterdam (data available under consortium agreement). Data were shared through e-mail.	To understand features of roads in the Amsterdam infrastructure network that could impact degradation and/or inspection and maintenance planning.	Project Data Storage (U:)	PhD Team, consisting of: Amy Sterrenberg (PhD Candidate), Jantien Stoter (Promotor) and Charalampos Andriotis (Co-promotor, dialy supervisor). City of Amsterdam (original owner, cannot acces Project Data Storage (U:))
Amsterdam Pavement Infrastructure Inspection Records 2015-2024	.csv	Re-use of existing data from the City of Amsterdam (data available under consortium agreement). Data were shared through e-mail.	To understand and model the degradation process of roads in the Amsterdam infrastructure network.	Project Data Storage (U:)	PhD Team as defined above. City of Amsterdam (original owner, cannot acces Project Data Storage (U:))
Amsterdam Pavement Infrastructure Maintenance Records: KO Maintenance 2019-2024 (seperate annual files)	.gpkg	Re-use of existing data from the City of Amsterdam (data available under consortium agreement)	To understand and model the degradation process of and maintenance effects for roads in the Amsterdam infrastructure network.	Project Data Storage (U:) (primary) Sharepoint, owned by City of Amsterdam (secondary)	PhD Team as defined above. City of Amsterdam (original owner, cannot acces Project Data Storage (U:))

Amsterdam Pavement Infrastructure Maintenance Records: GOVV Maintenance 2019-2024	.shp (including .cpg, .dbf, .prj, .shx)	Re-use of existing data from the City of Amsterdam (data available under consortium agreement)	To understand and model the degradation process of and maintenance effects for roads in the Amsterdam infrastructure network.	Project Data Storage (U:) (to be added asap)	PhD Team as defined above. City of Amsterdam (original owner, cannot acces Project Data Storage (U:))
Data Preposessing Code	.py	Part of a package to be developed during the PhD. The licence under which it will be released t.b.d.	To prepare code for use in models specified below.	Project Data Storage (U:) (primary) GitHub/GitLab once completed.	PhD Team as defined above.
Infrastructure Model Code	.py	Part of a package to be developed during the PhD. The licence under which it will be released t.b.d.	To model infrastructure geometry and interconnectivities for the Amsterdam infrastructure network, and other (synthetic) networks.	Project Data Storage (U:) (primary) GitHub/GitLab once completed.	PhD Team as defined above.
Transition Model Code	.py	Part of a package to be developed during the PhD. The licence under which it will be released t.b.d.	To model the degradation process of and maintenance effects for roads in the Amsterdam infrastructure network, and other (synthetic) networks.	Project Data Storage (U:) (primary) GitHub/GitLab once completed.	PhD Team as defined above.
Reward Model Code	.py	Part of a package to be developed during the PhD. The licence under which it will be released t.b.d.	To compute a measure of quality per (generated) maintenance strategy. This is independent of the Amsterdam data.	Project Data Storage (U:) (primary) GitHub/GitLab once completed.	PhD Team as defined above.
Prescriptive Maintenance Model Code (AI*)	.py	Part of a package to be developed during the PhD. The licence under which it will be released t.b.d.	To generate maintenance strategies based on the infrastructure model, transition model and reward model, for the Amsterdam network, and synthetic networks.	Project Data Storage (U:) (primary) GitHub/GitLab once completed.	PhD Team as defined above.

* The AI model mentioned here will be a Reinforcement Learning model. This model does **not** directly

use data for its training. Rather, it uses trial-and-error learning. Additionally, the research foccusess directly on model interpretability, aiming to lift concerns regarding the 'black-box' nature of AI models. The AI Act, AI at TU Delft and European Approach to AI will be consulted during development.
** The data Schiphol (as mentioned in Q1), is not included here. However, an agreement was signed for use (see Q10). If we do use this data, the DMP will be updated accordingly.

II. Storage and backup during the research process

- 250 GB – 5 TB

- TU Delft OneDrive
- Another storage system – please explain below, including provided security measures
- GitHub/other version control repository (external) – please explain below
- Project Data Storage (U:) drive at TU Delft

Project Data Storage (U:) is used for all data. This ensures both the PhD candidate and their supervisors can access data at any time. The TU Delft OneDrive is used in a more temporary way; mostly for reviewing publications. Github will be used to save code. The other storage systems is Sharepoint, owned by City of Amsterdam. This is used for commercial data only. It can be accessed by all consortium partners. Security is organised by the City of Amsterdam.

III. Data/code documentation

- Procedure – A description of data processing procedure(s) (such as laboratory setup, simulation workflows).
- Data – Data dictionary explaining the variables used
- Metadata – I will adhere to the metadata standards used by the data repository where the data will be shared (see section V)
- Software – Usage documentation (README file, docstrings, and in-line comments)
- Data – README file or other documentation explaining how data are organised
- Data – Methodology of data collection

Documentation will be developed for all code, but most importantly, for the final deliverable (Prescriptive Maintenance Model Code (AI) in Q3). This will include in-line comments, a README file, and related publications.

IV. Legal and ethical requirements, code of conducts

- No

- Yes, confidential data received from commercial, or other external partners

Data from commercial or other external partners include feature, inspection maintenance data for paved infrastructure and are provided to the research by consortium parnters.

The intellectual property rights are framed by a collaboration agreement between Delft University of Technology and AMS and includes the previously named consortium partners. The developed code, including re-use of data from the municipality of Amsterdam falls under this agreement, and can therefore not be shared openly. However, this data can be replaced with synthetic data, enabeling open accessibility of this work. No other restrictions on the re-used data exist. A separate agreement was signed by the TUD and Heijmans, for the management of company data issued by Heijmans. However, the data this argeement refers to is no longer intended to be used, and therefore not included in Q3.

V. Data sharing and long term preservation

- Not all data/code can be publicly shared – please explain below which data/code and the reason why public sharing is not possible

The datasets that are used in the published papers and are owned by other parties will not be made available on publication. Any synthetic data will be shared. Re-used third-party data is stored for the duration of this PhD.

- The data/code can't be shared in a data repository, but the metadata will be registered in 4TU.ResearchData with a persistent identifier (a DOI), and all research publications resulting from the project have a statement explaining: what additional datasets/materials exist, why access is restricted, who can use the data and under what circumstances
- I will share the code via git(lab)/subversion and also create a snapshot in a data repository
- All data/code will be uploaded to 4TU.ResearchData

Any data that can be shared will be uploaded to 4TU.ResearchData, this includes all developed code, and . For data that cannot be shared, metadata can be registered in 4TU.ResearchData. All research publications resulting from the project using this unsharable data will have a statement explaining: what additional datasets/materials exist, why access is restricted, who can use the data and under what circumstances

VI. Data management responsibilities and resources

My promotor Jantien Stoter, group 3D GeoInformation, Section Urban Data Science of the Department of Urbanism, with email address J.E.Stoter@tudelft.nl.

My co-promotor Charalampos Andriotis, group AiDAPT Lab, Department of Architectrual Engineering +

Technology, with email address C.Andriotis@tudelft.nl.

4TU.ResearchData is able to archive 1TB of data/code per researcher per year free of charge for all TU Delft researchers. We do not expect to exceed this and therefore there are no additional costs of long term preservation.

B

- Faculty of Architecture and the Built Environment (ABE)

C

Appendix C

The following tables include all Graduate School courses I have currently completed and plan to follow.

	GS credits	Status	Year
Discipline Related Skills			
ABE 023: Research Data Management	1	Completed	1
ABE009: Research Proposal for Architecture & the Built Environment	4	In progress	1 to 2
Reinforcement Learning Specialization Course	5	Planned	2
Machine Learning Course, e.g. GEO5017,CEGM2003 or DSAIT4010	5	Planned	2
Total GS Credits	15		
GS Credits obtained by Go/No-Go date	1		
GS Credits in-progess by Go/No-Go date	4		
Resarch Skills			
R1.A1 | Research Design	3	Completed	1
R2.B1 | Speedreading & Mindmapping	1.5	Completed	1
Learning on the Job: Addressing a large audience (examples: speaker at major international conference / workshop incl. conf. paper*	1	Completed	1
Learning on the Job: Writing the first conference paper*	1	Completed	1
R1.B1 | Navigating Academic Publishing	1	Planned	2
R3.B1 | Engineering Ethics	2.5	Planned	2

Remaining Learning on the Job*, e.g. Writing an international, peer-reviewed journal article, Supervising a Msc student or Paper review	3	Planned	2 and 3
R2.D1 | How to Formulate Successful Propositions for your PhD Defence	2	Planned	3 or 4
Total GS Credits	15		
GS Credits obtained by Go/No-Go date	6.5		
*5/5 GS credits are obtained in Learning on the Job			
Transferable Skills			
T4.G1 - AI | PhD Start-up Module A-I: Introduction to the Graduate School	0.5	Completed	1
T4.G1 - AII | PhD Start-up Module A-II: Navigating the PhD life	0.5	Completed	1
T4.G1 - AIII | PhD Start-up Module A-III: Conquering challenges	0.5	Completed	1
T4.G1 - B | PhD Start-up Module B - Scientific Integrity	0.5	Completed	1
T3.A1 | Foundations of Educational Design	1	Completed	1
T1.F1 | LinkedIn for Researchers	1	Completed	1
T1.D4 | Advanced English for the University	4	Completed	1
T1.B3 | Writing a Scientific Article in English	5	In progress	1 to 2
T3.A2 | Small Group Teaching and Lecturing	1	Planned	2
T4.F2 | Career Development - Making Impact with your Thesis**	1.5	Planned	3
Total GS Credits	15.5		
GS Credits obtained by Go/No-Go date	8		
**1.5/1 GS credits are obtained in Career Development			

Table C.1: Overview of GS courses for discipline related skills, research skills and transferable skills. The total number of GS credits to obtain is 45 in total; 15 per category. All mandatory courses have been completed.

D
Appendix D

The following pages contain the first conference paper, submitted to ICOSSAR'25 on May 10th, 2025, titled: Modelling Stochastic Degradation and Maintenance Effects for the Road Network of Amsterdam: A Multi-attribute Data-driven Approach.

14th International Conference on Structural Safety and Reliability - ICOSSAR'25
1-6 June, 2025 - Los Angeles California, USA
R. Ghanem, E. Johnson, S. Masri and A. Olivier (Eds)

MODELLING STOCHASTIC DEGRADATION AND MAINTENANCE EFFECTS FOR THE ROAD NETWORK OF AMSTERDAM: A MULTI-ATTRIBUTE DATA-DRIVEN APPROACH

AMY G.E. STERRENBERG, CHARALAMPOS P. ANDRIOTIS, JANTIEN E. STOTER

Faculty of Architecture and the Built Environment, Delft University of Technology
2628 BL Delft, The Netherlands
A.G.E.Sterrenberg@tudelft.nl, C.Andriotis@tudelft.nl, J.E.Stoter@tudelft.nl

D

Key words: Stochastic degradation modelling; Maintenance impact estimation; Life-cycle extension; Predictive maintenance; Data-driven infrastructure management; Markov models

Abstract. Data-driven prediction of infrastructure aging is challenging due to the complex stochastic nature of degradation effects and the ill-documented historical records. Degradation modeling is, however, crucial for predictive maintenance that is key for infrastructure integrity. This study presents a multi-attribute, data-driven approach for modelling stochastic degradation and maintenance effects of roads, mining an extensive database of geo-located historical inspection and maintenance records from the municipality of Amsterdam. Inspection data track pavement conditions at irregular intervals across ten discrete states per road segment, following the Dutch CROW 146 protocol. Damage severity and extent for eight damage modes is captured, i.e., for transverse unevenness, irregularities, ravelling, edge damage, crack formation, joint filling, joint width, and settling. The maintenance dataset includes $>$25k minor interventions across 17k road segments, indicating repair requirements, and 200+ major maintenance projects, covering 21k segments where interventions are planned, all without verifying completion. This complicates accurate modelling of natural degradation as it is confounded by maintenance effects. To address the issue of irregular inspections, degradation is first modelled as a continuous-time Markov chain. Thereby, transition rates are estimated, which are then converted to discrete-time Markov chain transition probability matrices to eventually support regular maintenance planning. We further learn the effects of minor and major maintenance activities, as defined and recorded in the database. Based on the estimated degradation transitions, pre-maintenance and post-maintenance state distributions are estimated. Instantaneous maintenance transition matrices are computed by minimizing the cross-entropy between the pre-maintenance state after the intervention and the post-maintenance state. The model allows for a multi-attribute approach, segmenting roads based on construction material (e.g., asphalt, tiled pavement) and traffic loads (e.g., residential, commercial/pedestrian). The approach is exemplified for tiled pavements for a section of the road network of Amsterdam, where the effects of minor and major maintenance are ablated for long-term predictions. Although applied to Amsterdam, this method is relevant to any infrastructure system with discrete state datasets, providing a foundation for data-driven decision-making in infrastructure management.

1 INTRODUCTION

Urban transportation infrastructure is critical for mobility, economic activity, and the serviceability of cities. The maintenance of these systems is essential to preserve operational reliability and safety. However, infrastructure management involves complex, multi-objective decision-making under resource constraints. Authorities do not only prioritise interventions based on urgency and cost-effectiveness, but also consider broader system-level impacts such as network disruptions, emissions, accessibility, and safety risks. To control these impacts as demands on infrastructure increase, efficient resource allocation becomes more and more crucial.

Reactive, unplanned or corrective maintenance, where damage is addressed as it is observed, remain common practice by operating agencies. While these approaches minimise short-term expenditures, they are generally associated with higher overall maintenance costs, increased risk of asset failure, and reduced system reliability in the long run [1, 2, 3, 4]. In contrast, predictive maintenance (PdM) offers a more proactive alternative, using condition monitoring and data analysis to anticipate infrastructure needs and optimise intervention timings [1, 3, 5].

Accurate degradation modelling facilitates the development of effective PdM decision-making systems. While data quality and availability have been limiting factors in the past, data-driven degradation models have become more and more prevalent and have been extensively explored for various applications, using methods such as statistical inference, probabilistic graphs, including Markov models, and machine learning [6]. Similar models have been developed for pavements (e.g., [7, 8, 9, 10, 11, 12, 13, 14, 15]). These models are typically calibrated using indicators of pavement condition, such as the pavement condition index (PCI), international roughness index (IRI), pavement serviceability index (PSI), or pavement condition rating (PCR). Such continuous indices allow modeling through continuous-state stochastic models such as gamma processes [16]. Additionally, most studies focus on asphalt and concrete pavements, for which these indicators are available, limiting their applicability to urban road networks characterised by heterogeneous attributes, such as diverse pavement materials, traffic loads, damage modes and other functional classifications [17]. Moreover, models explicitly quantifying the effect of maintenance activities on road condition are currently limited. Although some studies have explored post-maintenance condition changes (e.g., [18, 19]), there is limited integration between degradation and maintenance modelling in a unified probabilistic framework. This omission constrains the accuracy of long-term maintenance planning and limits the potential for rigorous predictive optimisation methods, such as Markov decision processes and reinforcement learning, to be applied in practice [20, 16, 21, 22].

This study presents a data-driven, multi-attribute framework for the modelling of both road degradation and maintenance effects, designed to support predictive maintenance in heterogeneous road networks present in urban environments. The model is developed using inspection, maintenance and road data from the municipality of Amsterdam, the Netherlands. The inspection data includes information on the severity and extent of damage across up to eight predefined damage modes, with up to six damage modes depending on pavement construction type. The methodology allows for condition transitions to be modelled as probabilistic state changes conditioned on segment-specific attributes, including construction type and traffic pattern categories. These transitions are initially formulated as a continuous-time Markov chain (CTMC) to address the irregular time intervals present in the observational data and are subsequently converted into a discrete-time Markov chain (DTMC) to support integration with regular, time-stepped maintenance planning frameworks. The resulting transitions form a high-dimensional tensor, which enables detailed, scenario-based simulation of network condition under varying intervention policies. As a result, the proposed framework supports future integration with decision-support tools for high-fidelity predictive maintenance planning at scale, offering an interpretable and empirically grounded basis for modelling deterioration of diverse road networks. Results regarding future prediction in an area of Amsterdam are discussed, ablating minor and major maintenance scenarios.

2 METHODS

2.1 Data

Historical inspection and maintenance data provided by the municipality of Amsterdam are used. The inspection dataset contains manually collected road condition data in accordance with the CROW 146b protocol, which is the national standard for infrastructure inspections in the Netherlands. Inspections are conducted per road segment, defined as a continuous section of road between intersections with a maximum length of 100 metres. The dataset is primarily used by the municipality for maintenance planning, administrative reporting, and compliance with regulatory obligations.

D

Inspection records have been collected since approximately 2015, with road segments inspected at irregular time intervals, mostly every two to three years. Each inspection record includes a condition label that reflects the most severe observed damage within the segment. These labels—G, $L1$–$L3$, $M1$–$M3$, and $E1$–$E3$—represent a combination of severity (G = Good, L = Light, M = Moderate, E = Serious) and extent of the affected area (1 = small, 2 = average, 3 = large).

Eight damage modes are used in the classification: transverse unevenness; irregularities; ravelling; edge damage; crack formation; joint filling; joint width; and settling. Up to six unique damage modes are relevant per pavement category: asphalt segments exhibit transverse unevenness, irregularities, ravelling, edge damage, cracking, and settling; tiled pavements show transverse unevenness, irregularities, joint width, and settling; and concrete surfaces are assessed for irregularities, cracking, joint filling, and settling. However, no or very limited degradation is represented in inspection records for settling and joint filling. Although the CROW protocol defines boundary conditions for each label and damage mode, expert opinion suggests a degree of human error in the dataset.

Additional features are available for each road segment, including geographical information (i.e., neighbourhood, region and geometry), construction type (e.g., asphalt, concrete, element pavement, semi-paved, synthetic, unpaved), specific surface material, year of construction or last major reconstruction, date of last conservation treatment, surface area, and traffic pattern classification (e.g., heavy, residential, cycle path). The completeness of this data varies across records.

In addition to inspection records, two separate datasets of maintenance records are used: one for minor interventions and one for major interventions. The minor maintenance dataset contains 25,044 intervention points identified by road inspectors, which we can link to 17,813 unique road segments. Each record includes coordinates, date of data collection, pavement type, damage category, damage size, and corresponding unit of measurement. The major maintenance dataset includes 228 planned interventions covering larger areas, represented spatially as polygons or multi-polygons. These polygons typically include multiple road segments. Each record documents the date of data collection, project name, category, budget, asset group, source, project status, and planned phases of execution: preparation start, intervention start, and completion date, each recorded by year. The total area scheduled for maintenance is also included.

2.2 Modelling Approach

In this study, both the degradation and maintenance effects are modelled through probabilistic Markovian transition matrices, derived from empirical data. This section first outlines the fitting of degradation probability matrices, followed by a description of how minor and major maintenance effects are incorporated.

Degradation Modelling

Degradation of road infrastructure is modelled as a stochastic process using a continuous-time Markov chain (CTMC). This model represents transitions between a finite set of condition states in continuous time, which evolve over time depending on discrete actions. The probability of transitioning to a future state adheres to the Markov property, meaning that the probability of moving to a future state depends solely on the current state and not on the sequence of preceding states. The CTMC model is based on several key assumptions. First, transitions between states are assumed to be independent across road segments; that is, the degradation of one segment does not influence others. Second, the model assumes stationarity of transition rates, implying that these remain constant over time.

A Markov process is defined as $\{X(t), \, t \geq 0\}$ on a finite state space $S = \{1, 2, \ldots, n\}$. In this work, the state space has $n = 10$ states, corresponding to the inspection labels G–$E3$, described in Section 2.1. The transition between states is characterized by a transition probability matrix, where the probability of transitioning from state $i \in S$ to state $j \in S$ in exactly time Δt from an initial time t is denoted as [23]:

$$P_{ij}(t) = P(X(t + \Delta t) = j | X(\Delta t) = i) \tag{1}$$

This defines the transition probability matrix $P(t)$ at time t:

$$P(t) = \begin{bmatrix} P_{11}(t) & P_{12}(t) & \cdots & P_{1n}(t) \\ P_{21}(t) & P_{22}(t) & \cdots & P_{2n}(t) \\ \vdots & \vdots & \ddots & \vdots \\ P_{n1}(t) & P_{n2}(t) & \cdots & P_{nn}(t) \end{bmatrix} \tag{2}$$

Each row of $P(t)$ sums up to 1, and all entries satisfy $0 \leq P_{ij}(t) \leq 1$. The transition probability matrix for timestep t is used in discrete-time Markov processes, where $P_{ij}(t)$ can be estimated empirically by the observed transition frequencies [24, 25, 26]:

$$P_{ij}(t) = \frac{n_{ij}}{n_i} \tag{3}$$

where n_{ij} is the number of transitions from state i to state j over the observation period, and n_i is the total number of transitions from state i to any other state in the state space. However, given that the available inspection data in this study contains variable time intervals, a discrete-time approach is not suitable. Therefore, we employ CTMCs to model the degradation process. In this case, transition probability matrices are derived from transition rate matrices. Assuming exponentially distributed sojourn and transition times, the transition rate matrix Q takes the form:

$$Q = \begin{bmatrix} -\lambda_1 & \lambda_{12} & \cdots & \lambda_{1n} \\ \lambda_{21} & -\lambda_2 & \cdots & \lambda_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \lambda_{n1} & \lambda_{n2} & \cdots & -\lambda_n \end{bmatrix} \tag{4}$$

Following Eq 1, each off-diagonal entry λ_{ij} for $i \neq j$ is defined as the rate of transition from state i to state j [27, 28]:

$$\lambda_{ij} = \lim_{\Delta t \to 0^+} \frac{P(X(\Delta t) = j \mid X(0) = i)}{\Delta t} \tag{5}$$

The diagonal elements λ_i in the transition rate matrix Q reflect the total rate at which the segment leaves state i, and are defined such that each row of Q sums to zero:

$$\lambda_i = \sum_{j \neq i} \lambda_{ij} \tag{6}$$

Under the exponential distribution assumptions, the transition rates λ_{ij} are estimated from the observed transitions in the inspection dataset. The estimate for each λ_{ij} is given by [27, 29]:

$$\lambda_{ij} = \frac{n_{ij}}{\sum_{k=1}^{n_i} T_i^{(k)}} \tag{7}$$

where: $T_i^{(k)}$ is the observed sojourn time in state i before transitioning, for the k-th transition. To obtain the corresponding transition probability matrix $P(t)$ for a desired time horizon t, we compute the matrix exponential [23]:

$$P(t) = e^{Qt} \tag{8}$$

This equation allows us to derive discrete-time transition probability matrices, which can subsequently be used in advanced decision-making algorithmic frameworks for predictive maintenance (Figure 1), such as a Markov decision process or within (deep) reinforcement learning.

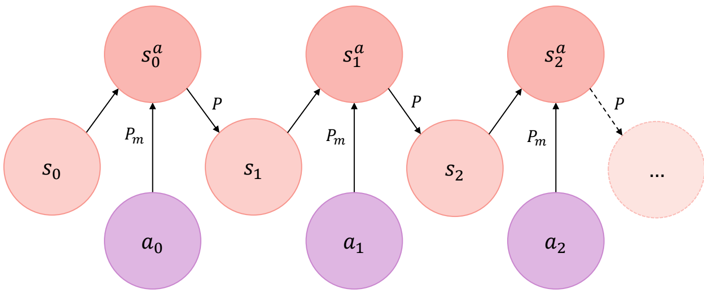


Figure 1: Probabilistic graph of the discrete time Markov chain model.

Maintenance Effect Modelling

Maintenance effects are modelled separately for minor and major maintenance interventions. Transitions for which a maintenance action is recorded are considered in this process. Condition improvements without maintenance logs are excluded, as we lack sufficient data to interpret them reliably. As maintenance records include planned interventions, without confirmation of execution, the dataset may contain cases of failed, postponed, or unperformed maintenance. Additionally, some maintenance-logged transitions result in deterioration of condition. These cases can be interpreted as either: (1) failed maintenance, (2) maintenance unrelated to the observed damage mode, or (3) rapid post-maintenance degradation. Each inspection record includes the condition of the segment for multiple damage modes, dependent on road construction category. We assume all damage modes are subject to the maintenance intervention, regardless of the resulting change in condition.

The effect of a maintenance action taken at time t_m is modelled as an instantaneous transition probability matrix P_m, representing the maintenance effect between the time just before the maintenance intervention, t_m^-, and the time immediately after, t_m^+. Using the estimated degradation transition rate matrix Q_d from Eq.6 and the state probability distributions at the start, $p(t_{\text{start}})$, and end, $p(t_{\text{end}})$, of the observed transition, the predicted state vector just before maintenance is computed as:

$$p(t_m^-) = p(t_{\text{start}}) \cdot e^{Q_d(t_m - t_{\text{start}})} \tag{9}$$

The predicted state just after maintenance is computed through a backward transition as:

$$p(t_m^+) = p(t_{\text{end}}) \cdot e^{-Q_d(t_{\text{end}} - t_m)} \tag{10}$$

For major maintenance, the exact intervention date t_m is unknown. We assume it occurs at the midpoint of the time interval between inspections. Next, P_m is estimated for k observed maintenance events so that:

$$\begin{bmatrix} p(t_m^-)^{(1)} \\ p(t_m^-)^{(2)} \\ \vdots \\ p(t_m^-)^{(k)} \end{bmatrix} \cdot P_m = \begin{bmatrix} p(t_m^+)^{(1)} \\ p(t_m^+)^{(2)} \\ \vdots \\ p(t_m^+)^{(k)} \end{bmatrix} \tag{11}$$

We assume that P_m in Eq.11 has a known shape but unknown parameters. The shape is given by a linear combination between a perfect repair and a no-repair action. These transition matrices are denoted as P_m^1 and P_m^2, respectively. P_m^1 represents a transition probability matrix where $p(s' = G \mid s) = 1$ for all $s \in S$. This indicates that, regardless of the current state s, the system deterministically transitions to state G, effectively resetting the system to the best condition. P_m^2 is the identity matrix I over the state space S, implying that no transition occurs and the system remains in its current state. As such, the instantaneous transition for each damage mode assumes the following form:

$$P_m = w_1 \cdot P_m^1 + w_2 \cdot P_m^2, \quad w_1 + w_2 = 1 \tag{12}$$

Subsequently, we optimize the weights w_1 and w_2 based on minimizing the cross-entropy loss between predicted and observed post-maintenance state distributions across all N observed transitions:

$$L = \frac{1}{N} \sum_{j=1}^{N} \left(- \sum_{i=1}^{n} p_i^{(j)}(t_m^+) \log \hat{p}_i^{(j)}(t_m^+) \right) \tag{13}$$

where $p_i^{(j)}(t_m^+)$ is the probability of sample j being in state i at time t_m^+ based on Eq 10 and $\hat{p}_i^{(j)}(t_m^+)$ is the predicted probability of sample j being in state i at time t_m^+ based on Eq 11.

D

3 RESULTS

3.1 Data Selection

The dataset comprises comprehensive inspection records, corresponding to eight unique damage modes with up to six distinct damage types per road segment, in addition to feature data concerning road construction materials and traffic pattern classifications. This level of detail facilitates the specification of degradation and maintenance effects across various combinations of material and traffic pattern attributes. However, the dataset does not provide sufficient coverage to enable robust modelling of all such combinations.

With regard to construction materials, tiled pavements and asphalt pavements constitute the majority of the network. Specifically, 80.1% of roads in Amsterdam are classified as tiled pavements, and 15.9% as asphalt pavements. All remaining material categories individually represent less than 3% of the road network. While further disaggregation of the transition model (e.g., by traffic pattern classification) is possible, for the sake of brevity, aggregated results for tiled pavements across all traffic pattern classifications and for damage modes 'transverse unevenness', 'irregularities' and 'joint width' are presented here.

3.2 Degradation Modelling

Degradation processes are modelled by estimating transition probability matrices derived from observed condition transitions. Figure 2 presents the 8-year transition probability matrices for the three dominant damage modes observed in tiled pavements within the Amsterdam road network: transverse unevenness, irregularities, and joint width. Each matrix displays the starting states along the vertical axis and the corresponding ending states along the horizontal axis. States are coded as follows: G (good/no damage), $L1 - L3$ (minor damage), $M1 - M3$ (moderate damage), and $E1 - E3$ (serious damage), with numerical indices denoting the spatial extent of damage (1 = small, 2 = average, 3 = large).

Table 1 summarises the total number of recorded transitions for each damage mode, as well as the proportion of observations in which no change in condition occurred ($G \rightarrow G$). High proportions

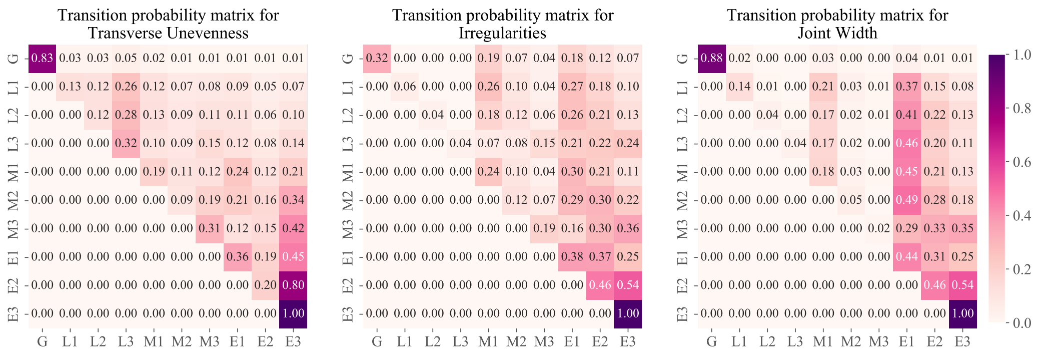


Figure 2: Eight-year transition probability matrices for the three relevant damage modes in tiled pavements across the Amsterdam road network: transverse unevenness, irregularities, and joint width. Condition states range from G (good/no damage) to E3 (severe damage affecting a large area), with severity increasing from top to bottom and left to right.

Table 1: Logged transitions and proportion of 'G' → 'G' transitions by damage mode

Damage Mode	Number of logged transitions	Number of 'G' → 'G' transitions
Transverse unevenness	73,249	67,087 (91.6%)
Irregularities	60,086	22,878 (38.1%)
Joint width	78,577	74,220 (94.5%)

of such stable states, particularly in joint width (94.5%) and transverse unevenness (91.6%), suggest relatively slow deterioration processes for those damage types. In contrast, the proportion of unchanged states is considerably lower for irregularities (38.1%), indicating more dynamic or variable degradation behaviour.

D

Analysis of the dataset shows that some condition states, particularly those representing more severe or extensive damage, are sparsely populated. As such, the reliability of transition probability estimates from these states, specifically $M3$, $E2$, and $E3$, is limited. Notably, for irregularities, the starting states $L2$ and $L3$ are also especially underrepresented. To isolate degradation processes, transitions reflecting stable or worsening conditions are included in the estimation. The most severe state ($E3$) is treated as an absorbing or terminal state in this model.

The values along the diagonal of each matrix indicate the probability that, over an eight-year period, segments remain in their current condition state. These values are especially high for $G \rightarrow G$ transitions for the damage modes 'transverse unevenness' and 'joint width'. Diagonal values are also higher for states indicating high damage severity ($E1$–$E3$), with an exception for $E2$ in transverse unevenness. For transverse unevenness, however, state transitions $L3 \rightarrow L3$ and $M3 \rightarrow M3$ are also relatively high. Non-zero values in the upper triangle suggest a gradual risk of deterioration. In particular, transitions from any state except G to $M1$ and $E1$ appear relatively likely for 'irregularities', and 'joint width' damage types. This trend is especially pronounced for joint width, suggesting that once deterioration initiates, it escalates more rapidly in this damage mode. Zero entries in the upper triangle of the matrix imply the absence of observed transitions to these states. However, although certain states are rounded down to zero in the matrices presented in Figure 2, the associated transition rates indicate existence of small probabilities. Finally, for transverse unevenness and irregularities, some transitions that indicate an increase in damage severity, while damage extent decreases or vice versa, such as $M3 \rightarrow E1$, are found to have slightly lower values compared to other transitions in the same row.

3.3 Maintenance Effect Modelling

Maintenance effects were modelled through the estimation of instantaneous transition probability matrices, derived from observed condition transitions that could be directly associated with either minor or major maintenance interventions. The weights of these matrices, estimated using observed transitions combined with a set of prototype matrices based on Eq. 12, are represented in Table 2. These describe the prototype matrices with the lowest cross-entropy relative to the empirical transition data, as outlined in the methodology section.

As dictated in Eq 12, w_1 indicates how often maintenance interventions result in perfect repair (transition to state G) and w_2 indicates no observable change in condition. Results are shown in Table 3. Major maintenance consistently yields higher probabilities of perfect repair compared to minor maintenance. When comparing across damage modes, joint width exhibits the highest likelihood of

Table 2: Learned weights for maintenance impacts by damage mode and maintenance type.

Weight	Transverse Unevenness		Irregularities		Joint Width	
	Minor	Major	Minor	Major	Minor	Major
w_1	0.597	0.701	0.284	0.442	0.585	0.766
w_2	0.403	0.299	0.716	0.558	0.415	0.234

perfect repair, followed closely by transverse unevenness, whereas irregularities show substantially lower perfect repair rates. Additionally, for damage modes 'transverse unevenness' and 'joint width', the probability for perfect repair is always higher than the probability for no change in condition, indicating that maintenance interventions more commonly result in a perfect repair than no repair. For irregularities, maintenance actions have a smaller effect.

3.4 Application

Figure 3 illustrates a section of the Amsterdam road infrastructure network, to which the previously developed degradation and maintenance effect models have been applied. These comparative visualisations highlight the varying impact of maintenance strategies on the long-term condition of the network. The initial state of the network is shown in Figure 3a. Figure 3b illustrates the condition development over a eight-year period under natural degradation, without any intervention, indicating the condition state as the highest probability in the distribution over states for transverse unevenness. While some segments remain in their initial condition category, others exhibit progressive deterioration, particularly those that began in more severe damage states. Segments initially in perfect or

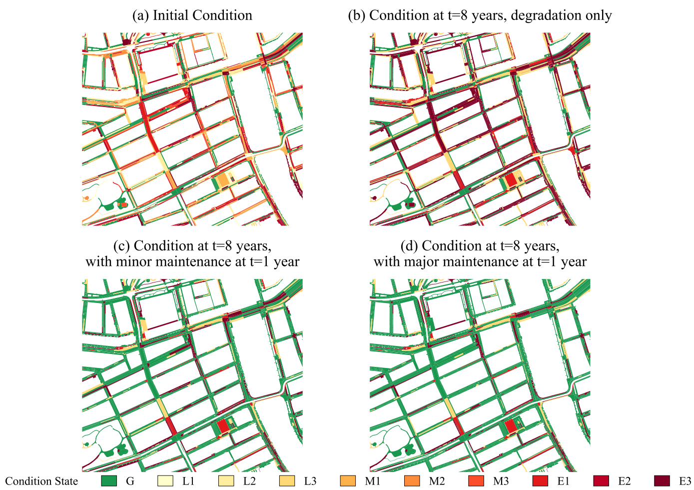


Figure 3: Visualisation of the condition evolution of a section of Amsterdam's road network under different maintenance scenarios, assuming tiled pavements for all segments: (a) Initial condition state; (b) Projected condition after eight years of natural degradation without intervention; (c) Condition after eight years with minor maintenance applied uniformly after one year; (d) Condition after eight years with major maintenance applied uniformly after one year. The comparison illustrates the differential impact of maintenance strategies on long-term infrastructure condition.

lightly deteriorated states generally show slower rates of degradation. Figure 3c shows the same scenario with minor maintenance applied uniformly to all road segments after year one, while Figure 3d reflects the scenario in which major maintenance is implemented at the same time point. In both cases, a substantial number of segments are restored to the optimal condition state; however, a subset of road segments shows no improvement following intervention. This proportion is notably higher under the minor maintenance scenario than under major maintenance, indicating the limited effectiveness of minor interventions in reversing deterioration in certain cases.

4 DISCUSSION

The data-driven model developed in this work enables degradation and maintenance effects transition modelling, which can serve as a foundation for predictive maintenance systems. The model captures the stochastic nature of deterioration and is grounded in condition data and classification schemes that align with those used by human maintenance planners. This makes it not only analytically robust but also potentially interpretable and actionable in operational contexts.

At the same time, several limitations exist with respect to the statistical analysis. First, the dataset was not originally collected for predictive modelling. It only records the most prominent manifestation of each damage mode and lacks precision regarding the exact timing and nature of maintenance interventions, as discussed in Section 2.1. This restricts our ability to draw strong conclusions about causal links between specific maintenance actions and changes in condition for particular damage types. As a result, observed condition improvements are associated with general maintenance categories (i.e., minor or major), as it is difficult to associate them with targeted treatments of, for example, transverse unevenness or joint width,

Another key constraint is data sparsity, particularly in higher-severity condition states. Some transitions, especially from rarely observed states, such as $E2$ or $E3$, are based on a few records, limiting the statistical confidence of the corresponding transition probabilities. This restricts the model's applicability in representing deterioration and maintenance effects across the full range of condition states.

Expanding the dataset to better cover underrepresented road types or damage states would improve the robustness and predictive power of the model, as well as increase its generalisability across the network. Despite these limitations, the modelling approach presented in this work demonstrates the potential of inspection records to support data-informed maintenance strategies. With improved data infrastructure, such models can significantly contribute to planning and prioritising maintenance interventions in urban road networks.

5 CONCLUSIONS

A transition model for infrastructure degradation and maintenance effects is presented. The model consists of transition probability matrices for a selection of road construction materials and traffic pattern labels and is aimed at applications in predictive maintenance models. Historical inspection and maintenance records from the municipality of Amsterdam were used, comprising data collected per road segment in accordance with the CROW 146b protocol. The inspection records include condition labels based on eight damage categories, together with additional features, such as road material, construction year, and traffic pattern category. Records of both minor and major maintenance activities are available at the same spatial resolution as the inspection data, enabling the modelling of condition state transitions under natural degradation, minor maintenance, and major maintenance. Degradation was modelled as a continuous-time Markov chain. This approach accounted for the irregular inspection intervals by estimating stationary transition rates from observations, assuming exponentially distributed transition times. Accordingly, transition probability matrices were derived through the exponential of the rate matrix. To address data uncertainties, including unexecuted or incorrectly logged maintenance, transition matrices were introduced, as a linear combination of perfect repair and no (or unsuccessful) repair. The maintenance effect matrix was estimated by comparing predicted and observed post-maintenance states, selecting the matrix weights by the lowest cross-entropy loss. Results were presented for tiled pavements and three damage modes: transverse unevenness; irregularities; and joint width. The degradation analysis showed that for the lowest and highest condition states, higher transition probabilities are generally found along the matrix diagonal. This suggests limited observed change in these states. Non-zero upper-triangular values, especially for transitions not originating from state G, indicate a comparatively faster deterioration once initiated. However, data sparsity in more severe condition states limits the reliability of certain estimates. Results also show that both minor and major maintenance commonly lead to perfect repair for 'transverse unevenness' and 'joint width' damage modes. For irregularities, maintenance actions were found more likely to not affect road condition. For all damage modes, major maintenance consistently delivers higher rates of full restoration. Among the three damage types, joint width was most responsive to maintenance, followed by transverse unevenness. Irregularities were less responsive. Overall, the results highlight differences in degradation pace and maintenance effectiveness across damage types and emphasise the need to account for these distinctions in infrastructure planning and maintenance strategies.

References

[1] Laura Swanson. "Linking maintenance strategies to performance". In: *Int. J. Prod. Econ.* 70.3 (2001), pp. 237–244.

[2] Keith R. Mobley. *An Introduction to Predictive Maintenance*. 2nd. Elsevier, 2002.

[3] P. Coandă, M. Avram, and V. Constantin. "A state of the art of predictive maintenance techniques". In: *IOP Conf. Ser. Mater. Sci. Eng.* Vol. 997. 1. 2020.

[4] Hikmet Erbiyik. "Definition of Maintenance and Maintenance Types with Due Care on Preventive Maintenance". In: *Maint. Manag. - Curr. Challenges, New Dev. Futur. Dir.* (2023).

[5] Mohammad M. Hamasha, Ala H. Bani-Irshid, Sahar Al Mashaqbeh, Ghada Shwaheen, Laith Al Qadri, Mohammad Shbool, Dania Muathen, Mussab Ababneh, Shahed Harfoush, Qais Albedoor, and Adnan Al-Bashir. "Strategical selection of maintenance type under different conditions". In: *Sci. Rep.* 13.1 (2023).

[6] Anna Jarosz, Marta Zagorowska, and Jerzy Baranowski. "Recent advances in data-driven methods for degradation modelling across applications". In: (2025).

[7] Rabi G. Mishalani and Samer M. Madanat. "Computation of Infrastructure Transition Probabilities Using Stochastic Duration Models". In: *J. Infrastruct. Syst.* 8.4 (2002), pp. 139–148.

[8] K. P. George, A. S. Rajagopal, and L. K. Lim. "Models for predicting pavement deterioration". In: *Transp. Res. Rec.* 1215 (1989), pp. 1–7.

[9] Turki I. Al-Suleiman and Adnan M.S. Shiyab. "Prediction of Pavement Remaining Service Life Using Roughness Data - Case Study in Dubai". In: *Int. J. Pavement Eng.* 4.2 (2003), pp. 121–129.

[10] Nader Abdelaziz, Ragaa T. Abd El-Hakim, Sherif M. El-Badawy, and Hafez A. Afify. "International Roughness Index prediction model for flexible pavements". In: *Int. J. Pavement Eng.* 21.1 (2020), pp. 88–99.

[11] Saeid Alimoradi, Amir Golroo, and Seyed Mohammad Asgharzadeh. "Development of pavement roughness master curves using Markov Chain". In: *Int. J. Pavement Eng.* 23.2 (2022), pp. 453–463.

[12] S. Madeh Piryonesi and Tamer E. El-Diraby. “Data Analytics in Asset Management: Cost-Effective Prediction of the Pavement Condition Index”. In: *J. Infrastruct. Syst.* 26.1 (2020).
[13] Ningyuan Li, Wei Chau Xie, and Ralph Haas. “Reliability-based processing of markov chains for modeling pavement network deterioration”. In: *Transp. Res. Rec.* 1524 (1996), pp. 203–213.
[14] Kiyoshi Kobayashi, Kiyoyuki Kaito, and Nam Lethanh. “A Bayesian Estimation Method to Improve Deterioration Prediction for Infrastructure System with Markov Chain Model”. In: *Int. J. Archit. Eng. Constr.* 1.1 (2012), pp. 1–13.
[15] Jhenyffer Lorrany Matias de Oliveira, Gary Davis, Alireza Khani, and Mihai Marasteanu. “Heterogeneous Markov Chain Model to Predict Pavement Performance and Deterioration”. In: *Transp. Res. Rec.* 2676.9 (2022), pp. 568–581.
[16] M. Saifullah, K. G. Papakonstantinou, C. P. Andriotis, and S. M. Stoffels. “Multi-agent deep reinforcement learning with centralized training and decentralized execution for transportation infrastructure management”. In: i (2024), pp. 1–42.
[17] Amir Shtayat, Sara Moridpour, Berthold Best, and Shahriar Rumi. “An Overview of Pavement Degradation Prediction Models”. In: *J. Adv. Transp.* 2022 (2022).
[18] Putu Mandiartha, Colin F. Duffield, Russell G. Thompson, and Marcus R. Wigan. “Measuring pavement maintenance effectiveness using Markov Chains analysis”. In: *Struct. Infrastruct. Eng.* 13.7 (2017), pp. 844–854.
[19] Yaning Qiao, Andrew Dawson, Tony Parry, and Gerardo W. Flintsch. “Immediate effects of some corrective maintenance interventions on flexible pavements”. In: *Int. J. Pavement Eng.* 19.6 (2018), pp. 502–508.
[20] Samer Madanat. “Optimal infrastructure management decisions under uncertainty”. In: *Transp. Res. Part C* 1.1 (1993), pp. 77–88.
[21] C. P. Andriotis and K. G. Papakonstantinou. “Managing engineering systems with large state and action spaces through deep reinforcement learning”. In: *Reliab. Eng. Syst. Saf.* 191.April (2019), p. 106483.
[22] Konstantinos G. Papakonstantinou, Charalampos P. Andriotis, and Masanobu Shinozuka. “POMDP and MOMDP solutions for structural life-cycle cost minimization under partial and mixed observability”. In: *Struct. Infrastruct. Eng.* 14.7 (2018), pp. 869–882.
[23] Sheldon Ross. *Introduction to Probability Models*. 11th ed. Elsevier, 2014.
[24] Yi Jiang and Kumares C. Sinha. “Bridge service life prediction model using the Markov chain”. In: *Transp. Res. Rec.* 1223 (1989), pp. 24–30.
[25] Kamalesh Panthi. “A Methodological Framework for Modeling Pavement Maintenance Costs for Projects with Performance-based Contracts”. In: (2009), pp. 1–191.
[26] Ángela Alonso-Solorzano, Heriberto Pérez-Acebo, Daniel J. Findley, and Hernán Gonzalo-Orden. “Transition probability matrices for pavement deterioration modelling with variable duty cycle times”. In: *Int. J. Pavement Eng.* 24.2 (2023).
[27] Yasunari Inamura. “Estimating Continuous Time Transition Matrices From Discretely Observed Data”. In: *Bank Japan Work. Pap. Ser.* E07.6 (2006), pp. 1–40.
[28] Ward Whitt. “Continuous-time Markov chains”. In: *Dept. of Industrial Engineering and Operations Research, Columbia University, New York* (2006), p. 164.
[29] Toualith Jean Marc Meango and Mohamed Salah Ouali. “Failure interaction model based on extreme shock and Markov processes”. In: *Reliab. Eng. Syst. Saf.* 197 (2020).