Maintenance Strategies for Sewer Pipes with Multi-State Degradation and Deep Reinforcement Learning

Lisandro A. Jimenez-Roa [1], Thiago D. Simão [2], Zaharah Bukhsh [2], Tiedo Tinga [1], Hajo Molegraaf [3], Nils Jansen [4,5], and Mariëlle Stoelinga [1,4]

[1] *University of Twente, Enschede, 7522 NB, The Netherlands*
{l.jimenezroa, t.tinga, m.i.a.stoelinga}@utwente.nl

[2] *Eindhoven University of Technology, Eindhoven, 5612 AE, The Netherlands*
{t.simao@tue.nl, z.bukhsh}@tue.nl

[3] *Rolsch Assetmanagement, Enschede, 7521 AG, The Netherlands.*
hajo.molegraaf@rolsch.nl

[4] *Radboud University, Nijmegen, 6525 XZ, The Netherlands.*
n.jansen@science.ru.nl

[5] *Ruhr-University Bochum, Bochum, 44801, Germany*

Abstract

Large-scale infrastructure systems are crucial for societal welfare, and their effective management requires strategic forecasting and intervention methods that account for various complexities. Our study addresses two challenges within the Prognostics and Health Management (PHM) framework applied to sewer assets: modeling pipe degradation across severity levels and developing effective maintenance policies. We employ Multi-State Degradation Models (MSDM) to represent the stochastic degradation process in sewer pipes and use Deep Reinforcement Learning (DRL) to devise maintenance strategies. A case study of a Dutch sewer network exemplifies our methodology. Our findings demonstrate the model's effectiveness in generating intelligent, cost-saving maintenance strategies that surpass heuristics. It adapts its management strategy based on the pipe's age, opting for a passive approach for newer pipes and transitioning to active strategies for older ones to prevent failures and reduce costs. This research highlights DRL's potential in optimizing maintenance policies. Future research will aim improve the model by incorporating partial observability, exploring various reinforcement learning algorithms, and extending this methodology to comprehensive infrastructure management.

Abbreviations

DRL	Deep Reinforcement Learning.
IHTMC	Inhomogeneous Time Markov Chain.
MDP	Markov Decision Process.
MPO	Maintenance Policy Optmization.
MSDM	Multi-State Degradation Model.
PPO	Proximal Policy Optimization.
RL	Reinforcement Learning.

1. Introduction

Sewer network systems, crucial for public health, population well-being, and environmental protection, require maintenance to ensure their reliability and availability (Cardoso et al., 2016). This maintenance is challenged by limited budgets, environmental changes, aging infrastructure, and hard-to-predict system deterioration (Tscheikner-Gratl et al., 2019).

Optimizing maintenance policies for sewer networks requires methodologies that can efficiently explore a broad solution space while adapting to the system's dynamic constraints and complexities. Maintenance Policy Optmization (MPO) addresses these needs by developing and analyzing mathematical models to derive maintenance strategies (De Jonge & Scarf, 2020) that reduce maintenance costs, extend asset life, maximize availability, and ensure workplace safety (Ogunfowora & Najjaran, 2023).

This research explores the potential of Deep Reinforcement Learning (DRL) for MPO of sewer networks, first focusing on a component-level (i.e., pipe-level) analysis. DRL is a framework that merges neural network representation learning capabilities with Reinforcement Learning (RL), a branch of machine learning known for its effectiveness in sequential decision-making problems. RL is increasingly recognized for its role in developing cost-effective policies in MPO across diverse domains such as transportation, manufacturing, civil infrastructure and energy systems. It is emerging as a prominent paradigm in the search for optimal maintenance policies (Marugán, 2023).

This paper aims to achieve two primary objectives: first, to present a comprehensive model for pipe-level MPO analysis facilitated by DRL, considering degradation over the pipe length and employing inhomogeneous-time Markov chain models to simulate the nonlinear stochastic behavior associated with sewer pipe degradation; second, to assess the efficacy of the model's policy through a case study of a large-scale sewer

Lisandro A. Jimenez-Roa et al. This is an open-access article distributed under the terms of the Creative Commons Attribution 3.0 United States License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

network in the Netherlands, comparing it with heuristics, including condition-based, scheduled, and reactive maintenance.

We acknowledge as limitations in our approach the focus on *fully observable* state spaces, which means that inspection actions are not necessary, and our analysis is at the *component-level*. Future research will aim to broaden this scope to include partially observable state spaces and system-level analysis.

Contributions. This work's primary contributions include:

(i) We propose a framework to carry out maintenance policy optimization for sewer pipes considering the deterioration along the pipe length. This framework integrates Multi-State Degradation Models (MSDMs) and Deep Reinforcement Learning (DRL).
(ii) Our framework introduces a novel approach by encoding the prediction of the MSDM into the state space, aiming to harness prognostics that describe the degradation pattern of sewer pipes.
(iii) We demonstrate that DRL has the potential to devise intelligent strategic maintenance strategies adaptable to various conditions, such as pipe age.
(iv) We provide our framework in Python and all data used in this study at zenodo.org/records/11258904.

Paper outline. Section 2 presents the technical background. Section 3 outlines our research methodology. Section 4 formulates the MSDM. Section 5 details the framework for maintenance policy optimization via DRL. Section 6 presents our experimental setup. Section 7 analyzes the results. Section 8 discusses findings, concludes, and suggests future research.

Related work. In the past two decades, the need for integral sewer asset management has become evident (Abraham et al., 1998), emphasizing the necessity to understand the mechanisms of deterioration and develop predictive models for proactive and strategic sewer maintenance (Fenner, 2000). Sewer asset management encompasses maintenance, rehabilitation, and inspection and has been investigated through various methodologies, including risk-based strategies (Lee et al., 2021), multi-objective optimization (Elmasry et al., 2019), Markov Decision Processes (Wirahadikusumah & Abraham, 2003), considering the structure of the sewer network (Qasem & Jamil, 2021), machine learning applications (Montserrat et al., 2015; Caradot et al., 2018; Laakso et al., 2019; Hernández et al., 2021), and decision support frameworks (Taillandier et al., 2020; Khurelbaatar et al., 2021; Ramos-Salgado et al., 2022; Assaf & Assaad, 2023).

The integration of RL into sewer asset management is largely unexplored, with existing research mainly concentrating on *real-time control* for smart infrastructure, adapting to environmental changes such as storms. Mullapudi et al. (2020) uses DRL for controlling storm water system valves through simulation of varied storm scenarios. Yin et al. (2023) employ RL for *near real-time* control to minimize sewer overflows. Meanwhile, Zhang et al. (2023) and Tian et al. (2022) both examine improving the robustness of urban drainage systems, the former through decentralized *multi-agent RL* and the latter through *Multi-RL*, with Tian et al. (2024) further improving the model *interpretability* using DRL. Furthermore, Kerkkamp et al. (2022) investigates the sewer network MPO by combining DRL with Graphical Neural Networks to optimize maintenance actions grouping. Jeung et al. (2023) proposes a DRL-based *data assimilation* methodology to enhance storm water and water quality simulation accuracy by integrating observational data with simulation outcomes.

2. Technical background

2.1. Multi-state degradation model for sewer pipes

The modeling of sewer pipe network degradation has been explored through various methodologies, including physics-based, machine learning, and probabilistic models. For comprehensive discussions on this topic, the reader is directed to Ana & Bauwens (2010); Hawari et al. (2017); Malek Mohammadi et al. (2019); Saddiqi et al. (2023); Zeng et al. (2023).

We adopt a probabilistic approach employing Inhomogeneous Time Markov Chains (IHTMCs) to model the multi-state degradation of sewer pipes. This choice is motivated by the IHTMC's capability to better capture the degradation of long-lived assets such as sewer systems as a non-linear stochastic process, characterized by age-dependent transition probabilities between degradation states (Jimenez-Roa et al., 2024).

Inhomogeneous Time Markov Chains (IHTMCs). An IHTMC is a stochastic process $\{(X_t)\}_{t\geq 0}$, where $t \in [0, \infty)$ is continues and models *time*. The IHTMC is defined as a tuple $M = \langle \Omega, S^0, Q(t) \rangle$, where Ω is a set of K finite states indicating the *state space*, S^0_k is an *initial-state distribution* on Ω where $\sum_{k\in\Omega} S^0_k = 1$, and $Q(t) : \Omega \times \Omega \to \mathbb{R}$ is a *time-dependent transition rate matrix*, with entries $q_{ij}(t)$ for $i, j \in \Omega$ and $i \neq j$, representing the rate of transitioning from state i to state j at time t. The diagonal entries $q_{ii}(t)$ are defined such that the sum of each row in $Q(t)$ is zero, ensuring that the *outflow* from any state is equal to the sum of the *inflows* into other states. $Q(t)$ may be parameterized by hazard rates $\lambda(t|\theta)$ derived from the ratio $f(t|\theta)$ and $S(t|\theta)$, being respectively a *probability density function* and a *survival function*, where θ corresponds to the function hyper-parameters. The evolution over time of the IHTMC is governed by the *Forward Kolmogorov* equation:

$$\frac{\partial P_{ij}(t,\tau)}{\partial t} = \sum_{k\in S} P_{ik}(t,\tau) Q_{kj}(t) \tag{1}$$

Here, $P_{ij}(t, \tau) : \Omega \times \Omega \to [0, 1]$ is a continuous and differentiable function known as the *transition probability matrix*, indicating the probability of transitioning from state i to state j in the time interval t to τ, where $\tau > t$. From Eq. (1) one can obtain the *master equation of the Markov chain*, which models the flow of probabilities between states by including inflow and outflow terms:

$$\frac{\partial S_k(t)}{\partial t} = \sum_{i\in\Omega, i\neq k} S_i(t) Q_{ik}(t) - S_k(t) \Big(\sum_{j\in\Omega, j\neq k} Q_{kj}(t) \Big) \tag{2}$$

Here, $S_k(t)$ is the probability of being in state $k \in \Omega$ at time t, the term $\sum_{j\in\Omega, j\neq k} Q_{kj}(t)$ represents the rates of transition from state k to all the other states j (excluding self-transitions).

Pipe-element degradation model. We define a pipe element by K sequentially arranged states $S = [S_1, S_2, ..., S_k]$, where S_1 signifies the *pristine* condition and S_k represents the *worst condition*. This categorization is based on sewer network inspection data, which documents types of damage and their severities on a scale from 1 to 5, along with occasional instances of functional failures ($K = 6$). The transitions within our IHTMC, illustrated in Figure 1, permit only progression from a better to a worse state, prohibiting direct improvements without repairs, while allowing any severity level to escalate to functional failure.

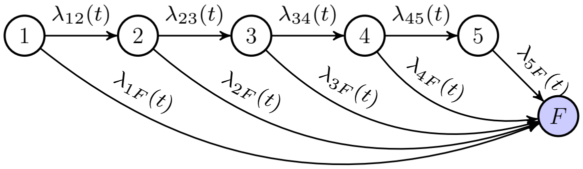


Figure 1. Markov chain structure for IHTMC.

Parametrization of IHTMC. We employed a parameterized approach for IHTMC, involving an assumption on the hazard function. In Section 4.2, we detail the parametrization used in our experimental setup. Several aspects related to the multi-state degradation model, including hyper-parameter tuning and interval-censoring, are beyond the scope of this paper. For further information, we recommend referring to (Jimenez-Roa et al., 2024).

2.2. Markov Decision Process

A Markov Decision Process (MDP) models a stochastic sequential decision process, where both costs and transition functions are dependent solely on the current state and action (Puterman, 1990). Formally, an MDP is described by the tuple $\langle \mathcal{S}, \mathcal{A}, P(s_{t+1}|s_t, a_t), \mathcal{R}(s_t, a_t, s_{t+1}), \pi_0, \gamma \rangle$, with $\mathcal{S}$ as *state space*, $\mathcal{A}$ as the *action space*, $P(s_{t+1}|s_t, a_t)$ as the *transition probability function* indicating the probability of transitioning from state s_t to s_{t+1} given action a_t, where $s_t, s_{t+1} \in \mathcal{S}$ and $a_t \in \mathcal{A}$. The *reward function* $\mathcal{R}(s_t, a_t, s_{t+1})$ specifies the reward for moving from s_t to s_{t+1} by action a_t. The *initial state* π_0 represents the distribution across $\mathcal{S}$, and $\gamma \in [0, 1]$ is the *discount factor* that balances immediate versus future rewards.

2.3. Deep Reinforcement Learning

Deep Reinforcement Learning (DRL) produces virtual agents that interact with environments to learn optimal behaviors through trial and error, as indicated by a reward signal (Arulkumaran et al., 2017). DRL has found applications in robotics, video games, and navigation systems.

We utilize DRL to train agents in virtual environments exhibiting degradation following the MSDM pattern, as detailed in Section 5. Specifically, we apply Proximal Policy Optimization (PPO) (Schulman et al., 2017), a policy gradient method in RL.

PPO aims to optimize the policy an agent uses for action selection, maximizing expected returns. It addresses stability and efficiency issues encountered in previous algorithms like *Trust Region Policy Optimization* by offering a simpler and less computationally expensive method to ensure minor policy updates.

This is achieved through an innovative objective function that penalizes significant deviations from the previous policy, fostering stable and consistent learning. The term "proximal" denotes maintaining proximity between the new and old policies, facilitating a stable training process and rendering PPO popular across various RL applications.

3. Methodology

Our methodology, illustrated in Figure 2, comprises six steps, detailed below.

Step 1. Perform data handling of historical inspection records, selecting subsets (cohorts) of interest, and calibrating the MSDM on this data. This step is beyond the scope of this paper; for details, see Jimenez-Roa et al. (2022, 2024). The results of this step are given in Section 4.

Step 2. After calibrating the MSDM, integrate these models into an environment suitable for RL applications. We present the details of our environment integrating MSDM in Section 5. In addition, we define environments for training RL agents. This is to test different MSDM hypotheses; details on this can be found in Section 6.

Step 3. Train DRL agents with PPO. Use `optuna` for hyper-parameter tuning and `Stable Baselines3` for RL implementation. Details are in Section 7.1.

Step 4. Train and select the RL agents with the optimal hyper-parameters on the *training* environments. In essence, these agents learn the dynamics described by the MSDM encoded in the environment.

Step 5. Compare the maintenance policies advised by the RL agents using the *test* environment against the heuristics: Condition-Based Maintenance (CBM), Scheduled Maintenance (SchM), and Reactive Maintenance (RM). Find the definition of these heuristics in Section 6.2.

Step 6. Analyze and compare the behavior of the maintenance strategies for the different RL models and heuristics. Reflect on the policies advantages and disadvantages. Find in Section 7.2 the overview of this comparison, and in Section 7.3 are the details along episodes.

4. Multi-state degradation models

4.1. Case study

Our case study conducts a detailed examination of the sewer pipe network in Breda, the Netherlands, which comprises 25,727 sewer pipes covering 1,052 km, mostly built after 1950. The network is primarily made of concrete (72%) and PVC (27%), with the shapes of the pipes being predominantly round (95%) and ovoid (5.4%). These pipes are designed for transportation (98.2%), with 88% being up to 60 meters in length. Additionally, 98.3% have a diameter of up to 1 meter, with the most common diameter being 0.2 meters, and they carry mixed (63%), rain (21%), and waste (16%) contents. The condition of the pipes is evaluated through visual inspections according to the European standard EN 13508 (EN13508, 2012; EN13508-2, 2011), focusing on identifying and classifying damage with specific codes. This study specifically addresses the damage code *BAF*, which signifies *surface damage* and was observed in 35.3% of the inspections.

4.2. Parametrization

We consider three distributions for hazard rate functions: Exponential, Gompertz, and Weibull. The hazard rates $\lambda(t|\cdot)$ for these distributions are specified as follows:

$$\textit{Exponential}\text{ function:}\quad \lambda^E(t|\epsilon) = \epsilon, \tag{3a}$$

$$\textit{Gompertz}\text{ function:}\quad \lambda^G(t|\alpha, \beta) = \alpha\beta e^{\beta t} \tag{3b}$$

$$\textit{Weibull}\text{ function:}\quad \lambda^W(t|\eta, \rho) = \frac{\rho}{\eta}\left(\frac{t}{\eta}\right)^{\rho-1} \tag{3c}$$

In Eq. (3a), a constant hazard rate indicates that the degradation model assumes a *homogeneous* time, exhibiting *memory-less* properties. Eq. (3b) and Eq. (3c) present varying hazard rates, which indicates *inhomogeneous* time.

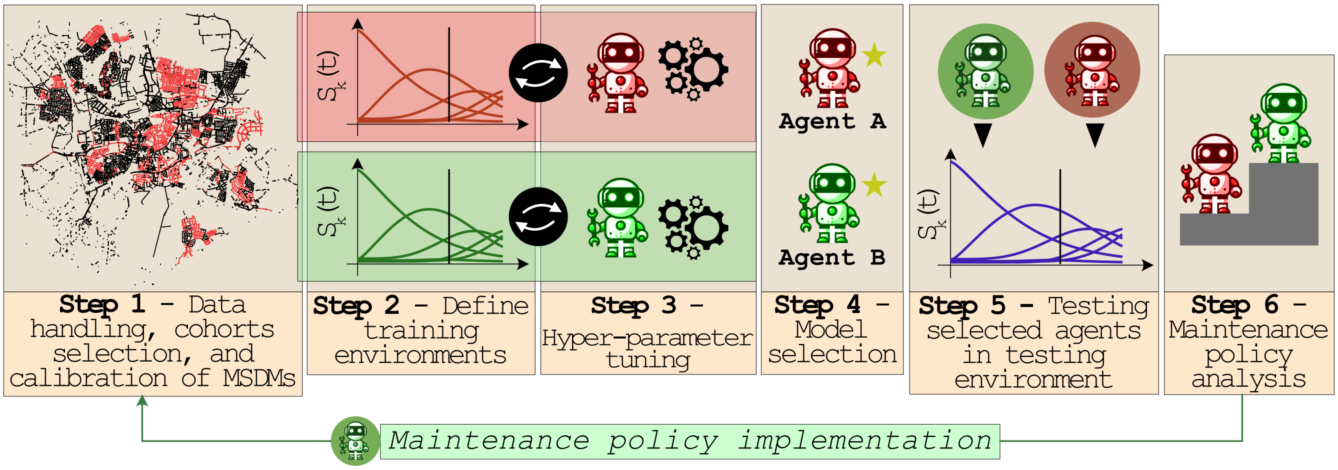

$S_k(t)$
Agent A
Agent B
Step 1 - Data handling, cohorts selection, and calibration of MSDMs
Step 2 - Define training environments
Step 3 - Hyper-parameter tuning
Step 4 - Model selection
Step 5 - Testing selected agents in testing environment
Step 6 - Maintenance policy analysis
Maintenance policy implementation

Figure 2. Methodology overview for sewer pipe maintenance policy optimization using Deep Reinforcement Learning and Multi-State Degradation models.

4.3. Solving the Multi-State Degradation Model

In Figure 1, we defined the structure of the Markov chain to model degradation in a sewer pipe, and in Section 4.2 we introduced the hazard rate functions. In the following, we present the corresponding system of differential equations.

$$\frac{\partial S_1(t)}{dt} = -\big(\lambda_{12}(t|\cdot) + \lambda_{1F}(t|\cdot)\big)S_1(t) \tag{4a}$$

$$\frac{\partial S_2(t)}{dt} = \lambda_{12}(t|\cdot)S_1(t) - \big(\lambda_{23}(t|\cdot) + \lambda_{2F}(t|\cdot)\big)S_2(t) \tag{4b}$$

$$\frac{\partial S_3(t)}{dt} = \lambda_{23}(t|\cdot)S_2(t) - \big(\lambda_{34}(t|\cdot) + \lambda_{3F}(t|\cdot)\big)S_3(t) \tag{4c}$$

$$\frac{\partial S_4(t)}{dt} = \lambda_{34}(t|\cdot)S_3(t) + \big(-\lambda_{45}(t|\cdot) - \lambda_{4F}(t|\cdot)\big)S_4(t) \tag{4d}$$

$$\frac{\partial S_5(t)}{dt} = \lambda_{45}(t|\cdot)S_4(t) - \lambda_{5F}(t|\cdot)S_5(t) \tag{4e}$$

$$\begin{aligned}\frac{\partial S_F(t)}{dt} &= \lambda_{1F}(t|\cdot)S_1(t) + \lambda_{2F}(t|\cdot)S_2(t) + \lambda_{3F}(t|\cdot)S_3(t)\\ &\quad + \lambda_{4F}(t|\cdot)S_4(t) + \lambda_{5F}(t|\cdot)S_5(t)\end{aligned} \tag{4f}$$

Eq. 4 is solved using numerical methods, specifically the `LSODA` algorithm from the FORTRAN `odepack` library implemented in SciPy (Jones et al., 2001–). This algorithm solves systems of ordinary differential equations by employing the `Adams/BDF` method with automatic stiffness detection.

4.4. Parametric Multi-State Degradation Models

We extract a subset from our case study data set to construct a cohort with concrete sewer pipes carrying *mixed and waste content* (cohort `CMW`), representing 37.1% of the sewer network. The model parameters for this cohort are detailed in Appendix A in Tables 7 and 8.

Figure 3 illustrates the MSDMs predictions, detailing the stochastic dynamics of sewer pipe degradation for pipes in cohort `CMW`. As Figure 1 describes, this degradation is segmented into five sequentially ordered severity levels ($k = 1$ to $k = 5$), plus a functional failure state ($k = F$). Differences in the y-axis scales are intentional, to emphasize details and behaviors that various degradation models express across severity levels.

Gray circles represent the frequency per severity level from the inspection dataset. Jimenez-Roa et al. (2022) details how these frequencies are computed. Vertical black lines in Figure 3 mark the last available data point for each severity level.

Additionally, Figure 3 presents the *Turnbull* non-parametric estimator, which assumes no specific distribution for survival times (Turnbull, 1976). In our context, this estimator represents the ground truth of stochastic degradation behavior in sewer pipes.

Tables 1 presents the Root Mean Square Error (RMSE) computed with respect to the Turnbull estimator, for each MSDM assumption, for cohorts `CMW`. These results show that models employing Gompertz and Weibull distributions yield smaller RMSEs compared to the one using the Exponential distribution.

Table 1. RMSE with respect Turnbull estimator, per severity level k and total RMSE, cohort: `CMW`.

	Exponential	Gompertz	Weibull
$S_{k=1}(t)$	3.38E-02	3.27E-02	3.34E-02
$S_{k=2}(t)$	7.04E-02	3.70E-02	3.57E-02
$S_{k=3}(t)$	6.27E-02	2.81E-02	4.38E-02
$S_{k=4}(t)$	4.28E-03	1.13E-02	5.06E-03
$S_{k=5}(t)$	8.33E-03	1.09E-02	3.04E-02
$S_{k=F}(t)$	9.19E-03	1.17E-02	3.62E-03
Total	4.13E-02	2.45E-02	2.96E-02

These MSDMs serve two crucial roles within our environment: first, they drive the degradation behavior of sewer pipes, effectively emulating how sewer pipes degrade over time. Second, the output from the MSDMs is incorporated as prognostic information, available to the agent to support decisions at any

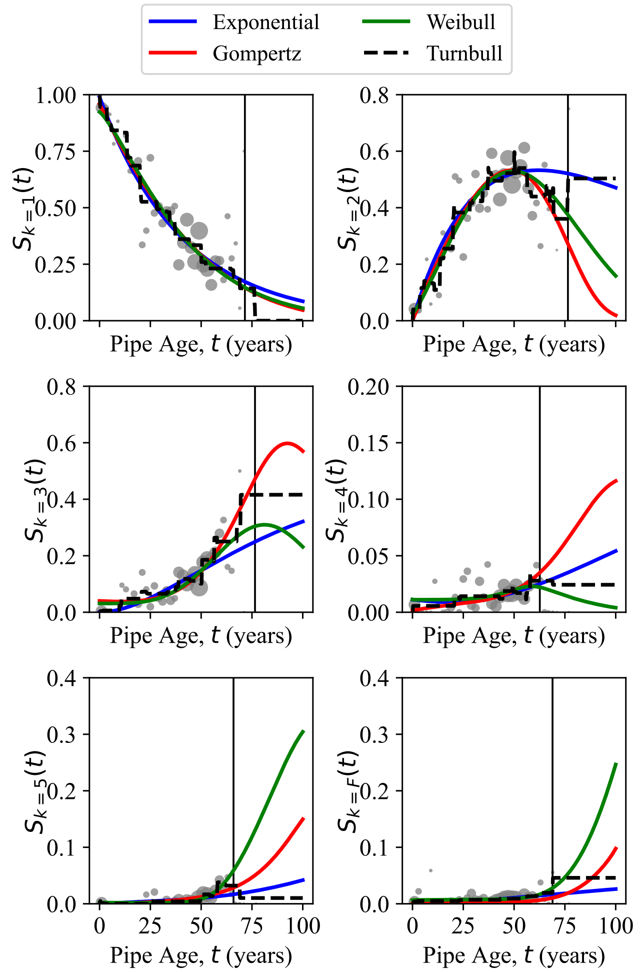


Figure 3. Probability of being in state $k \in \Omega$ at pipe age t $S_k(t)$, using three hazard functions modeled via Exponential, Gompertz, and Weibull probability density functions. The Turnbull non-parametric estimator indicates the ground truth. The gray circles indicate the frequency based on the inspection data set.

time point. This latter aspect is considered a novel feature of our framework. Details on the MDP are provided in the section below.

5. Definition of Markov Decision Process for Maintenance Policy Optimization of a Sewer Pipe considering pipe length degradation

Figure 4 provides the workflow that the RL agent uses to learn maintenance policies for sewer pipes, considering degradation along the pipe length. In the following sections, we provide the details of the environment, namely the state and action spaces, as well as the transition probability and reward functions.

5.1. State space ($\mathcal{S}$)

Our approach focuses on developing age-based maintenance policies, incorporating the sewer pipe's age into the state representation. Our state space is *continuous* and it is structured to include three key components: (i) the age of the pipe, (ii) the *health vector*, and (iii) the stochastic prediction of severity levels. We next describe the last two components.

5.1.1. Health vector (h)

In modeling the degradation of linear structures like sewer pipes, it is essential to represent changes accurately along their length. For this purpose, we define a *health vector* (**h**), which quantitatively measures the degradation at various points along the pipe. The vector is crucial in our framework, particularly influencing the reward function as described in Section 5.4.

Construction of h: We discretize the pipe into segments of equal length ΔL, with $\Delta L < L$, where L is the total length of the pipe. The number of segments, n_d, is calculated using the ceiling function to ensure it remains an integer even if L is not perfectly divisible by ΔL:

$$n_d = \left\lceil \frac{L}{\Delta L} \right\rceil \tag{5}$$

Each segment's degradation level is initially assessed and categorized into *severity levels* according to the MSDM. As the degradation progresses, the state of each segment changes following the transition probabilities described by the matrix $P_{i,j}$, where i is the current severity level, and j is the subsequent severity level, as described by the forward Kolmogorov equation (Eq. 1).

Notice that by doing this, we assume there is no statistical dependency between segments, which is a strong assumption that needs further research. However, for simplicity, we maintain this assumption in our degradation model.

Quantifying Degradation: The distribution of severity levels across the pipe is captured in vector **d**, with each element indicating the severity level of a segment. To quantify this distribution in the health vector **h**, we first count the number of segments at each severity level k using the following expression:

$$n_{d_k} = \sum_{i=1}^{n_d} \mathbf{1}_{\{\mathbf{d}_i = k\}} \tag{6}$$

where $\mathbf{1}$ is the indicator function that is 1 if the condition is true and 0 otherwise. The health vector **h** is then determined by normalizing these counts to reflect the proportion of segments at each severity level:

$$\mathbf{h}_k = \frac{n_{d_k}}{n_d} \tag{7}$$

Here, n_{d_k} is the number of segments at severity level k. Thus, $\mathbf{h}_k$ becomes part of the state space indicating the *level of degradation* present in the pipe.

5.1.2. Stochastic prediction of severity levels

To enable the agent to access information provided by the MSDM, we incorporate the prediction of severity levels into the state space. This is accomplished by solving Eq. 2, yielding a distribution $S_k(t)$.

Finally, our state space is defined as a tuple with 13 elements:

$$\mathcal{S} = \langle \text{Pipe Age}, \mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3, \mathbf{h}_4, \mathbf{h}_5, \mathbf{h}_F, S_1, S_2, S_3, S_4, S_5, S_F \rangle$$

5.2. Action space ($\mathcal{A}$)

Our action space $\mathcal{A}$ is *discrete* with dimensionality $|\mathcal{A}| = 3$. At each time step t, the agent selects an action a_t. If the decision at time t is *do nothing*, a_t is set to 0. To perform *maintenance*, a_t is set to 1, and to *replace* the pipe, a_t is set to 2. The outcomes of these actions are discussed in Section 5.3.

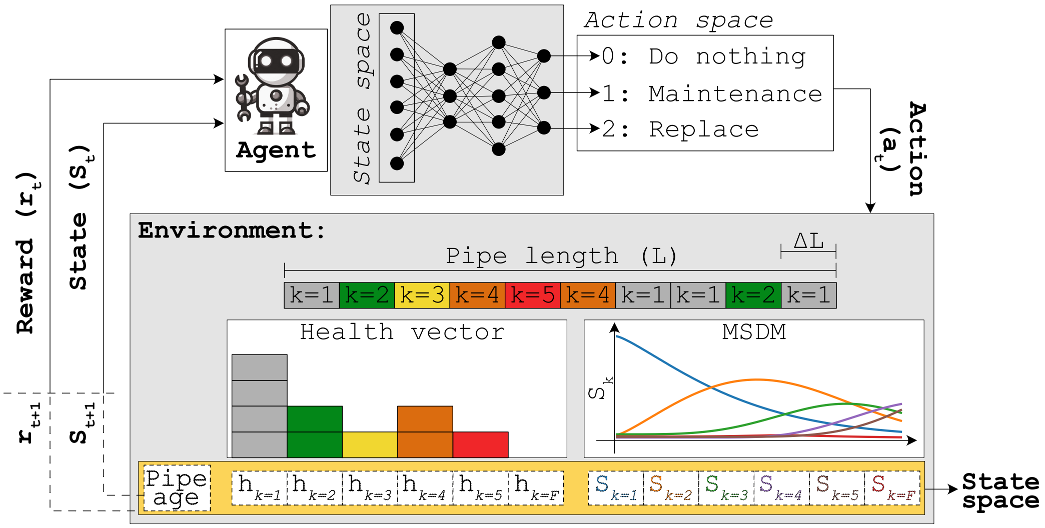


Figure 4. Environment for maintenance policy optimization of a sewer pipe via Deep Reinforcement Learning, considering degradation along the pipe length.

5.3. Transition function (P)

Our transition function $P(s_{t+1}|s_t, a_t)$ is *stochastic*, dependent on time t, and considers both the actions $a \in \mathcal{A}$ and the current s_t and next state s_{t+1} dynamics described by the MSDM. We illustrate the behavior of P with the following example.

For a 30-year-old pipe with length $L = 40$ meters and discretized in segments of length $\Delta L = 1$, let the current state space be $s_{t=30} \in \mathcal{S}$:

$$s_{t=30} = \langle 30, 0.60, 0.35, 0.025, 0.025, 0.0, 0.0, \\ 0.475, 0.436, 0.069, 0.010, 0.005, 0.005\rangle \, .$$

$s_{t=30}$ indicates the age of the pipe is 30 years. From Eq. 7, the number of segments at severity k is determined by multiplying the health vector ($\mathbf{h}_k$):

$$\mathbf{h}_k = [0.60, 0.35, 0.025, 0.025, 0.0, 0.0]$$

by 40 meters, yielding $n_{d_k} = [24, 14, 1, 1, 0, 0]$, indicating that, out of the 40 meters of pipe length, 24 segments of 1 meter are at severity $k = 1$, 14 at severity $k = 2$, and so forth.

The distribution $S_k(t = 30.0)$ predicts the probability of being in a severity level k at age $t = 30$. This is achieved by evaluating $t = 30.0$ in the corresponding MSDM.

$$S_k(t = 30.0) = [0.475, 0.436, 0.069, 0.010, 0.005, 0.005]$$

Assuming the agent takes an action every half year, we illustrate the effect of each action in $\mathcal{A}$ below.

- If $a_t = 0$: the agent decides to "do nothing", the pipe's degradation evolves in line with the MSDM progression. Here the new state space becomes $s^{a=0}_{t=30.5}$.

$$s^{a=0}_{t=30.5} = \langle 30.5, 0.575, 0.35, 0.05, 0.025, 0.0, 0.0, \\ 0.470, 0.439, 0.071, 0.010, 0.05, 0.05\rangle$$

Notice that the pipe age increased to 30.5, and $n_{d_k} = [23, 14, 2, 1, 0, 0]$, where a segment with severity $k = 1$ progressed to $k = 2$, and one segment with $k = 2$ advanced to $k = 3$. Additionally, $S_k(t)$ is updated by evaluating $t = 30.5$.

- If $a_t = 1$: the agent decides to "perform maintenance," all damage points with severity levels $k \in \{3, 4, 5\}$ are moved to $k = 2$. Consequently, this action does not affect damage points with severity levels $k \in \{1, 2, F\}$. The new state space becomes $s^{a=1}_{t=30.5}$.

$$s^{a=1}_{t=30.5} = \langle 30.5, 0.60, 0.40, 0.0, 0.0, 0.0, 0.0, \\ 0.47, 0.439, 0.071, 0.010, 0.05, 0.05\rangle$$

Notice that the pipe age increased to 30.5, and $n_{d_k} = [24, 16, 0, 0, 0, 0]$. However, $S_k(t)$ is updated by evaluating $t = 30.5$, same as when $a_t = 0$.

- If $a_t = 2$: the agent decides to "replace" the pipe, resetting its condition to as good-as-new. The new state space is $s^{a=2}_{t=0.0}$:

$$s^{a=2}_{t=0.0} = \langle 0.0, 1.0, 0.0, 0.0, 0.0, 0.0, 0.0, \\ 0.986, 0.014, 0.0, 0.0, 0.0, 0.0\rangle .$$

The pipe age is reset to 0.0, with $n_{d_k} = [40, 0, 0, 0, 0, 0]$, and $S_k(t)$ is updated for $t = 0.0$.

5.4. Reward function ($\mathcal{R}$)

Our reward function $\mathcal{R}(s_t, a_t, s_{t+1})$ assigns a reward r_t at every decision point t, determined by the current state s_t and action a_t. This function integrates the costs of maintenance (C_M), replacement (C_R), and failures (C_F). $\mathcal{R}$ is *sparse* because it issues a non-zero value only when failures occur or interventions are undertaken.

Maintenance cost C_M is calculated as per Eq. 8, where it combines a variable cost based on severity k with a fixed logistic cost of €500, covering the expenses related to maintenance.

These costs vary with the severity level k, as detailed in Table 2. Note that no maintenance costs are associated with $k = F$ because maintenance cannot be performed on a segment that has already failed. In this case, the agent must replace.

$$C_M = -(\mathbf{h}_k \cdot c_M^k + 500) \tag{8}$$

Table 2. Maintenance costs per severity k per segment (c_M^k)

	$k=1$	$k=2$	$k=3$	$k=4$	$k=5$	$k=F$
$c_M^k =$	0	0	-€500	-€700	-€900	N.A.

Replacement costs (C_R) is computed with Eq. 9:

$$C_R = -(450 + 0.66D + 0.0008D^2)L \tag{9}$$

Here, L and D denote the pipe's length in meters and diameter in millimetres, respectively. C_R is in Euros (€).

The cost of failure, denoted by C_F, entails assigning a substantial penalty when the agent allows a segment of the pipe to achieve a failure state ($k = F$). This penalty cost is established at €-100,000. Our reward function is then:

$$r_t = \frac{C_M + C_R + C_F}{100'000 + 900 \times 40} = \frac{C_M + C_R + C_F}{136'000} \tag{10}$$

where r_t represents the reward obtained at time t, the normalization constant $136'000$ corresponds to the most expensive penalty possible at time t. Thus, r_t is defined within the interval $[-1, 0]$. This reward function aims for the agent to balance maintenance actions with the prevention of undesirable pipe conditions.

6. Experimental setup

6.1. Setup

We will evaluate our framework with a single pipe of constant length (40 meters) and diameter (200 mm) from the cohort `CMW`, which carries mixed and waste content. Given the constant dimensions, the replacement cost C_R, as defined in Eq. 9, is €24,560. The pipe age, when initializing the episode, is randomly sampled from the uniform distribution $U \sim [0, 50]$, allowing the agent to learn the behavior of pipes within this age range. Additionally, we evaluate the policy in steps of half a year and $\Delta L = 1$ meter.

In the methodology section, we describe the training of two agents: **Agent-E** and **Agent-G**. **Agent-E** is trained in an environment where sewer pipe degradation follows the MSDM parameterised with an *Exponential* probability density function, while **Agent-G** is trained in an environment where degradation follows the MSDM parameterised with a *Gompertz* probability density function.

Both agents are tested in an environment where sewer pipe degradation follows the MSDM parameterized with the *Weibull* probability density function.

During training, each agent follows a specific state space, defined as follows:

$$\mathcal{S}_{\text{Training}}^{\textbf{Agent-E}} = \langle \text{Pipe Age}, \mathbf{h}_k^E, S_k^E(t) \rangle \tag{11a}$$

$$\mathcal{S}_{\text{Training}}^{\textbf{Agent-G}} = \langle \text{Pipe Age}, \mathbf{h}_k^G, S_k^G(t) \rangle \tag{11b}$$

Here, $\mathcal{S}$ represents the state space for each agent during training. The subscripts E and G denote the *Exponential* and *Gompertz* probability density functions, respectively. Each agent's objective is to learn an optimal maintenance strategy based on their environment's dynamics.

For testing, both agents are evaluated in the same environment, with the state space defined as follows:

$$\mathcal{S}_{\text{Testing}}^{\textbf{Agent-E}} = \langle \text{Pipe Age}, \mathbf{h}_k^W, S_k^E(t) \rangle \tag{12a}$$

$$\mathcal{S}_{\text{Testing}}^{\textbf{Agent-G}} = \langle \text{Pipe Age}, \mathbf{h}_k^W, S_k^G(t) \rangle \tag{12b}$$

In both cases, $S_k^E(t)$ and $S_k^G(t)$ remain consistent with the training phase, reflecting the MSDM predictions. However, the health vector $\mathbf{h}_k$ follows the degradation behavior described by the *Weibull* probability density function, indicated by the subscript W.

6.2. Comparison of maintenance strategies

We compare the RL agent's performance against maintenance policies based on heuristics. For this, we define the following:

- **Condition-Based Maintenance (CBM)**: Maintenance actions are based on the sewer pipe's condition. Specifically, replacement ($a_t = 2$) is performed if `pipe_age` ≥ 70 or $\mathbf{h}_{k=F} \geq 0.0$; maintenance ($a_t = 1$) is conducted if $\mathbf{h}_{k=4} \geq 0.1$ or $\mathbf{h}_{k=5} \geq 0.05$; otherwise, no action ($a_t = 0$) is taken.
- **Scheduled Maintenance (SchM)**: Actions are time-based. Replacement ($a_t = 2$) is executed if $\mathbf{h}_{k=F} \geq 0.0$; maintenance ($a_t = 1$) occurs every 10 years; otherwise, no action ($a_t = 0$) is taken.
- **Reactive Maintenance (RM)**: Replacement is undertaken only upon pipe failure, i.e., replacement ($a_t = 2$) is performed if $\mathbf{h}_{k=F} \geq 0.0$; otherwise, no action ($a_t = 0$) is taken.

Note that CBM and SchM are defined based on plausible values. However, these heuristics can be further calibrated for enhanced performance, which is beyond the scope of this paper.

7. Results

7.1. Implementation and hyper-parameter tuning

Our framework uses `Stable Baselines3` (Raffin et al., 2021), comprising robust implementations of RL algorithms in PyTorch (Ansel et al., 2024). Specifically, we utilize the PPO algorithm. Hyper-parameter optimization is performed using `optuna` (Akiba et al., 2019), a framework dedicated to automating the optimization of hyper-parameters.

The search space encompasses: exponentially-decaying learning rate with a decay rate of 0.05, with an initial learning rate ranging from 10^{-5} to 10^{-2}, discount factor (γ) from 0.8 to 0.9999, entropy coefficient from 0.0001 to 0.01, steps per update (`n_steps`) from 250 to 3000, batch sizes from 16 to 256, activation functions ('tanh', 'relu', 'sigmoid'), policy network architectures ([16, 16], [32, 32], [64, 64], [32, 32, 32]), and training epochs (`n_epochs`) from 5 to 100.

We set up `optuna` to conduct 500 trials, aiming to maximise cumulative reward in 100 episodes. Table 3 details the optimal hyper-parameters identified. These parameters are used to obtain the results discussed in Sections 7.2 and 7.3, where our agents are trained over a total of 5 million time steps.

Table 3. Optimal hyper-parameters found using `optuna`.

Hyper-parameter	Value
Learning rate	0.0003
Discount factor	0.995
Entropy coefficient	0.008
Steps per update (`n_steps`)	2080
Batch size	104
Activation function	Sigmoid
Policy network architecture	[32, 32, 32]
Training epochs (`n_epochs`)	50

7.2. Policy analysis: overview

This section offers a broad evaluation of the policies, with a detailed analysis over episodes presented in Section 7.3. We compare the agents' performances with the heuristics detailed in Section 6.2 across 100 simulations in the **test** environment (Eq. 12), considering pipe ages of 0, 25, and 50 years, aiming to evaluate policy efficacy concerning degradation over varying pipe ages.

Table 4 presents the *mean policy cost* for Agent-E, Agent-G, CBM, SchM, and RM, highlighting the best and second-best policies in blue and red, with corresponding means and standard deviations from the simulations.

Table 4. Policy cost comparison: Mean and standard deviation (Std.) of costs for Agent-E, Agent-G, CBM, SchM, and RM, evaluated over 100 episodes in the test environment. Costs, in thousands of Euros (€), for pipe ages of 0, 25, and 50 years.

	Pipe age: 0		Pipe age: 25		Pipe age: 50	
Policy	Mean	Std.	Mean	Std.	Mean	Std.
Agent-E	51.3	80.8	116.5	97.7	156.8	121.2
Agent-G	**39.7**	66.2	**78.7**	96.6	**127.1**	128.3
CBM	51.3	107.2	112.3	88.5	**110.7**	86.6
SchM	**42.5**	70.9	**78.9**	96.4	159.8	95.9
RM	48.6	76.6	135.8	86.5	165.7	80.8

From these results, we observe that Agent-G's policy generally outperforms others for pipe ages of 0 and 25 years, securing a second-best position for pipes aged 50 years. It is noted that the cost of all policies increases with pipe age, which aligns with expectations as older pipes require more interventions.

After reviewing the mean policy costs, our focus shifts to the specific actions involved in each policy. Table 5 provides a summary of the actions executed by each policy across simulations for different pipe ages. For new pipes, the SchM policy leads in maintenance activities ($a_t = 1$), with Agent-G following. In terms of replacements ($a_t = 2$), Agent-E is the foremost in implementing this action, with CMB in second place. Both Agent-G and SchM exhibit lower replacement frequencies, explaining the mean policy costs since maintenance actions incur lower expenses compared to the penalties and replacement costs resulting from pipe failures.

Table 5. Percentage of actions per policy obtained with Agent-E, Agent-G, CBM, SchM, and RM, evaluated over 100 episodes in the test environment, for different pipe ages.

Pipe age	Action	Agent-E	Agent-G	CBM	SchM	RM
0	$a_t = 0$	99.5	97.51	99.54	94.76	99.61
	$a_t = 1$	0.0	2.21	0.05	4.95	0.00
	$a_t = 2$	0.5	0.28	0.41	0.29	0.39
25	$a_t = 0$	98.81	94.96	98.14	94.56	98.92
	$a_t = 1$	0.00	4.50	0.62	4.94	0.00
	$a_t = 2$	1.19	0.53	1.24	0.50	1.08
50	$a_t = 0$	98.4	94.52	98.05	93.99	98.68
	$a_t = 1$	0.0	4.43	0.67	4.88	0.00
	$a_t = 2$	1.6	1.05	1.28	1.13	1.32

For pipes aged 25 years, Agent-G executes more maintenance actions ($a_t = 1$), similar to SchM. Agent-E opts for no maintenance, aligning more with RM's strategy. Although CBM carries out some maintenance actions, replacement actions predominate, indicating a greater tendency to permit pipe failures, which explains the observed differences in mean policy costs.

For pipes aged 50 years, CMB offers the most cost-effective policy, with Agent-G's following. CMB conducts fewer maintenance actions and more replacements than Agent-G, accounting for the cost disparity. The policies of Agent-E, RM, and SchM have similar costs. Despite SchM conducting more maintenance, its high number of replacements suggests the maintenance interval requires adjustment. These results indicate that the strategies of CBM, SchM, and RM are less efficient for older pipes due to their higher failure probability.

Regarding the *mean pipe severity level* to assess the impact of various policies on pipe degradation, as shown in Table 6. Our analysis reveals a notable correlation between the average actions per policy, detailed in Table 5, and the mean pipe severity level. Specifically, the Agent-G control strategy tends to maintain pipes within a severity level of $k \in [1, 2, 3]$, whereas the Agent-E, CBM, SchM, and RM policies often result in higher severity levels $k \in [4, 5, F]$, which correlates with increased policy costs.

Table 6. Percentage of severity level per policy obtained with Agent-E, Agent-G, CBM, SchM, and RM, evaluated over 100 episodes in the test environment, for different pipe ages.

Pipe age	Severity	Agent-E	Agent-G	CBM	SchM	RM
0	$k = 1$	59.77	58.75	59.94	59.84	58.88
	$k = 2$	33.27	39.14	32.67	38.05	33.15
	$k = 3$	5.39	1.70	6.00	1.79	6.36
	$k = 4$	1.38	0.28	1.13	0.26	1.30
	$k = 5$	0.18	0.13	0.25	0.04	0.31
	$k = F$	0.01	0.01	0.01	0.01	0.01
25	$k = 1$	50.49	41.72	46.88	39.07	46.62
	$k = 2$	38.96	55.27	43.09	55.55	40.86
	$k = 3$	8.37	2.63	8.48	4.85	9.80
	$k = 4$	1.37	0.29	1.18	0.41	1.51
	$k = 5$	0.78	0.07	0.36	0.10	1.18
	$k = F$	0.02	0.01	0.02	0.01	0.03
50	$k = 1$	57.93	44.65	55.01	40.92	54.36
	$k = 2$	32.58	51.40	36.14	50.46	33.09
	$k = 3$	7.50	3.29	7.20	7.34	9.32
	$k = 4$	1.31	0.39	1.19	0.59	1.64
	$k = 5$	0.65	0.25	0.43	0.67	1.55
	$k = F$	0.03	0.02	0.02	0.03	0.03

To summarize, our findings indicate that the Agent-G's policy, derived using DRL, implements a dynamic management strategy that varies with the pipe's age. This strategy encompasses a more passive approach with new pipes, transitioning to active intervention as the pipes age. This indicates the agent's preference for more frequent maintenance actions rather than allowing pipe failures, which incur higher penalties and replacement costs.

Moreover, Agent-G outperforms Agent-E, illustrating the impact of the degradation model assumption. Specifically, Agent-G's prognostic model used during training aligns more closely with the test environment's degradation pattern than Agent-E's, potentially explaining why Agent-G is better equipped to navigate and understand the degradation pattern. This, in turn, enables it to devise a more effective maintenance policy by leveraging a more accurate degradation model.

7.3. Policy analysis over episode

In Section 7.2, we present an overview of policy performances. This section delves into the details per episode to provide further understanding on these policies. Figures 5, 6, and 7 detail the performance of the Agent-E, Agent-G, CMB, and SchM policies for pipes with ages 0, 25 and 50, respectively. The RM heuristic is excluded from this analysis due to its straightforward approach: allowing the pipe to fail before replacing it.

Figure 5 shows that for a brand new pipe: (a) Agent-G performs maintenance on the pipe at approximately 32 years old; (b) Agent-E opts to replace the pipe when it is around 35 years old, which may be attributed to the presence of elements with higher severity levels in that specific episode; (c) CBM chooses not to act, which results in the least expensive policy in this comparison. However, it is observed that some pipe sections reach severity level $k = 5$ throughout the episode. Not taking any action is deemed risky since progressing to $k = F$ becomes more likely and incurs higher costs; (d) SchM effectively controls severity levels but is more expensive than Agent-G's policy due to more frequent maintenance actions.

Figure 6 shows that for a pipe aged 25: (a) Agent-G exhibits increased activity, indicating more frequent maintenance actions, especially as the pipe ages to 50, shortening the maintenance intervals; (b) Agent-E postpones any action until the pipe fails, at which point it replaces the pipe with a new one, akin to RM; (c) CBM also initiates maintenance around the pipe's 50-year mark. However, degradation escalates from age 60, leading to failure at 66. The inability to manage this increased severity results in significant penalty costs, diminishing the effectiveness of this policy; (d) Similarly, SchM manages severity levels effectively until the pipe reaches approximately 70 years of age, at which point degradation accelerates, resulting in failure at 73.

Figure 7 shows that for a pipe aged 50: (a) Agent-G opts to replace the pipe at age 50, followed by maintenance in the subsequent time step. This decision is likely influenced by parts of the pipe being at severity levels $k \in 3, 4$. Such a scenario is plausible, as new pipes can exhibit high severity levels at a young age due to defects in the material or errors during the construction and installation process. This concept is represented in the MSDM by the initial probability state vector (S_k^0). Additionally, Agent-G recommends maintenance at the interval when the pipe reaches the age of 26 years; (b) Agent-E suggests replacement at approximately 62 years, without recommending further maintenance; (c) CMB advocates for maintenance at about 65 years, followed by replacement at 70 years, in line with heuristics described in Section 6.2; (d) SchM consistently performs maintenance at regular intervals, yet faces significant degradation, culminating in failure around 97 years.

8. Discussion and Conclusions

In this paper, we explore the applications of Prognostics and Health Management (PHM) in sewer pipe asset management. Our study focuses on component-level (i.e., pipe-level) maintenance policy optimization by integrating stochastic multi-state degradation modeling and Deep Reinforcement Learning (DRL). The goal is to assess the effectiveness of DRL in deriving cost-effective maintenance strategies tailored to the specific conditions and requirements of sewer pipes.

A key contribution of our work is the integration of prognostics models with a maintenance policy optimization framework. We utilize a tailored reward function that aligns with damage severity levels, enabling a more complex and realistic maintenance optimization setup.

Our methodology includes a real-world case study from a Dutch sewer network, which provides historical inspection data. Through hyper-parameter tuning and policy analysis, we benchmark our optimized policies against traditional heuristics, including condition-based, scheduled, and reactive maintenance.

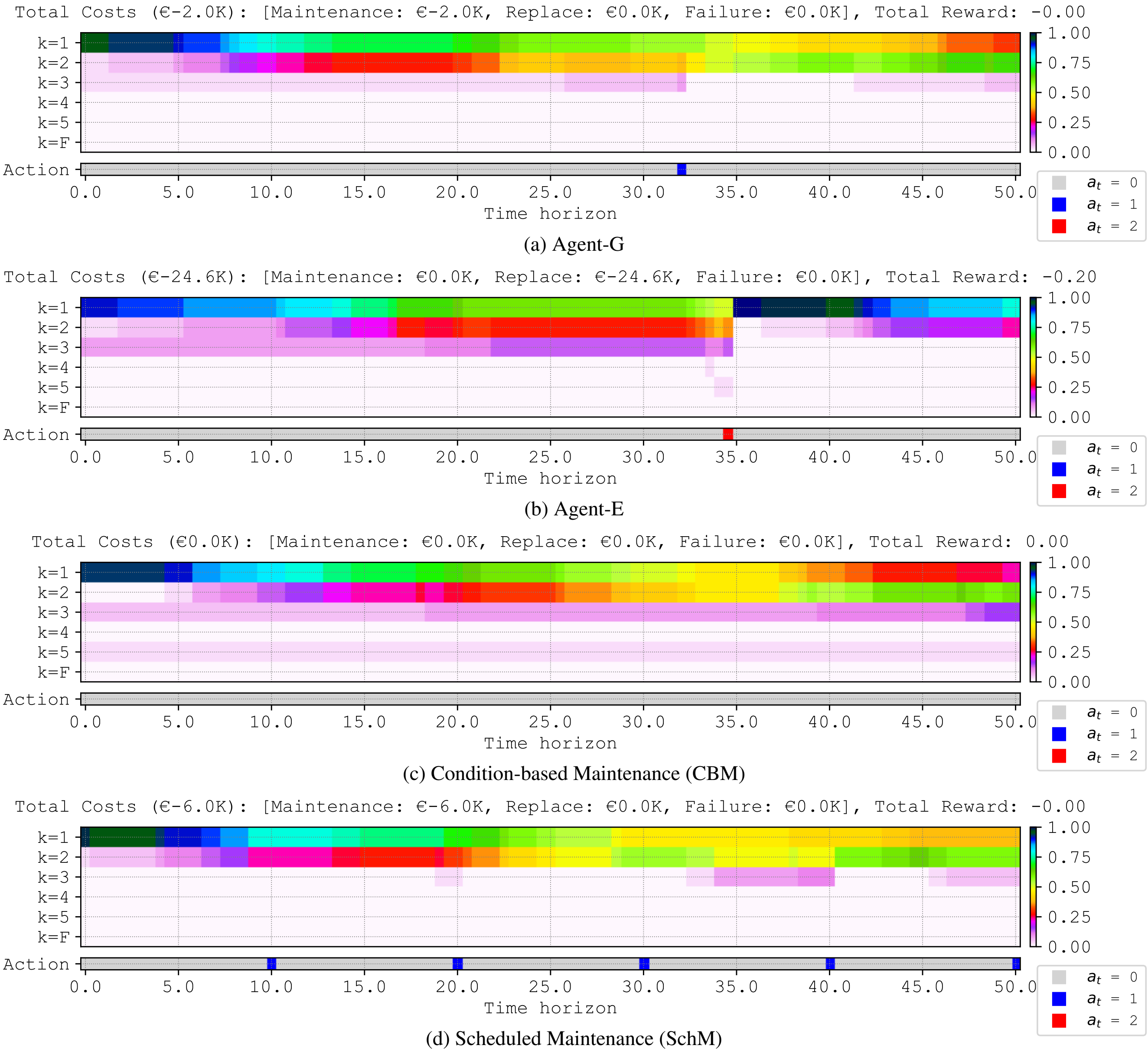
Our findings suggest that agents trained with the Proximal Policy Optimization algorithm are highly capable of developing strategic maintenance policies, adapting to pipe age, and surpassing heuristic baselines by learning cost-effective dynamic management strategies.

To evaluate the impact of degradation model assumptions, we trained one agent using the Gompertz probability density function and another using the Exponential probability density function.

During testing, both agents were assessed in an environment parameterized with the Weibull probability density function. The Gompertz-trained agent, whose behavior more closely resembled the Weibull model, demonstrated better generalization, resulting in more effective maintenance policies compared to the Exponential-trained agent.

Future work: The following directions are identified:

- Advancing toward partially observable state spaces with the introduction of inspection actions, considering context, and leveraging deep learning capabilities.
- Utilizing knowledge acquired by agents to develop explainable and robust heuristics.
- Although this paper focused on a single cohort of pipes, studies in Jimenez-Roa et al. (2022, 2024) show different cohorts exhibit varied dynamics, highlighting the importance of understanding how RL agents adapt.
- Comparing RL-based approaches with other policy optimization algorithms to better understand the capacity of RL methods to achieve global-optima maintenance strategies.
- Investigating various reward functions (e.g., dense) and RL algorithms to determine the most effective for devising maintenance policies.
- Extent to system-level analysis and evaluate aspects such



(a) Agent-G

(b) Agent-E

(c) Condition-based Maintenance (CBM)

(d) Scheduled Maintenance (SchM)

Figure 5. Behavior of policies over an episode for a **new pipe**, showing the health vector over the pipe age and actions per policy: (a) Agent-G, (b) Agent-E, (c) Condition-based Maintenance (CBM), and (d) Scheduled Maintenance (SchM).

as scalability.

- Moving toward multi-infrastructure asset management to promote coordinated management for optimizing costs and minimizing disruption from interventions.

Acknowledgements

This research has been partially funded by NWO under the grant PrimaVera (https://primavera-project.com) number NWA.1160.18.238.

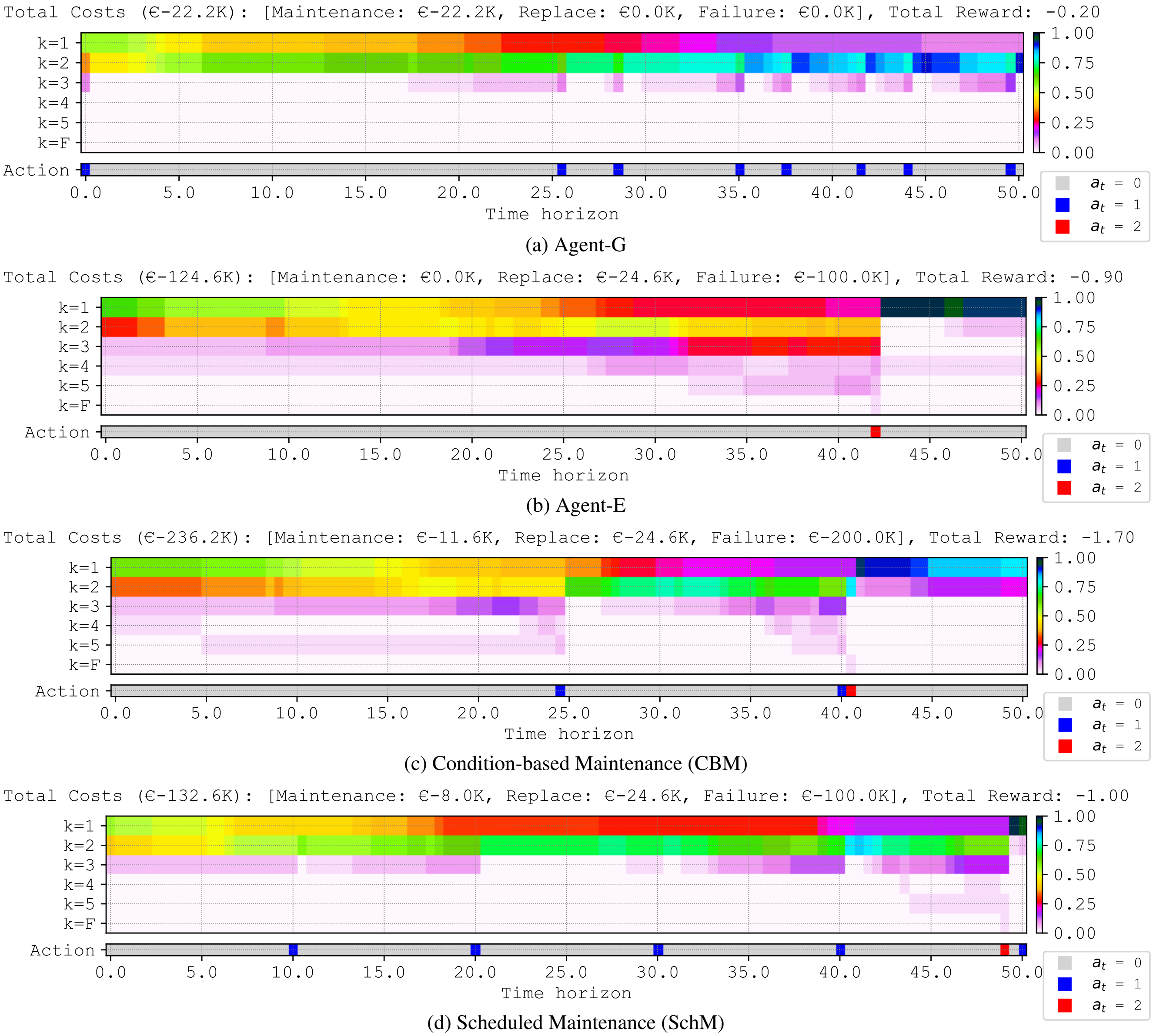
References

Abraham, D. M., Wirahadikusumah, R., Short, T., & Shahbahrami, S. (1998). Optimization modeling for sewer network management. *Journal of construction engineering and management*, *124*(5), 402–410.

Akiba, T., Sano, S., Yanase, T., Ohta, T., & Koyama, M. (2019). Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD international conference on knowledge discovery and data mining.*

Ana, E., & Bauwens, W. (2010). Modeling the structural deterioration of urban drainage pipes: the state-of-the-art in statistical methods. *Urban Water Journal*, *7*(1), 47–59.

Ansel, J., Yang, E., He, H., Gimelshein, N., Jain, A., Voznesensky, M., … others (2024). Pytorch 2: Faster machine



(a) Agent-G

(b) Agent-E

(c) Condition-based Maintenance (CBM)

(d) Scheduled Maintenance (SchM)

Figure 6. Behavior of policies over an episode for a **pipe aged 25**, showing the health vector over the pipe age and actions per policy: (a) Agent-G, (b) Agent-E, (c) Condition-based Maintenance (CBM), and (d) Scheduled Maintenance (SchM).

learning through dynamic python bytecode transformation and graph compilation. In *Proceedings of the 29th acm international conference on architectural support for programming languages and operating systems, volume 2* (pp. 929–947).

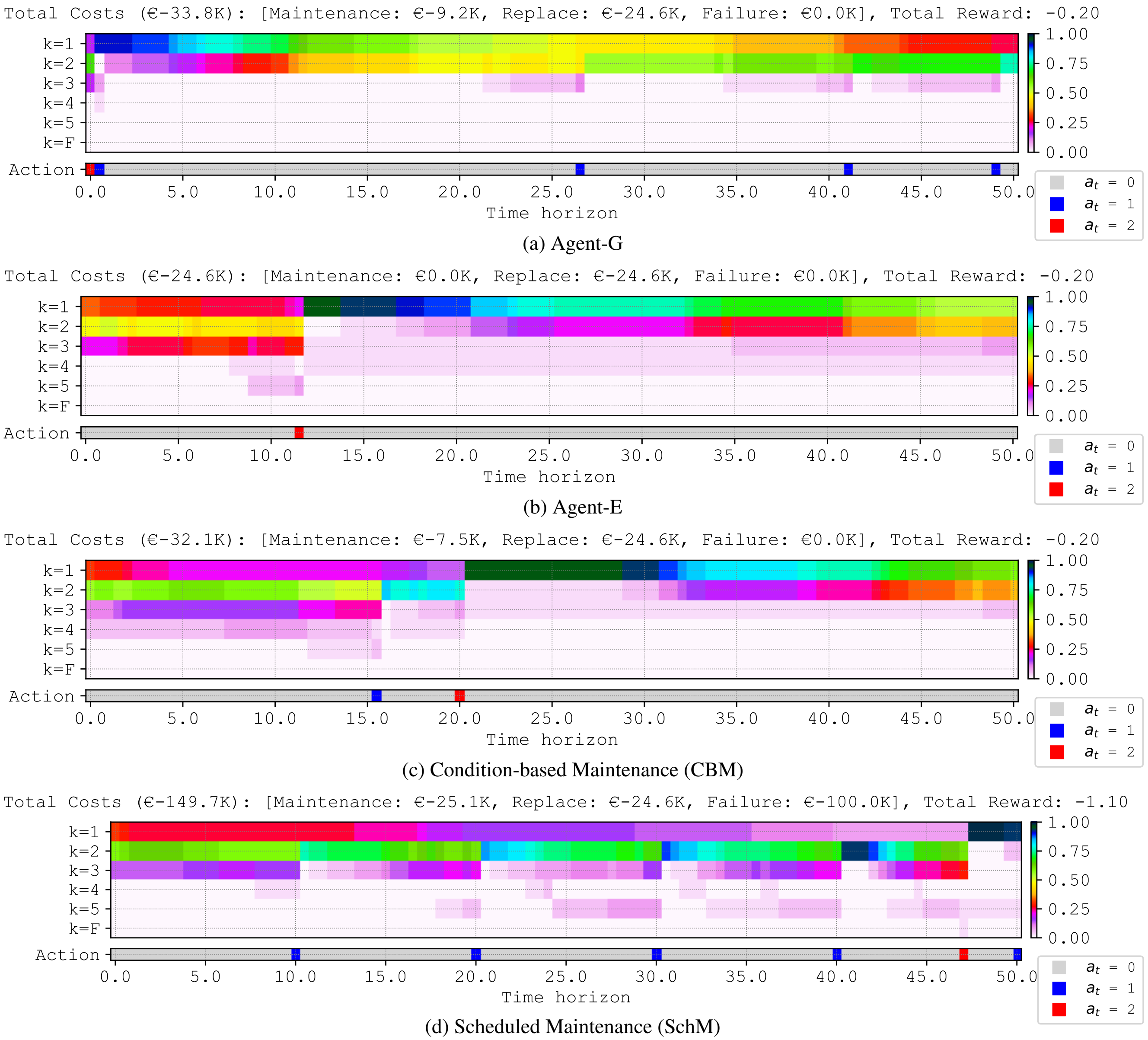
Arulkumaran, K., Deisenroth, M. P., Brundage, M., & Bharath, A. A. (2017). Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, *34*(6), 26–38.

Assaf, G., & Assaad, R. H. (2023). Optimal preventive maintenance, repair, and replacement program for catch basins to reduce urban flooding: Integrating agent-based modeling and monte carlo simulation. *Sustainability*, *15*(11), 8527.

Caradot, N., Riechel, M., Fesneau, M., Hernandez, N., Torres, A., Sonnenberg, H., . . . Rouault, P. (2018). Practical benchmarking of statistical and machine learning models for predicting the condition of sewer pipes in berlin, germany. *Journal of Hydroinformatics*, *20*(5), 1131–1147.

Cardoso, M., Almeida, M. d. C., & Santos Silva, M. (2016). Sewer asset management planning–implementation of a structured approach in wastewater utilities. *Urban Water Journal*, *13*(1), 15–27.

De Jonge, B., & Scarf, P. A. (2020). A review on maintenance optimization. *European journal of operational research*, *285*(3), 805–824.



(a) Agent-G

(b) Agent-E

(c) Condition-based Maintenance (CBM)

(d) Scheduled Maintenance (SchM)

Figure 7. Behavior of policies over an episode for a **pipe aged 50**, showing the health vector over the pipe age and actions per policy: (a) Agent-G, (b) Agent-E, (c) Condition-based Maintenance (CBM), and (d) Scheduled Maintenance (SchM).

Elmasry, M., Zayed, T., & Hawari, A. (2019). Multi-objective optimization model for inspection scheduling of sewer pipelines. *Journal of Construction Engineering and Management*, *145*(2), 04018129.

Fenner, R. A. (2000). Approaches to sewer maintenance: a review. *Urban water*, *2*(4), 343–356.

Hawari, A., Alkadour, F., Elmasry, M., & Zayed, T. (2017). Simulation-based condition assessment model for sewer pipelines. *Journal of Performance of Constructed Facilities*, *31*(1), 04016066.

Hernández, N., Caradot, N., Sonnenberg, H., Rouault, P., & Torres, A. (2021). Optimizing svm models as predicting tools for sewer pipes conditions in the two main cities in colombia for different sewer asset management purposes. *Structure and Infrastructure Engineering*, *17*(2), 156–169.

Investigation and assessment of drain and sewer systems outside buildings - Part 1: General Requirements (Standard). (2012, October). Avenue Marnix 17, B-1000 Brussels: European Committee for Standardization (CEN).

Investigation and assessment of drain and sewer systems outside buildings - Part 2: Visual inspection coding system (Standard). (2011, May). Avenue Marnix 17, B-1000 Brussels: European Committee for Standardization (CEN).

Jeung, M., Jang, J., Yoon, K., & Baek, S.-S. (2023). Data assimilation for urban stormwater and water quality simulations using deep reinforcement learning. *Journal of Hydrology*, *624*, 129973.

Jimenez-Roa, L. A., Heskes, T., Tinga, T., Molegraaf, H. J., & Stoelinga, M. (2022). Deterioration modeling of sewer pipes via discrete-time markov chains: A large-scale case study in the netherlands. In *32nd european safety and reliability conference, esrel 2022: Understanding and managing risk and reliability for a sustainable future* (pp. 1299–1306).

Jimenez-Roa, L. A., Tinga, T., Heskes, T., & Stoelinga, M. (2024). Comparing homogeneous and inhomogeneous time markov chains for modelling degradation in sewer pipe networks. In *Proceedings of the european safety and reliability conference (esrel 2024).* (Under review)

Jones, E., Oliphant, T., Peterson, P., et al. (2001–). *SciPy: Open source scientific tools for Python.* Retrieved from http://www.scipy.org/

Kerkkamp, D., Bukhsh, Z. A., Zhang, Y., & Jansen, N. (2022). Grouping of maintenance actions with deep reinforcement learning and graph convolutional networks. In *Icaart (2)* (pp. 574–585).

Khurelbaatar, G., Al Marzuqi, B., Van Afferden, M., Müller, R. A., & Friesen, J. (2021). Data reduced method for cost comparison of wastewater management scenarios–case study for two settlements in jordan and oman. *Frontiers in Environmental Science*, *9*, 626634.

Laakso, T., Kokkonen, T., Mellin, I., & Vahala, R. (2019). Sewer life span prediction: Comparison of methods and assessment of the sample impact on the results. *Water*, *11*(12), 2657.

Lee, J., Park, C. Y., Baek, S., Han, S. H., & Yun, S. (2021). Risk-based prioritization of sewer pipe inspection from infrastructure asset management perspective. *Sustainability*, *13*(13), 7213.

Malek Mohammadi, M., Najafi, M., Kaushal, V., Serajiantehrani, R., Salehabadi, N., & Ashoori, T. (2019). Sewer pipes condition prediction models: A state-of-the-art review. *Infrastructures*, *4*(4), 64.

Marugán, A. P. (2023). Applications of reinforcement learning for maintenance of engineering systems: A review. *Advances in Engineering Software*, *183*, 103487.

Montserrat, A., Bosch, L., Kiser, M., Poch, M., & Corominas, L. (2015). Using data from monitoring combined sewer overflows to assess, improve, and maintain combined sewer systems. *Science of the Total Environment*, *505*, 1053–1061.

Mullapudi, A., Lewis, M. J., Gruden, C. L., & Kerkez, B. (2020). Deep reinforcement learning for the real time control of stormwater systems. *Advances in water resources*, *140*, 103600.

Ogunfowora, O., & Najjaran, H. (2023). Reinforcement and deep reinforcement learning-based solutions for machine maintenance planning, scheduling policies, and optimization. *Journal of Manufacturing Systems*, *70*, 244–263.

Puterman, M. L. (1990). Markov decision processes. *Handbooks in operations research and management science*, *2*, 331–434.

Qasem, A., & Jamil, R. (2021). Gis-based financial analysis model for integrated maintenance and rehabilitation of underground pipe networks. *Journal of Performance of Constructed Facilities*, *35*(5), 04021046.

Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., & Dormann, N. (2021). Stable-baselines3: Reliable reinforcement learning implementations. *The Journal of Machine Learning Research*, *22*(1), 12348–12355.

Ramos-Salgado, C., Muñuzuri, J., Aparicio-Ruiz, P., & Onieva, L. (2022). A comprehensive framework to efficiently plan short and long-term investments in water supply and sewer networks. *Reliability Engineering & System Safety*, *219*, 108248.

Saddiqi, M. M., Zhao, W., Cotterill, S., & Dereli, R. K. (2023). Smart management of combined sewer overflows: From an ancient technology to artificial intelligence. *Wiley Interdisciplinary Reviews: Water*, *10*(3), e1635.

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Taillandier, F., Elachachi, S., & Bennabi, A. (2020). A decision-support framework to manage a sewer system considering uncertainties. *Urban Water Journal*, *17*(4), 344–355.

Tian, W., Fu, G., Xin, K., Zhang, Z., & Liao, Z. (2024). Improving the interpretability of deep reinforcement learning in urban drainage system operation. *Water Research*, *249*, 120912.

Tian, W., Liao, Z., Zhi, G., Zhang, Z., & Wang, X. (2022). Combined sewer overflow and flooding mitigation through a reliable real-time control based on multi-reinforcement learning and model predictive control. *Water Resources Research*, *58*(7), e2021WR030703.

Tscheikner-Gratl, F., Caradot, N., Cherqui, F., Leitão, J. P., Ahmadi, M., Langeveld, J. G., ... others (2019). Sewer asset management–state of the art and research needs. *Urban Water Journal*, *16*(9), 662–675.

Turnbull, B. W. (1976). The empirical distribution function with arbitrarily grouped, censored and truncated data. *Journal of the Royal Statistical Society: Series B (Methodological)*, *38*(3), 290–295.

Wirahadikusumah, R., & Abraham, D. M. (2003). Application of dynamic programming and simulation for sewer management. *Engineering, Construction and Architectural Management*, *10*(3), 193–208.

Yin, Z., Leon, A. S., Sharifi, A., & Amini, M. H. (2023). Optimal control of combined sewer systems to minimize sewer overflows by using reinforcement learning. In *World* *environmental and water resources congress 2023* (pp. 711–722).

Zeng, X., Wang, Z., Wang, H., Zhu, S., & Chen, S. (2023). Progress in drainage pipeline condition assessment and deterioration prediction models. *Sustainability*, *15*(4), 3849.

Zhang, Z., Tian, W., & Liao, Z. (2023). Towards coordinated and robust real-time control: A decentralized approach for combined sewer overflow and urban flooding reduction based on multi-agent reinforcement learning. *Water Research*, *229*, 119498.

BIOGRAPHIES

Lisandro A. Jimenez-Roa is a doctoral candidate in Computer Science at the University of Twente, The Netherlands. He has a background in civil engineering and has contributed to various projects in structural health monitoring, finite element modeling, and damage detection through data analytics and machine learning. His current research focuses on Prognostics and Health Management, specifically engineering systems within the PrimaVera project (**https://primavera-project.com**), emphasizing multi-state stochastic degradation modeling and maintenance policy optimization using Reinforcement Learning techniques.

Thiago D. Simão is an Assistant Professor in the Eindhoven University of Technology, the Netherlands. He obtained his PhD in Computer Science at Delft University of Technology. Previously, he was a PostDoc researcher at Radboud University Nijmegen. His research interests lie primarily in reliably automating sequential decision-making, focusing on reinforcement learning.

Zaharah Bukhsh is an assistant professor at Eindhoven University of Technology, Eindhoven, Netherlands . She holds a Master's degree in computer science and a Ph.D. in engineering technology from University of Twente, Enschede, Netherlands. Her research focuses on developing data-driven methods with deep learning and deep reinforcement learning. Her research targets broad application areas including asset management, scheduling, and resource optimization. She has contributed to several H2020 and NWO research projects.

Tiedo Tinga is a full professor in dynamics based maintenance at the University of Twente since 2012 and full professor of Life Cycle Management at the Netherlands Defence Academy since 2016. He received his Ph.D. degree in mechanics of materials from Eindhoven University in 2009. He is chairing the smart maintenance knowledge center and leads a number of research projects on developing predictive maintenance concepts, mainly based on the physics of failure models, but also following data-driven approaches.

Hajo Molegraaf completed his PhD at the University of Groningen and has worked as an assistant and postdoc researcher at the University of Geneva and Yale University. Since October 2022, Molegraaf joined as a Research Fellow within the Formal Methods and Tools (FMT) group in the EEMCS faculty at Twente. Additionally, Molegraaf is a co-founder and software developer at Rolsch Assetmanagement, a company based in Enschede, The Netherlands.

Nils Jansen is a full professor at the Ruhr-University Bochum, Germany, and leads the chair of Artificial Intelligence and Formal Methods. The mission of his chair is to increase the trustworthiness of Artificial Intelligence (AI). Prof. Jansen is also an associate professor at Radboud University, Nijmegen, The Netherlands. He was a research associate at the University of Texas at Austin and received his Ph.D. with distinction from RWTH Aachen University, Germany. His research is on intelligent decision-making under uncertainty, focusing on formal reasoning about the safety and dependability of artificial intelligence (AI). He holds several grants in academic and industrial settings, including an ERC starting grant titled Data-Driven Verification and Learning Under Uncertainty (DEUCE).

Mariëlle Stoelinga is a full professor of risk analysis for high-tech systems, both at the University of Twente and Radboud University, the Netherlands. She holds a Master's degree in Mathematics & Computer Science, and a Ph.D. in Computer Science. After her Ph.D., she has been a postdoctoral researcher at the University of California at Santa Cruz, USA. Prof. Stoelinga leads various research projects, including a large national consortium on Predictive Maintenance and an ERC consolidator grant on safety and security interactions.

APPENDIX A. PARAMETERS OF MULTI-STATE DEGRADATION MODELS

Table 7. MSDM hyper-parameters for cohort `CMW`, using hazard functions modeled with the *exponential* ($\lambda^E(t|\epsilon)$), *Gompertz* ($\lambda^G(t|\alpha,\beta)$), and *Weibull* ($\lambda^W(t|\eta,\rho)$) probability density functions.

	$\lambda^E(t\|\epsilon)$	$\lambda^G(t\|\alpha,\beta)$		$\lambda^W(t\|\eta,\rho)$	
$i \to j$	ϵ	α	β	η	ρ
$1 \to 2$	2.4E-02	2.3E+00	8.4E-03	1.3E+00	4.4E+01
$2 \to 3$	9.4E-03	2.1E-02	5.5E-02	2.9E+00	7.7E+01
$3 \to 4$	5.7E-03	3.3E+00	2.8E-03	3.5E+00	8.1E+01
$4 \to 5$	1.8E-02	2.4E+00	8.7E-03	7.0E+00	5.5E+01
$1 \to F$	3.0E-18	1.4E-01	3.1E-04	4.1E-06	4.6E+01
$2 \to F$	6.0E-04	8.8E-01	7.0E-19	2.7E-04	4.6E+01
$3 \to F$	1.0E-18	2.2E-03	4.5E-02	3.0E-05	4.7E+01
$4 \to F$	1.0E-18	9.8E-05	8.6E-03	1.1E-03	4.5E+01
$5 \to F$	1.0E-18	7.0E-19	3.8E-01	1.7E+00	5.9E+01

Table 8. Initial state vector S_k^0 for MSDM of cohort `CMW`.

S_k^0	Exponential	Gompertz	Weibull
$k = 1$	9.89E-01	9.58E-01	9.23E-01
$k = 2$	1.26E-17	0.00E+00	2.59E-02
$k = 3$	3.70E-23	4.00E-02	3.10E-02
$k = 4$	1.11E-02	1.61E-03	1.13E-02
$k = 5$	2.11E-22	2.00E-15	2.07E-03
$k = F$	3.87E-22	1.56E-04	6.40E-03