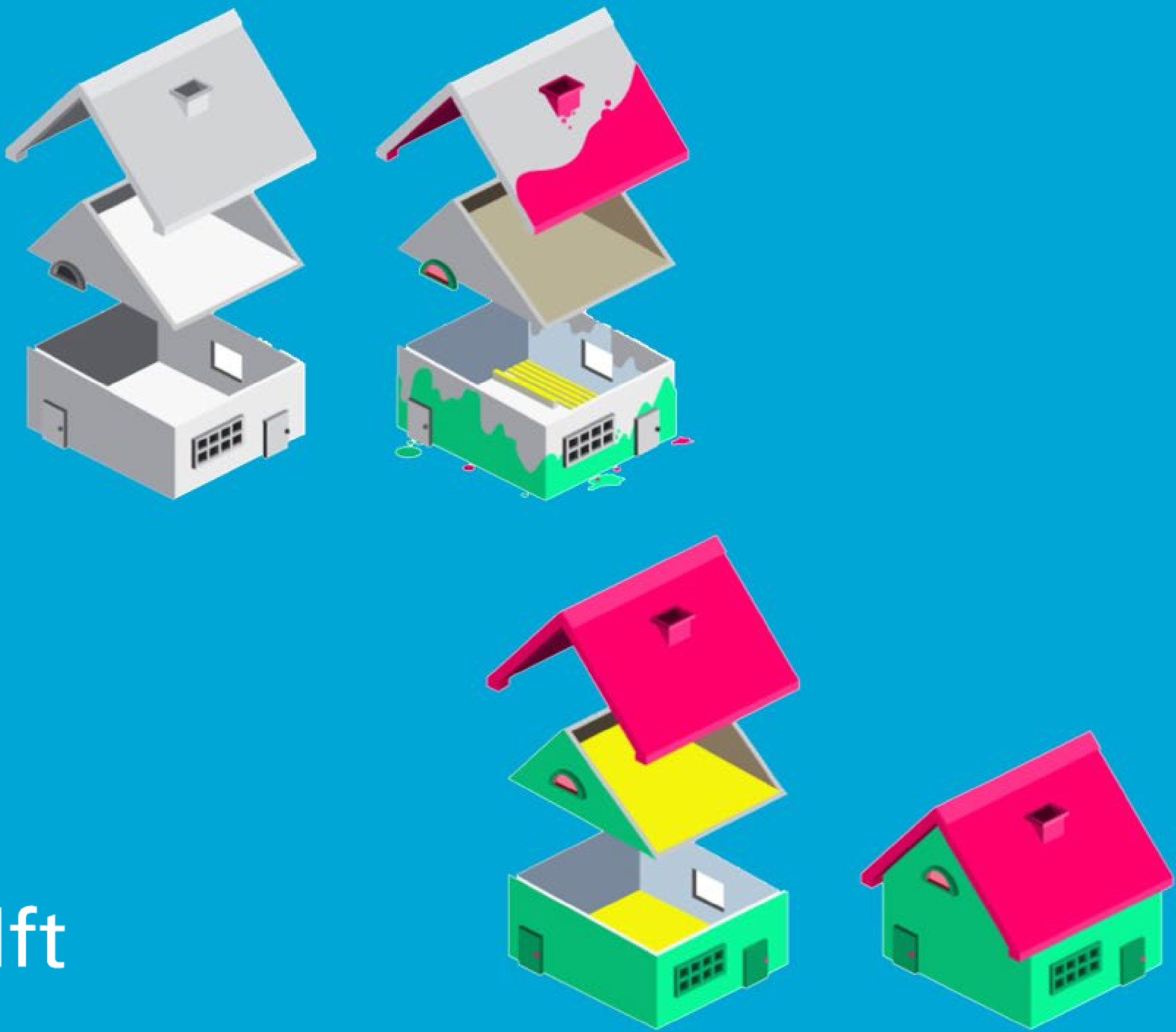
Retrofitting Planning Optimization



TU Delft

Delft Technical University, Department of Architecture, Building Technology track

Retrofitting Planning Optimization

Thesis Report

Course : MSc in Architecture, Urbanism & Building Sciences (Building Technology Track)

Nefeli Karadedou-Isoua

Student number : 5629535

Main Mentor : Charalampos Andriotis

Second Mentor : Michela Turrin

13/06/2024

TABLE OF CONTENTS

ABSTRACT

In 2020, as part of the European Green Deal, the European Commission initiated the Renovation Wave strategy with the goal of doubling the annual rate of energy renovation for buildings by 2030 (*Delivering the European Green Deal - European Commission*, n.d.). However, there is a delay in how quickly we renovate compared to the standards (Bouckaert et al., 2021). The reasons are mainly economical (Esser Anne et al., 2019) or based on the facts that a lot of homeowners choose to conduct retrofitting because of trigger points (Energy Saving Trust, 2015).

Planning the house energy renovation in steps can be more economically feasible in some cases(Fritz et al., 2019).

Up until now there has been some limited research on the optimization of the retrofitting planning as can be seen in the works of (Maia et al., 2021) and (Maia et al., 2023). However, planning the cost optimal timing of the steps to find the correct sequence and time of actions based on the various variables of the environment (e.g. degradation rate of the materials, economic growth, budget allocation among others) can become a very complex problem since a lot of uncertainties about the environment are involved.

Retrofitting planning correlates to typical predictive maintenance problem where some uncertainty about the environment can be involved and each action will have consequences on the actions that can be taken later , as explained by (Ogunfowora & Najjaran, 2023)

There are various techniques that can solve sequential problems, including mixed integer linear programming(Littman, 1996) but the most upcoming ones come from the realm of reinforcement learning algorithms. Reinforcement learning is a machine learning approach that focuses on finding the optimal policy (optimal action for each scenario we are in now based on what we want to achieve in the future)(Sutton & Barto, 2018). For this reason, reinforcement learning will be used as an alternative to produce a roadmap of the staged retrofitting actions.

Working with multiple components of a system has been addressed before by (Andriotis & Papakonstantinou, 2018) and (Krachtopoulos, 2023) among others. However, the studies involving maintenance of buildings is rather limited. Ferreira et al.,(2023) provided a basis workflow however they didn't seem to take into account the building's energy demand as part of their objective function.

Based on the limited research having been conducted in this domain the following thesis will focus on the development solving of the building retrofitting planning optimization using Reinforcement Learning approaches.

The following research will be developed into two sections. The first section will consist of framing the theories needed to create a basic understanding of some of the aspect involved on the formulation and solution of the problem. The first chapter will consist of

the basic retrofitting theories as well as the basic notions of retrofitting in steps, as well as the optimization case studies . The theory and basic principles of Markov Decision Processes and Reinforcement Learning algorithms if Q-learning and Value Iteration will be explained at the second chapter to establish an understanding of the way that the planning optimization will be addressed. Based on those chapters conclusions will be drawn about the importance of conducting the following research. Chapter three will involve building energy performance principles. The chapter will delve also into the aspects affecting the building degradation and focus on the aspect of insulation degradation. Conclusions about the building behavior and the overall methodology needed to be followed for simulating the building's performance through time will be drawn from there.

The second section will be focusing on the implementation of the theory into practice. An overview of the general methodology will be presented, and different steps and bottlenecks of the overall process will be discussed. In this part, the different parts of the code will be explained and incorporated in diagrams.

PART 1 : LITERATURE REVIEW

1. BACKGROUND

The Energy Performance of Buildings Directive (EPBD) is a set of rules created by the European Union to tackle the challenge of reducing energy usage in buildings and making them more environmentally friendly. Its main goal is to make sure that by the year 2050, all buildings in Europe will use much less energy(*Energy Performance of Buildings Directive*, n.d.) This is crucial because currently, buildings in Europe consume 40% of all the energy used in the EU. EPBD works alongside the Energy Efficiency Directive, which promotes various energy-saving measures(*New Energy Efficiency Directive Published - European Commission*, n.d.).

According to their findings, 85% of buildings in the EU are constructed before 2000, and out of those, a concerning 75% have poor energy performance. ((*New Energy Efficiency Directive Published - European Commission*, n.d.) . Moreover, residential buildings, which make up around 75% of the total building stock, hold enormous potential for energy savings. However, improving energy efficiency in buildings isn't straightforward. It's often costly, and not everyone can afford it. That's why the EU is devising special plans and rules to help. For instance, the revised EPBD is aiming (among other things) , to increase the rate of building renovations, especially for the worst-performing ones. Fixing up buildings to use less energy could save a lot of money for people living in them (*Energy Performance of Buildings Directive*, n.d.).

Despite these efforts, progress in deep energy efficiency renovations remains sluggish. For instance, the International Energy Agency (IEA) estimates that the annual rate of such renovations for existing buildings is less than 1% (Thijs Vandenbussche, 2021). This underscores the importance of tools like building renovation passports[1], which provide crucial information for homeowners, helping them understand the costs and benefits of renovating their buildings (Fabbri et al., 2016)

One of the ways the directive works is through the development of roadmaps for retrofitting of buildings. The BRP is a tool that provides a long-term, step-by-step renovation roadmap for a specific building, resulting from an energy audit, outlining relevant measures and renovations that could improve the energy performance (Fritz et al., 2019). In short, these roadmaps help owners decide on what to retrofit and when. (Sesana et al., 2020)Even more, some of these directives incorporate the possibility of planning the house energy renovation in steps in order to make it more economically feasible.

In summary, the EPBD is a vital tool in the EU's efforts to make buildings more energy-efficient and environmentally friendly. By tackling issues like poor energy performance and promoting renovations, it's paving the way for a greener future while also considering the financial well-being of building owners and tenants.

With the rise of these tools to create personalized roadmaps, came also the rise of frameworks and optimization tools that try to optimize the timing and packages of renovation measures. The optimization methodologies tried to outline methodologies to deal with different aspects of roadmap creations. Some analysed economical aspects , others included CO2 emissions and Life cycle costs. Interesting aspect was the case of planning optimization of building components involving the uncertainty towards their physical degradation.

Even though all these cases had interesting aspects to offer in the overall literature review , they underlined the lack of case studies focusing on planning optimizations. The following thesis will try to delve into literature review to outline important factors that should be considered and draw a methodology based on different aspects, with hope to create a basis for future works to take inspiration from and consider developing more wholistic tools.

[1] Available in both print and electronic formats, the Building Renovation Passport is a document that offers a thorough, step-by-step plan for a total renovation of a particular building over a period of 15 to 20 years. It provides owners with customized guidance on remodelling options and lays out the stages of the renovation for everyone involved.(Fabbri et al., 2016).

2. RETROFITTING PRINCIPLES

2.1. Step by step retrofitting

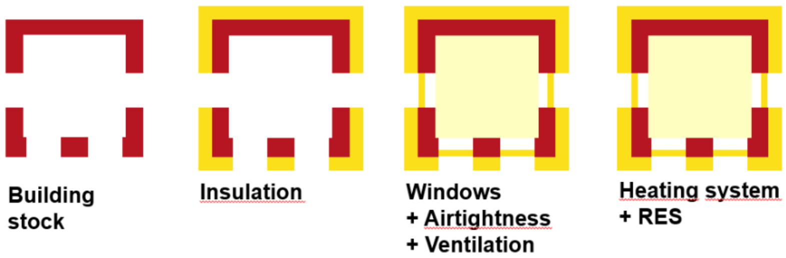


Figure 1 Retrofitting components step by step(Overall Retrofit Plan for Step-by-Step Retrofits to EnerPHit Standard(*Passive House Institute*, n.d.)

Step-by-step retrofitting is a process of improving the energy efficiency of a building through a series of planned renovations over time. This approach is particularly useful for buildings that were constructed before energy efficiency standards were widely adopted. The step-by-step retrofitting process has direct implications on the improvement of building stocks' energy efficiency, and consequently, the achievement of decarbonization targets set for 2050 (Maia et al., 2021).

Staged renovations are the most common across Europe, with 85% of renovations in Germany being staged. This approach allows for less disruptive and more cost-efficient renovation measures by aligning them with given 'trigger points'. (Fritz et al., 2019) Trigger points are circumstances that initiate home improvement projects unrelated to energy savings, providing an opportunity to modernize the energy performance of houses. There are two methods for retrofitting a house: the single-step method, where all measures are implemented simultaneously, and a phased approach, including room-by-room, measure-by-measure, and step-by-step sub-categories. The step-by-step retrofitting strategy highlights the adaptability of building energy retrofitting to align with stakeholders' cost constraints.

The step-by-step retrofitting process is a strategic approach to building renovation that considers the timing, cost, and interdependencies of various renovation measures. It aims to maximize energy savings and contribute to the decarbonisation of the building sector(Maia et al., 2021)

2.2. Retrofitting strategies

In general, two types of measure exist for building retrofitting: technical system measures, involving changes to building services, and envelope measures, such as adding insulation to envelope elements (Sewnath, 2024)

Technical system measures, encompass domestic hot water, heating, cooling, lighting, and mechanical ventilation, are integral components, with a step-by-step approach emphasizing comprehensive packages at each stage (Maia et al., 2021)

Envelop measures on the other hand focus on adding or changing insulation layers to walls, roofs, or floors, and upgrading windows and doors (Konstantinou, 2014).

Konstantinou, (2014) classified the envelop interventions into five categories: "wrap it," "add-in," "replace," "Add-on," and "cover-it."

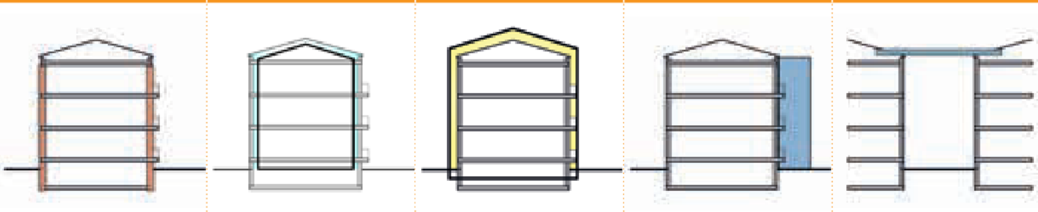
	Replace	Add-in	Wrap-it	Add-on	Cover-it
					
Description	Old façade elements removed and replaced with new ones	Upgrade from the inside	'Wrapping' the building in a second layer	New structure is "added on" to the existing building	Cover parts or entire internal and external courtyards and atria
Intervention-variation	Replace the entire façade Replace parts	Internal insulation Cavity insulation Box window	External insulation, Cladding of the balconies Second skin façade	Small intervention, such as adding new balconies New building as an extension Additional floor	Cover parts or entire Heated or unheated space
Benefits	New components with better performance Eliminate the physical problems	Adequate for monu-mental status Increase the thermal resistance	Solve thermal bridges Increase the thermal resistance Different cladding possibilities Little disturbance	Out-dated façade no longer exterior New façade with performance Increase space Functional benefits	Create thermal buffer Enhance natural ventilation with stack effect Out-dated façade no longer exterior Additional space
Limitations	Great impact on users Higher costs	Critical connection thermal bridging need attention Big disturbance for users	Not applicable to mo-numental buildings Possible space limi-tation	Needs to be com-bined with other strategies for facades non-adjacent to new structure Structural limitation	Not applicable to all cases Depending on layout and function of the building Overheating risk

Table 4.1
Refurbishment strategies according to type of intervention in residential building refurbishment

Figure 2 Refurbishment strategies (Konstantinou, 2014)

(Maia et al., 2023) proposed the combination of measures for staged building renovations to achieve energy efficiency. Combining measures was considered crucial to avoid "lock-in effects," where partial renovations could hinder future energy savings or necessitate premature replacements[2].

In their paper , they emphasized the importance of considering the sequence and combination of renovation measures to maximize energy efficiency and minimize CO2 emissions over time:

- Roof and Upper Ceiling: Insulation of the roof or upper ceiling to reduce heat losses.
- External Wall and Windows: Insulation of external walls along with window replacement to improve energy efficiency and air-tightness.
- Heating System: Replacement of old heating systems with more energy-efficient options like heat pumps.
- Floor and Cellar Ceiling: Insulation of floors and cellar ceilings to enhance thermal performance.

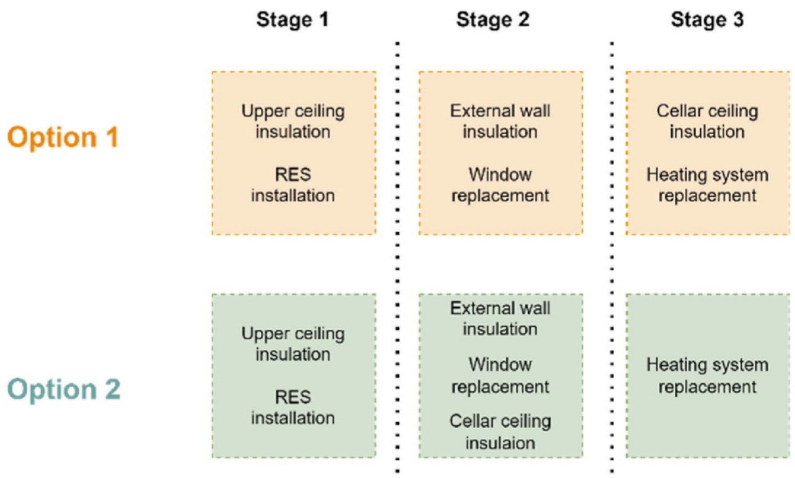


Figure 3 Stages of renovation roadmaps as proposed in (Maia et al., 2023)

[2] According to their paper, lock-in effects occur when anticipated energy savings are not realized, and circumstances prevent swift adjustments, resulting in prolonged periods of suboptimal energy efficiency. This situation leads to missed opportunities for maximizing energy savings.For instance, if a heating system is promptly replaced, but subsequent improvements to the building envelope's quality are delayed, the new heating system may operate inefficiently due to over-dimensioning, resulting in suboptimal performance during partial operation loads. Even though the term seems to be used in other sources as well, further research should be made about the mechanics and importances of the effect in the overall building performance before and after the retrofitting action.

2.3. Economic aspects of step-by-step retrofitting

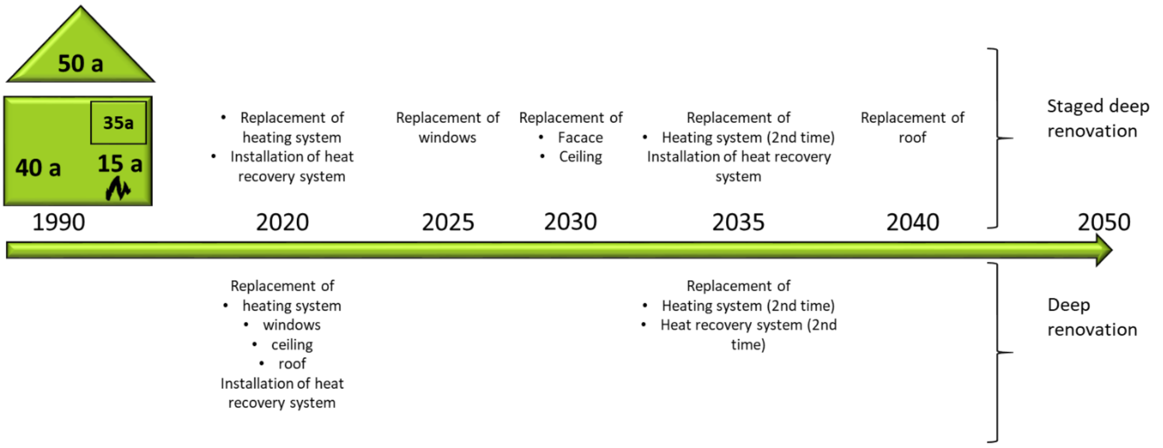


Figure 4 Timeline of the two long-term renovation approaches for a 1990s building case with comparison between stepped approach and deep renovation approach (Fritz et al., 2019)

(Fritz et al., 2019) assessed the economic feasibility of staged retrofitting versus one-step deep renovation for six typical German residential buildings. They used the net present value[3] method to compare the costs and benefits of both strategies over a period of 32 years, considering different parameters such as interest rate, energy price, and carbon price. They found that for the exemplary residential buildings built in 1980s and 1990s, the staged deep renovation was cheaper than the one-step deep renovation by 4.8 % to 9.3 %. However, for the 1960s buildings, the one-step deep renovation was more economic feasible due to the high initial heat demand and the large saving potential of efficiency measures. The authors also discussed the impact of various parameters such as energy and carbon prices, interest rate, and timing of the measures on the economic viability of the two strategies.

In their research (Maia et al., 2021) presented a model to deliver the optimum timing of step-by-step retrofitting activities. This model maximized the net present value of households' energy-related cash flows and delivered the optimum timing when each step should be performed. When comparing both single-step and step-by-step approaches, the step-by-step presented 11–22% higher cumulated energy savings. The renovation period would last between 1 and 14 years and 2 to 11 years, depending on whether interdependency of measures was considered. Based on the results, it was concluded that in low-income house-holds living in less energy efficient single-family houses, the

[3] NPV tells that money now is worth more than the same amount of money in the future. In an example, the $5 now is worth more than $1 every week for 5 weeks(*Net Present Value (NPV): What It Means and Steps to Calculate It*, n.d.).

retrofitting steps would – under consideration of maximizing NPV – be performed with time delay, if no loans or appropriated countermeasures were available.

2.4. Step by Step Retrofitting planning optimization

Lopes et al., (2021)highlighted the increasing interest in staged renovation as a suitable strategy and the increase of various initiatives providing methodological guidebooks, informing stakeholders of information of the building retrofitting measures, identifying steps in which they can be incorporated and planning the roadmaps in general terms, regarding the state of their buildings, what financial programs exist and how to financially plan their retrofitting actions and which to take first (Bastian et al., 2016) (Sesana et al., 2020).

2.4.1.1. STAGED RETROFITTING PLANNING

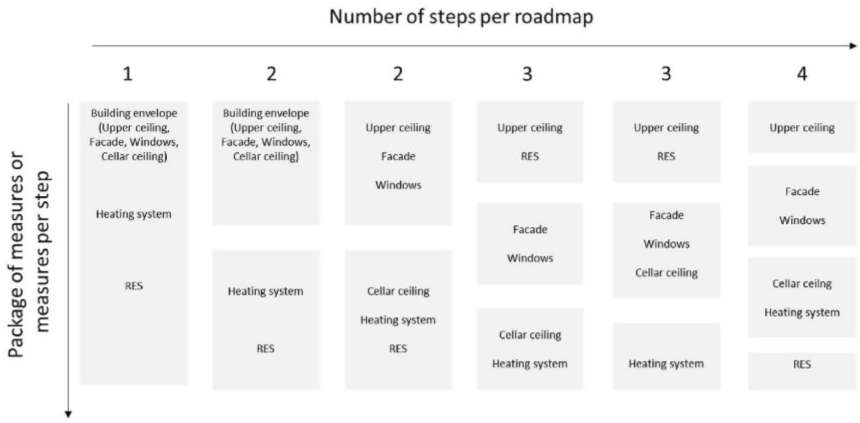


Figure 5 Different roadmap variants in terms of the number of steps and possible combinations of measures for individual buildings (Maia et al., 2023)

In their work Maia et al. delineated crucial factors of staged retrofitting planning. These factors include

1)the number of stages,

2)the allocation of measures within each stage, and

3) the sequence in which stages are executed.

They stated that crafting a roadmap comprising five or more steps will appear technically challenging, as it could elevate the risk of lock-in effects. However, such roadmaps might hold significance when considering the financial constraints of the building owner (Foxon, n.d.).

2.4.1.2. RETROFITTING PLANNING OPTIMIZATION

Even though there are a few papers discussing planning maintenance of a building in time steps to optimize the retrofitting planning schedule (Vollmer et al., 2022) there are only a few introducing a complete methodology of the planning optimization of the deep staged retrofitting while considering the interdependency between retrofitting actions and the sequence which they should follow.

In 2021 Maia et al., presented a framework for step-by-step retrofitting optimization that considered the budget restrictions, building material degradation process and the interdependency between retrofitting steps. Interdependency in this case meant the order that the steps should be conducted to avoid the lock in effect. They developed a mixed integer linear programming workflow that maximized the net present value of a household's energy cash flow. The model's primary goal was to calculate the optimum timing of implementation for each package of renovation measures in an optimization period of 30 years[4].

The NPV is calculated as the sum of cash flows over the optimization period, adjusted for the discount rate and residual value.

$$maxNPV = \sum_{t}^{T} \frac{CF_t}{(1+r)^t} + \frac{L_T}{(1+r)^{tp}}$$

Where:

- NPV = the energy related net present value in euros.
- CF_t = cash flow of energy related balance
- L_T= residual value of the retrofitting measures in year T
- r = interest rate (%)
- tp = annual depreciation time
- T = period in years

Cash flows were determined by subtracting investment costs, running energy costs, and operation and maintenance costs from the cumulated allocated energy-related assets. This provided the budget restrictions which the homeowners had to face.

$$CF_t = A_t - IC_t - EC_t - OMC_t$$

[4] From 2020 to 2050.

In this expression:

- CF_t represents the cash flow at time t.
- A_t = cumulated allocated energy related assets at time t.
- IC_t = sum of investment cost in the year t.
- EC_t = annual running energy costs in the time t (EUR/a).
- OMC_t= annual running operation and maintenance costs in the time t (EUR/a)

To define the right timing of the single-step approach, a simplified assumption was made: the right timing for a single-step is necessary for homeowners to accumulate money needed to perform the retrofitting. However, they stated that if government subsidies are available, this may change. Also, the single-step approach was preferable if the retrofitting was performed as soon as possible.

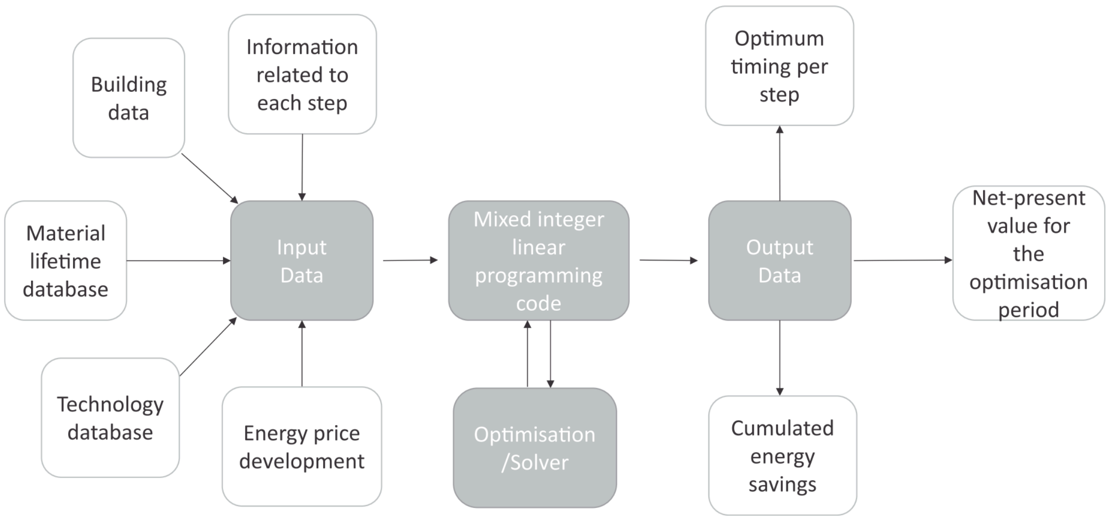


Figure 6 Code architecture for the optimization model of the case study (Maia et al., 2021)

According to the paper, the input data for the model consisted of technical and economic aspects:

Technical aspects:

- Specification of renovation measures and their combinations. Each combination was called a "step".
- Identification of each building case's materials
- Specifications of material's lifetime according to existing databases
- Calculations of material's ageing process.

Economical aspects:

- Costs of conducting each "step".
- Energy price developments per energy carrier
- The homeowner's budget restrictions.

The sensitivity analysis that they conducted revolved around how the model behaved according to the building case scenarios at hand.

The optimization model took a lot of assumptions, for example the interest rate being at 3% while the ageing rate of materials was considered through a Weibull distribution. The paper itself recognized that the constraints of the static input data that they used, underlining the need for an interface for the users to incorporate the correct data as the time processed. Even more the interdependency of renovation steps and homeowners' budget restrictions impact the model's results.

3. INTRODUCTION TO RL THEORY

In the previous chapter it was explained that the staged retrofitting planning can be seen as a sequential decision making problem and thus be formulated and solved using Reinforcement Learning algorithms.

In this section, the basic theories of reinforcement learning theory will be described. For better understanding, examples related to the context of the thesis will be used.

A simple problem version of the actual thing we are trying to solve will be used to explain the theories.

The problem:

A house has a starting state of energy demand. The house's energy demand is rising with rise of the U values. There is some uncertainty when the house will reach a certain threshold when the energy bills will be more costly on the long run than doing a retrofitting action that will return the house in a good energy performance state. Objective, for the homeowner , is to see when to perform each retrofitting action to the components of the envelop in order to pay the least amount of costs at the end of 60 years' time[5].

2.5. Sequential decision making

Sequential decision making is a crucial component of general decision-making theory. The fundamental concept is that the decisions made in the present not only impact the immediate future but also influence future decisions. Sequential decisions are significant as they lay the groundwork for subsequent actions(Littman, 1996).

In sequential decision making there are some words to describe the general theory:

An **environment** which is the world that we can perceive. For example, we can describe the environment as the house with all the dynamics that affect its performance. For example, its geometry, its components, the state that these components are in etc.

[5] We can assume that the house will be passed down to the next generation or that the owner has accepted the house at a young age , hence the years. The main objective is to determine the optimal timing of retrofitting a house during the building's lifespan.

An **agent** (a system responsible of interacting with the world and making decisions) is interacting with this environment (anything that is not the agent). In this case, the algorithm that is called to observe the environment and take decisions will be the agent.

A **policy**, is the agent's behavior given a certain observation, meaning the way that it will decide to act given the state of the environment.

An agent's actions are usually met with a reward. The reward describes the consequence of an action. For example, if the agent decides to retrofit a component of a house he will have to pay some investment. The reward is a very important component that defines the policy. In a usual loop of interaction, the agent observes the environment. Following a policy p that dictates what action to do based on the observation, he takes an action and receives a positive or negative reward consequently. The action brings him to a new state of the environment.

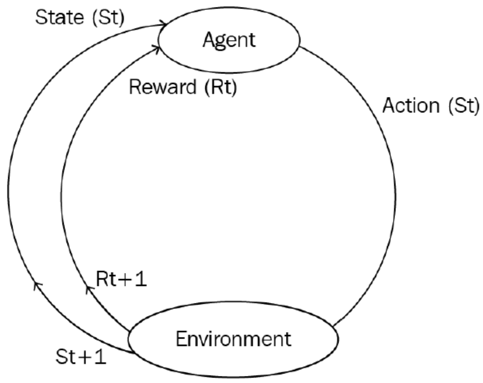


Figure 7 The agent environment interaction

2.6. Markov Decision Process

The Markov Decision Process (MDP) is a mathematical framework that was developed to model and formalize sequential decision-making problems. It provides a way to represent what decisions an agent makes in the environment that he is in at that time.

The Markov Decision Process can be described as a tuple consisting of the variables (S, A, P, R, g) which are the State, Action, Probability of Transition Probability, expected Reward and discount Factor(Sutton & Barto, 2018).

2.6.1. States

A state represents a configuration or scenario or representation of the environment at a point in time and includes the necessary information to determine the outcomes of actions taken from that state. In Markov Decision Processes, the Markov property which is the first part of the equation states that the state that we are in depends only on the previous state, meaning it is completely independent of the past. For example, if the state describes the environment of the house based on the energy demand, and that energy demand is suboptimal, probably because the agent hasn't performed any actions that would better it in the previous step.(Sutton & Barto, 2018)

A state S_t is Markov if and only if,

$$\mathbb{P}[S_{t+1} \mid S_t] = \mathbb{P}[S_{t+1} \mid S_1, S_2, \cdots, S_t]$$

Markov Property | Image: Rohan Jagtap

Figure 8 Markov Property as given in (*Markov Decision Process Explained | Built In*, n.d.)

2.6.2. <u>Transitions</u>

If there is certainty about the state that will be reached next from the current state, the system is described as deterministic. However, states often account for the uncertainty in the system, or else called stochasticity, which is typically modelled through transition probabilities between states. The Markov Process can be described as the probability of transitioning to the future state S', given that we are in the state S. An example of how transitions work based on a very simplified problem can be found on the appendix.

$$\mathcal{P}_{ss'} = \mathbb{P}[S_{t+1} = s' \mid S_t = s]$$

State transition probability. | Image: Rohan Jagtap

Figure 9 State transition probability equation as given in (*Markov Decision Process Explained | Built In*, n.d.)

Probabilities can be stationary or non-stationary. Stationary probabilities describe a system that the same probabilities describe the transitions of one state to another throughout all the optimization time. Nonstationary probabilities on the other hand are changing.

Some of the requirements of transition probability matrix are that:

- All probabilities are non-negative and no greater than 1 so 0=<p<=1
- The sum of each row should equal to 1

2.6.3. Rewards

Markov Reward process incorporates the expected reward which can mean all the possible rewards that we can receive after we transition to the next state. Rewards define how good or bad reaching a certain state or taking a certain action is. In the example problem terms, if the house is in a good state, we will receive a better reward in the form of lower energy bills. If we don't do something, it can be zero or negative (depending on the state that our house is in).

$$\mathcal{R}_s = \mathbb{E}[R_{t+1} \mid S_t = s]$$

Figure 10 Reward as given for Markov Reward Process in (*Markov Decision Process Explained | Built In*, n.d.)

2.6.4. Return

During an episode in reinforcement learning, the agent visits a series of states. The sum of the rewards that is going to receive from the series of actions that is going to take, is called a Return. The return is the objective function. For the retrofitting planning problem, the return can be described as the money that will be spent or saved over the optimization period.

$$G_t = R_{t+1} + R_{t+2} + R_{t+3} + \cdots + R_T$$

Figure 11 Return expression as given in (*Markov Decision Process Explained | Built In*, n.d.)

2.6.5. Discount Factor

The discount factor γ helps define the importance of the present reward over the future rewards. The discount g factor has a range of decimal numbers between 0 and 1. If the discount factor is closer to 1 it means that all the rewards will amount the same no matter how much into the future they are. If the discount factor is closer to 0, then the more the current reward gets more important compared to all the future rewards. For example, because of the inflation of the economy, the homeowner's money will have a smaller value as the time goes by. Based on that, the Return is described as below:

The return G_t is the total discounted reward from time-step t

$$G_t = R_{t+1} + \gamma R_{t+2} + \cdots = \sum_{k=0}^{\infty} \gamma^k R_{t+k+1}$$

Figure 12 Discounted return expression as given in (*Markov Decision Process Explained | Built In*, n.d.)

2.6.6. Policy

As stated earlier, a policy is defining the action that will be taken in a given state.

A stochastic policy is the probability that the agent will choose action a given that we are in state *S* over the probability of taking action b. A deterministic policy on the other hand always prescribes the same action for a given state. To give an example if the agent is following a stochastic policy, it will choose to say, change the boiler 30% of the times and do nothing 70% of the time. In a deterministic policy, it will just choose 100% one action in a given state, say “Do nothing” when the state of the energy demand of the house is ‘Good’.

A policy π is a distribution over actions given states,

$$\pi(a \mid s) = \mathbb{P}[A_t = a \mid S_t = s]$$

Figure 13 Policy expression as given in (*Markov Decision Process Explained | Built In*, n.d.)

2.6.7. Bellman Optimality Equation

A Bellman equation is a way to express the value of a decision-making process. It helps us understand how good it is to be in a certain situation (state) and take certain actions in that situation to maximize future rewards. Understanding the Bellman equations is crucial in reinforcement learning algorithms. By solving the Bellman equations, the best strategy (policy) can be found. The Bellman Optimality Equations are a specific form of the Bellman Equations used when we aim to find the best possible policy, or strategy, for maximizing rewards. It is a fundamental concept in reinforcement learning and dynamic programming.

The Bellman Optimality Equation expresses the relationship between the value of a state and the values of its successor states. There are two important components of this equation: one for the state-value function (V∗) and one for the action-value function (Q∗).

2.6.7.1. STATE-VALUE FUNCTION (V∗)

The optimal state-value function V∗(s) represents the maximum expected return (sum of discounted future rewards) that can be achieved from state s by following the best policy.

The Bellman Optimality Equation for the state-value function is described as :

$$V^*_{(s)} = max_a \sum_{s'} P(s'|s,a)[R(s,a,s')] + \gamma\, V^*_{(s')}$$

where:

- $V^*_{(s)}$ is the optimal value of state s.
- max_a is the maximum value over all possible actions (a).
- $P(s'|s,a)$ is the probability of transitioning to state s' from state s after taking action a.
- $R(s,a,s')$ is the reward received after transitioning to state s' from state s after taking action a.
- γ is the discount factor, which represents the present value of future rewards.
- $V^*_{(s')}$is the optimal value of the next state s'.

2.6.7.2. STATE-ACTION VALUE FUNCTION (Q∗)

The optimal action-value function Q∗(s,a) represents the maximum expected return that can be achieved from state s, taking action a, and thereafter following the best policy.

The Bellman Optimality Equation for the action-value function is:

$$Q^*_{(s,a)} = \sum_{s'} P(s'|s,a)[R(s,a,s') + \gamma max_{a'} Q^*_{(s',a')}]$$

where:

- $Q^*_{(s,a)}$ is the optimal value of taking action a in state s.
- ∑s' denotes the sum over all possible next states s'.
- P(s'|s,a)is the probability of transitioning to state s' from state s after taking action a.
- R(s,a,s')is the reward received after transitioning to state s' from state s using action a.
- γis the discount factor.
- $max_{a'} Q^*_{(s',a')}$ is the maximum value of the next state-action pair (s',a').

2.7. Taxonomy of Reinforcement Learning methods

As explained above, a lot of reinforcement learning and dynamic programming methods rely on Bellman equations . Reinforcement learning methods can be classified into various categories. Primarily, problems in reinforcement learning are divided into Prediction Problems and Control Problems.

Prediction Problems involve estimating the value function, which predicts future rewards given a state or a state-action pair. The objective is to forecast the cumulative rewards from the present state to the end of a given period.

Control Problems focus on finding the optimal policy that maximizes the cumulative reward over time. In these problems, the policy is not predefined, and the goal is to discover it through interaction or simulation. The optimal policy is one that achieves the best balance between various factors, such as costs and benefits, over the long term.(Littman, 1996)

Additionally, these problems can be further categorized based on the approach used: Model-based or Model-free.(Dong et al., 2020)

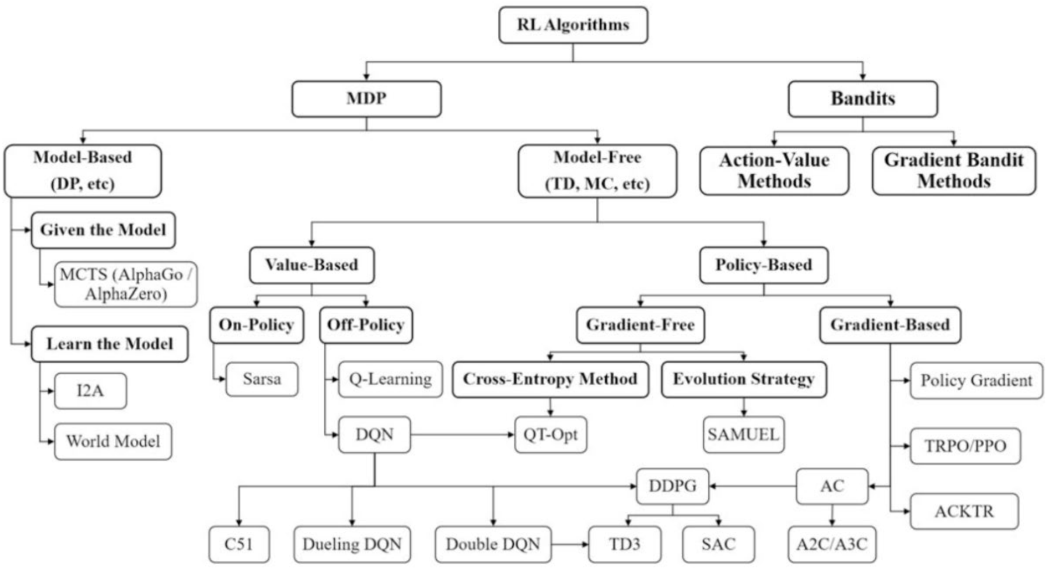


Figure 14 Taxonomy of the different RL methods(Dong et al., 2020)

Model-based Methods involve creating a detailed simulation of the environment, which can then be used for planning and making predictions about future states and rewards, guiding decision-making. Model-free Methods learn the value function or policy directly through interaction with the environment, without a predefined model.

Reinforcement learning also distinguishes between Value-Based and Policy-Based methods. Value-Based methods focus on optimizing the action-value function from where, after optimization is done, the optimal policy is derived. Policy-Based methods directly optimize the policy without relying on the value function. Combining these approaches leads to actor-critic algorithms, which use the value function to update the policy.

Another important distinction in reinforcement learning is between On-Policy and Off-Policy methods. On-Policy methods evaluate or improve the policy that is used to make decisions. The agent interacts with the environment based on the current policy, and this same policy is updated. Off-Policy methods on the other hand, evaluate and /or improve a policy different from the one used to generate the data. These methods allow the agent to use experiences collected by different policies (Dong et al., 2020).

2.8. Planning Algorithms

There are several approaches to solve MDPs, including value iteration, policy iteration, and linear programming (LP). These algorithms differ in their methodologies but share the common goal of optimizing decision-making.

Value iteration is an iterative algorithm that updates the value of each state by considering the expected rewards of future states, gradually converging to the optimal value function. Policy iteration, on the other hand, involves iteratively evaluating a policy and improving it until an optimal policy is found. Last, Linear programming is also a method that can be used MDP planning problems by using constraints to find the which actions will give the maximum value function.

2.8.1. Linear Programming

According to Sutton (Sutton & Barto, 2018) " Linear programming methods can also be used to solve MDPs, and in some cases their worst-case convergence guarantees are better than those of DP methods. But linear programming methods become impractical at a much smaller number of states than do DP methods (by a factor of about 100). For the largest problems, only DP methods are feasible."

Linear programming provides a powerful alternative for solving MDPs. In this approach, the MDP problem is formulated as a set of linear constraints and an objective function. The core idea is to express the optimal policy and value functions in terms of linear equations. The objective is to maximize the total expected reward, subject to the constraints imposed by the MDP's dynamics.

In an LP formulation, we define variables for the value of each state and use constraints to ensure these values are consistent with the MDP's transition probabilities and reward structure. The linear program maximizes the sum of the state values, ensuring that the expected reward from following the optimal policy is captured. Solving this linear program yields the optimal value function, from which an optimal policy can be derived.

To solve an MDP using Linear Programming, we can formulate it as an optimization problem. The LP approach focuses on finding the optimal value function $V(s)$ for each state s, which represents the maximum expected cumulative reward starting from state s.

Since the objective is to maximize the sum of the values of all states(Abbeel, n.d.):

$$Maximize \sum_{s \in S} V(s)$$

For each state s, the value function V(s) must satisfy the Bellaman equation, which ensures consistency with the transition dynamics and rewards of the MDP:

$$V(s) \geq R(s, \alpha) + \gamma \sum_{s\prime \in S} P(s' | s, \alpha) V(s') \qquad \forall s \in S, \forall \alpha \in A$$

Based in that function, the objective function can be written as:

$$Maximize: V(s1) + V(s2) + \ldots + V(sn)$$

Each V(s) can be solved as there are more knowns than unknowns. Given all the possible V(s) values we can use the constraints to check which actions will maximize the value of each state, thus determining the optimal policy.

2.8.2. Dynamic Programming

Dynamic Programming (DP) is a collection of algorithms based on the Bellman equations. Their aim is to compute the optimal policy (the optimal action that can be taken in each state) is a model-based approach.

Dynamic Programming methods provide a foundation for all the other RL methods. However, they are limited over the assumption of the perfect model of the environment and their great computational expense. (Sutton & Barto, 2018)

The key idea of DP is the use of the value functions to organize and structure the search for good policies. Dynamic programming can be separated into two approaches: the Value iteration and the Policy Iteration.

2.8.2.1. Value Iteration

Value Iteration uses the Bellman equations to iteratively update the value function until it converges to the optimal values. Once we have these optimal values, we can extract the best policy by leveraging the Policy Extraction Equation derived from the Bellman Expectation Equation.(Sutton & Barto, 2018)

Initialization

Initialize the value function $V(s)$ for all states s ,using arbitrary or zero values. This is done in a form of a table that each state is described by a value function.

Iterative Update

For each iteration calculate its value function using the Bellman equation for each state.

$$V(s) = max_a(R(s,a) + \gamma \sum_{s'} P(s'|s,a) V(s'))$$

Where :

- max_a is the action that maximizes the value.
- $R(s,a)$ is the immediate reward received after taking action a in state s
- γ is the discount factor
- $P(s'|s,a)$ is the probability of transitioning to state s' from state s after taking action a

Then update the value function for each state using the maximum expected future reward achievable from that state.

Convergence Check

- Compute the change in value Δ (delta) for each state:

$$\Delta = max_s|V_{new}(s) - V_{old}(s)|$$

- If Δ is less than a small threshold e, the values have converged, and the iteration stops.

Convergence in dynamic programming occurs when the value function no longer changes significantly between iterations or after a certain number of iterations. The iterative process continues, updating the value function table by repeatedly applying the Bellman update equation until the changes in values become sufficiently small. In simpler problems, convergence may be reached after a few iterations, as seen in the provided example where it happened after the second iteration. However, in more complex problems, convergence might require thousands of iterations.

To determine convergence, a threshold is often employed. For instance, if the value of the medium state in the previous iteration was 5.05 and the value of the medium state after the last iteration is 5.055, with a threshold set at 0.01 (or 5e-2), the value has reached convergence as it is not changing significantly anymore. This threshold helps in determining when to stop iterating and consider the values stable.

Policy extraction

$$\pi^*(s) = argmax_a(R(s,a) \; + \gamma \sum\nolimits_{s'} P(s'|s,a)V(s'))$$

(This equation tells to choose the action a that maximizes the expected sum of immediate and future rewards)

Once the value function has converged, extract the optimal policy $\pi^*(s)$ by selecting the action that maximizes the expression inside the max operator of the Bellman equation. The calculation of the value table provides insight into the desirability of being in each state and taking a certain action.

To extract the optimal policy, the final Q-table obtained after convergence is leveraged. This table contains the expected cumulative rewards (Q-values) for each state-action pair. From the Q-table, we select the action with the highest Q-value for each state, as it represents the action that maximizes the expected cumulative rewards. The q table determines the optimal action to take based on the state we are in, thus is the policy.

Value Iteration, for estimating $\pi \approx \pi_*$

Algorithm parameter: a small threshold $\theta > 0$ determining accuracy of estimation
Initialize $V(s)$, for all $s \in \mathcal{S}^+$, arbitrarily except that $V(terminal) = 0$

Loop:
| $\Delta \leftarrow 0$
| Loop for each $s \in \mathcal{S}$:
| $\quad v \leftarrow V(s)$
| $\quad V(s) \leftarrow \max_a \sum_{s',r} p(s',r|s,a)\big[r + \gamma V(s')\big]$
| $\quad \Delta \leftarrow \max(\Delta, |v - V(s)|)$
until $\Delta < \theta$

Output a deterministic policy, $\pi \approx \pi_*$, such that
$\pi(s) = \operatorname{argmax}_a \sum_{s',r} p(s',r|s,a)\big[r + \gamma V(s')\big]$

Figure 15 Pseudocode depicting the Value Iteration process (Sutton & Barto, 2018)

In simple terms, Value iteration is a method used to find the best strategy for an agent in a given environment. It works by repeatedly estimating the expected future rewards for each state. This estimation considers both the immediate rewards the agent receives and the value of the next states it might transition to.

The process continues iteratively, updating the value function until it reaches a stable solution. This stable solution represents the best possible expected return from each state, given the agent's actions and the dynamics of the environment.

In essence, value iteration helps the agent make informed decisions by determining the most rewarding actions to take from each state.

2.8.2.2. Policy Iteration

Policy iteration is another Dynamic Programming method that can be used to find the optimal policy for a Markov Decision Process (MDP). The process involves iteratively evaluating and improving a policy until it converges to the optimal policy.

In policy iteration, the three components—policy evaluation, policy improvement, and iteration—work together to find the optimal policy.

To find the optimal policy the methodology is as follows:

Random policy Initialization

Start with an arbitrary policy π. This means that the value function for each state starts with value zero and a random policy (random action for each state) is introduced.

$$V^{\pi}(s) = \sum_{s'} P^{a}_{ss'}[R^{a}_{ss'} + \underbrace{\gamma V^{\pi}(s')}_{\text{Random Values}}]$$

Figure 16 Random policy initialization graph taken from (Ravichandiran, 2018)

Policy Evaluation

The goal of Policy Evaluation is to determine the Value Function (V^{π}) for a given policy π. That means that the goal in this part is to compute the value function V^{π} for the current policy π. This step involves solving the Bellman expectation equation iteratively until the value function converges. The equation is:

$$V^{\pi} = \sum_{\alpha \in A} \pi(\alpha|s)\left(R(s,\alpha) + \gamma \sum_{s' \in S} P(s'|s,\alpha)V^{\pi}(s')\right)$$

In this part the goal is to compute the value of each state given the action that is provided by the policy.

However, after the first iteration , the value function that will be calculated will not be accurate since zero state values were given in the initialization. So, in the next iteration, to compute the value function , the updated values for each state will be used.

$$V^{\pi}(s) = \sum_{s'} P^{a}_{ss'}[R^{a}_{ss'} + \underbrace{\gamma V^{\pi}(s')}_{\text{Use state values from 1}^{\text{st}}\text{ iteration}}]$$

Figure 17 State value after 1st iteration (Ravichandiran, 2018)

The iterations will be repeated until the values of each state change very little.

Policy Improvement

The value function was computed using an arbitrary policy, so in the beginning it will not be optimal . However, after the convergence of the value function a new policy can be obtained and be used to update the old policy.

In order to extract the new policy , the State action value function will be computed using the value function computed in the previous step.

$$Q(s,\alpha) = \sum_{s'} P_{ss'}^{\alpha}[R_{ss'}^{\alpha} + \gamma V(s')]$$

Based on the Q function, the action that gives the maximum value of each state will be chosen for the new policy. That means that all the actions that can be taken from that state will be checked using the Q function and the one that brings the maximum Q value will be chosen.

This step is called 'Policy Improvement' , where the new policy that will be extracted is defined as:

$$\pi'(s) = arg \max_{\alpha \in A}\left(R(s,\alpha) + \gamma \sum_{s' \in S} P(s'|s,\alpha)V^{\pi}(s') \right)$$

Convergence Check

Every time a new policy is extracted, it is compared with the previous policy. If the policy is the same as the previous policy , then it can be concluded that the algorithm has converged and π is the optimal policy. Otherwise , the new policy is set to be the updated policy (π = π') and the steps of policy evaluation and improvement are repeated.

Policy Iteration (using iterative policy evaluation) for estimating $\pi \approx \pi_*$

1. Initialization
   $V(s) \in \mathbb{R}$ and $\pi(s) \in \mathcal{A}(s)$ arbitrarily for all $s \in \mathcal{S}$

2. Policy Evaluation
   Loop:
       $\Delta \leftarrow 0$
       Loop for each $s \in \mathcal{S}$:
           $v \leftarrow V(s)$
           $V(s) \leftarrow \sum_{s',r} p(s', r \mid s, \pi(s))\big[r + \gamma V(s')\big]$
           $\Delta \leftarrow \max(\Delta, |v - V(s)|)$
   until $\Delta < \theta$ (a small positive number determining the accuracy of estimation)

3. Policy Improvement
   policy-stable $\leftarrow$ *true*
   For each $s \in \mathcal{S}$:
       old-action $\leftarrow \pi(s)$
       $\pi(s) \leftarrow \operatorname{argmax}_a \sum_{s',r} p(s', r \mid s, a)\big[r + \gamma V(s')\big]$
       If *old-action* $\neq \pi(s)$, then *policy-stable* $\leftarrow$ *false*
   If *policy-stable*, then stop and return $V \approx v_*$ and $\pi \approx \pi_*$; else go to 2

Figure 18 Pseudocode for Policy Iteration process(Sutton & Barto, 2018)

2.9. Reinforcement Learning

2.9.1. Q- learning

Q-learning is a Reinforcement learning technique that uses the Q-values of state-action pairings to determine the best course of action. The goal is to learn the best actions to take in each state to maximize cumulative rewards over time. Q-learning is an off-policy algorithm because it updates the Q-values based on the maximum future reward $max_{a'} Q(s', a')$ regardless of the action actually taken in the next state. This means the learning is driven by the optimal action, not necessarily the action that was executed.

The steps of finding the optimal policy using Q-learning are:

Initialization:

Initialize a Q-table with starting values of zero for each item, Q(s,a), representing the estimated future reward for action a in state s.

$$Q(s, a) = 0 \qquad \forall s, a$$

Where:

- Q(s,a): Represents the Q-value for a given state s and action a.
- $\forall s, a$ means "for all states s and actions a." In other words, this initialization applies to every possible combination of states and actions in the environment.

Start a new episode:

Initialize the starting state s.

Repeat for each step of the episode until a terminal state is reached:

Action Selection:

For each state s, select an action a using an action selection policy called ε-greedy policy. This policy balances exploration (trying new actions) and exploitation (choosing actions that are known to be good).

- With probability ϵ, select a random action (exploration).
- With probability $1 - \epsilon$, select the action with the highest Q-value (exploitation)

$$\begin{cases} \text{random action with probability } \epsilon \\ \text{argmax}_a Q(s, a') \text{ with probability } 1 - \epsilon \end{cases}$$

Take action and Observe:

The action that is selected is passed to the environment to be implemented. The environment then and receives a reward R as a feedback and the observation of the next state S' .

Update Q-Value

Update the estimate Q-value for the state-action pair using the Bellman equation. The update rule is:

$$Q(s,a) \leftarrow Q(s,a) + \alpha(r + \gamma max_{a'} Q(s',a') - Q(s,a))$$

This can be explained as

$$Q_{t+1}(s,a) = (1 - a) \times old\ estimate. + a \times new\ estimate$$

Where:

- α is the learning rate which determines how much new information overrides old information
- γ is the discount factor determining the importance of future rewards
- $max_{a'} Q(s',a')$ is the maximum expected future reward for the next state s'.

Transition to Next State:

Set the current state to the next state: s⟵s'

Check for Terminal State:

If s' is a terminal state, end the current episode.

In terminal state, there are no future states to consider. Therefore the Q-Value update for the terminal state simplifies to

$$Q(s,a) \leftarrow Q(s,a) + \alpha(r + Q(s,a))$$

In this case the γ is not applied because there no future state to consider and the reward r is directly used to update the Q-value.

Determine Convergence:

Periodically run several episodes using the current policy derived from Q-values.

Calculate the average cumulative reward over these episodes.

If the average cumulative reward does not change significantly between evaluations, it indicates that the policy has stabilized and the Q-values have converged.

Figure 19 Pseudocode of Q-learning process (Sutton & Barto, 2018)

Q-learning (off-policy TD control) for estimating $\pi \approx \pi_*$

Algorithm parameters: step size $\alpha \in (0, 1]$, small $\varepsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(terminal, \cdot) = 0$

Loop for each episode:
  Initialize S
  Loop for each step of episode:
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Take action A, observe R, S'
    $Q(S, A) \leftarrow Q(S, A) + \alpha\big[R + \gamma \max_a Q(S', a) - Q(S, A)\big]$
    $S \leftarrow S'$
  until S is terminal

4. BUILDING PERFORMANCE DEGRADATION

3.1. Energy system for the build environment

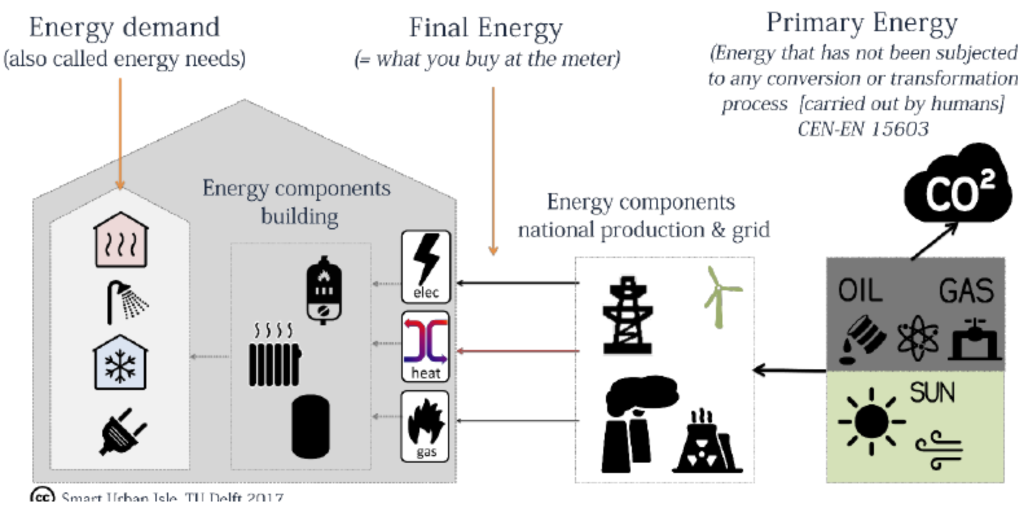


Figure 20 The Energy Supply Chain: energy demand, final energy and primary energy(Jansen et al., 2021)

The energy cycle starts with demand, which is the specific form of energy required—such as heat, cooling, or light. This demand must be met using renewable resources available on Earth. Various technical components are involved in the intermediate stages to convert, store, and distribute this energy, ensuring its availability at the right time and place. In the context of the built environment, the technical supply chain can be categorized into three levels: energy demand, final energy, and primary energy. Understanding these levels is essential for accurately assessing energy systems, as they have distinct implications and significance.

Energy demand refers to the specific form of energy needed by the end user. In buildings, this includes the amount of heat required for heating or cooling to maintain desired temperature conditions over time. The energy demand at the building level is determined by the building's energy system and is influenced by factors such as insulation and air tightness, as well as how the building is used. It does not depend on the specific technical systems or equipment used to meet this demand.

Final Energy

Final energy is the energy in the form of an energy carrier—such as gas, electricity, or heat—that is used on the consumer's side of the meter. According to Eurostat, it is "the energy consumed by end users," which typically includes households and other consumers.

Primary Energy

Primary energy is the original form of energy that has not undergone any conversion or transformation. It includes both fossil and renewable energy sources.

3.2. Theory of thermal energy balances

An inventory of all building-related energy flows is required to ascertain how much energy a room or layout requires to maintain the necessary degree of health and comfort. Heating is required if the total of these energy flows is negative because the building is not receiving enough heat. When the overall energy flows are positive, cooling is required. An energy balancing is the process of compiling an inventory of all energy flows. This inventory relies on the thermodynamic law, which says the quantity of energy entering an isolated system at a steady temperature is equal to the quantity of energy exiting the system.

$$Q_{in} = Q_{out} \ \ or \ \ Q_{in} - Q_{out} = 0$$

In the context of stationary heat balances, it can be stated that equilibrium in a system is achieved when the total heat flow amounts to zero. In stationary heat transfer scenarios, temperatures and heat fluxes remain constant over time. Within a stationary system, the combination of heat flows from both the interior and exterior must adhere to the conservation law by the result being zero. The equation of this looks like this :

$$Q_{trans} + Q_{vent} + Q_{inf} + Q_{sol} + Q_{int} + Q_{added} = 0$$

Where:

- Q_{trans} = the heat flows due transmission
- Q_{inf} = the heat flows due to infiltration
- Q_{vent} = the heat flows due to ventilation
- Q_{sol} = the heat flows due solar gains
- Q_{int} = the heat flows due to internal heat gains
- Qadded = the heat flows from any additional source placed to cool or heat the room
    - If Qadded < 0: Qadded = Qcooling
    - If Q_{added} = 0: no heating or cooling required.
    - If Qadded > 0: Qadded = Qheating

Heat flows appear positive when they enter the building system and negative when they leave it(Van Unen, 2019).

Transmission

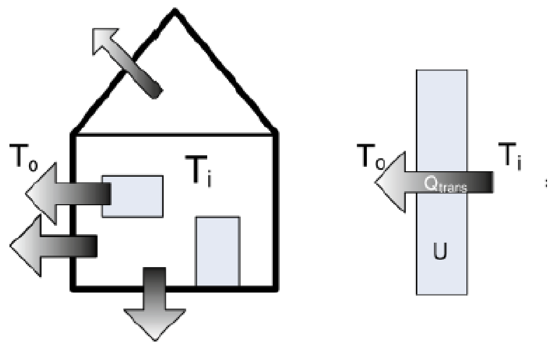


Figure 21 illustrates the transmission losses occurring through walls, windows, floors, and roofs [31].

In transmission, the heat will transfer through the building envelop from the hottest point to the coldest due to the differences between interior temperature and the exterior. The heat will transfer though the walls, glazing, roof, and floors in a combination of conduction, convection and radiation.

The total heat transfer due to transmission can be defined from the equation:

$$Q_{trans} = Q_{trans,s} + Q_{bridge}$$

Where :

- Q_{trans} = the transmission of heat happening through a construction, in Watts (W)[6]
- Q_{bridge} = the transmissions happening due to thermal bridges[7].

The heat transmission (losses or gains) through a construction can be found through the equation:

$$Q_{trans,s} = U \times A \times (T_e - T_i)$$

Where :

[6] W stands for watts, which is the unit of power. It quantifies the rate at which energy is transferred or converted. In the context of heat transfer, watts indicate how quickly heat flows through a material (such as walls, windows, or roofs) from one side to another.

[7] Thermal bridges, also known as thermal leakages, arise when insulation is not properly positioned or when there is a structural element such as a column, beam, or balcony that connects the interior with the exterior.

U : The total heat transfer coefficient (W/m²K, W= Watts)[8]

- A : The surface area of the construction wall(m^2)
- T_e : The outdoor temperature (Kelvin)
- T_i : the indoor temperature (Kelvin)

The thermal transmittance is calculated through the equation:

$$U = \frac{1}{\frac{1}{a_i} + R_c + \frac{1}{a_0}}$$

Where :

- a_i : The coefficience of radiation[9]
- a_o : The coefficience of convection
- R_c: the thermal insulance , meaning the ' ability of a material to resist heat flow'10

The total thermal resistance of a component can be determined by the equation

$$R_{c,tot} = R_{c,1} + R_{c,2} + R_{c,3} = \frac{d_1}{\lambda_1} + \frac{d_2}{\lambda_2} + \frac{d_3}{\lambda_3}$$

Where

- Rc, 1,2,3 is the thermal resistance of the materials 1,2,3 that consist the component. To put it simply, In a wall, they would be the layers that consist the wall.
- d : the thickness of each construction layer
- λ : the thermal conductivity of the material (W/mK)

[8] The U-value of the entire window construction is typically employed for windows, and dirt, rather than ambient air, serves as the transmission medium for ground-level signals.

[9] The a_i is often assumed (when the building's air speed is comparatively low) and for α_o the value strongly depends on the wind speed, but a yearly average value of 25 W/m2K is often applied.

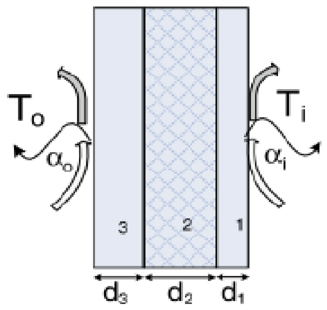
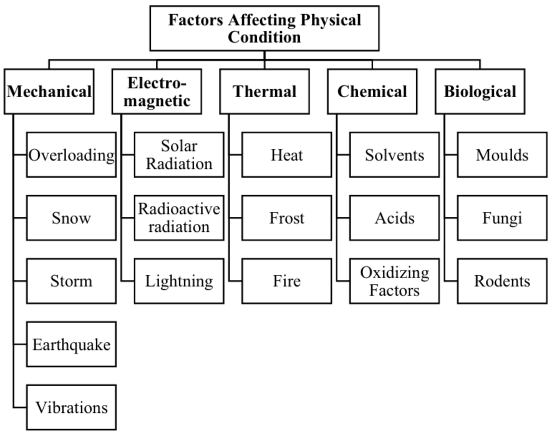


Figure 22 Wall construction depicting the thermal transmittance rom [31].

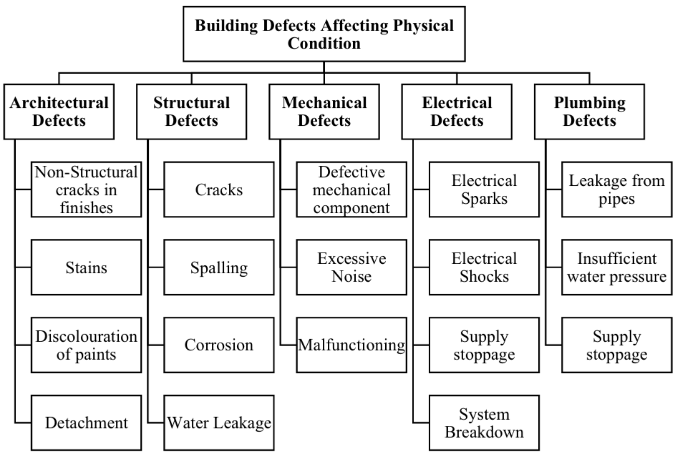
These equations are fundamental for estimating heat losses through the envelope of a house. By accurately determining the U-value and surface area of the building envelope, along with the temperature difference across the envelope, you can calculate the rate of heat loss and, subsequently, the energy demand required for heating the house.(Van Bueren Hein Van Bohemen et al., n.d.)

3.3. Building Performance degradation

3.3.1. Factors influencing the performance of the building



(a) Factors affecting physical condition of buildings



(b) Categories of physical defects in buildings

The thermal performance of the envelope might vary from one structure to another. Different factors might play a role in affecting the physical condition of the building and thus also the envelope's performance. If the envelope's performance is affected by these factors, it might affect the overall building. According to (Faqih et al., 2020), it is possible that a component of the building might influence the overall performance.

Figure 23 Factor affecting a building's physical condition (Faqih et al., 2020)

Mechanical factors like mechanical and kinetic loads (e.g. gravity, earthquake, man-made vibrations etc.), can cause structural defects affecting the functionality and safety of the building. Electromagnetic factors such as solar and ultraviolet radiation as well as extreme temperature factors can influence the heating and degradation rate of the building causing loss of function. Even more, chemical factors like waters, acids from bird droppings, salts and others can also affect the degradation of the building materials. Last, animals like rodents, worms, birds, encountered in all environments, can affect the building materials and make them degrade by contaminating them.

These factors can create defects, meaning deviations of the intended performance of building components. Even more, one defect can cause other defects to appear: moisture for example, can have a cascading effect causing discoloration, mold, corrosion and other problems in different building elements.

Environmental factors, such as poor indoor air quality, thermal condition influenced by climate change and, inadequate lighting and poor acoustics can also contribute to building performance as people will try to adjust the systems for comfort. (Faqih et al., 2020)

3.3.2. Simulations techniques and equations

There are different ways to simulate building performance degradation. (Eleftheriadis & Hamdy, 2018) considered the degradation of envelope elements and heat supply systems. In order to determine the degradation percentage of the heating systems and the insulation, they used equations that relied one the age factors.

The Energy Plus software was utilized to perform the annual heating demand of the building. The simulation perceived each floor as a thermal zone and accounted for mechanical ventilation(set as 0.33) and natural infiltration rates(0.17 air changes per hour). The results were then exported for post processing analysis in MATLAB to determine the amount of energy consumption deriving from heating, hot water , ventilation system and primary energy consumption. The equations then were used to simulate the performance change of the HVAC and insulation components in order to calculate the change of the performance in the span of 20 years. The authors simulated 4 separate scenarios with variations on the central heating system and maintenance quality for the analysis period. Based on the results they concluded that for their case study, the combined degradation of

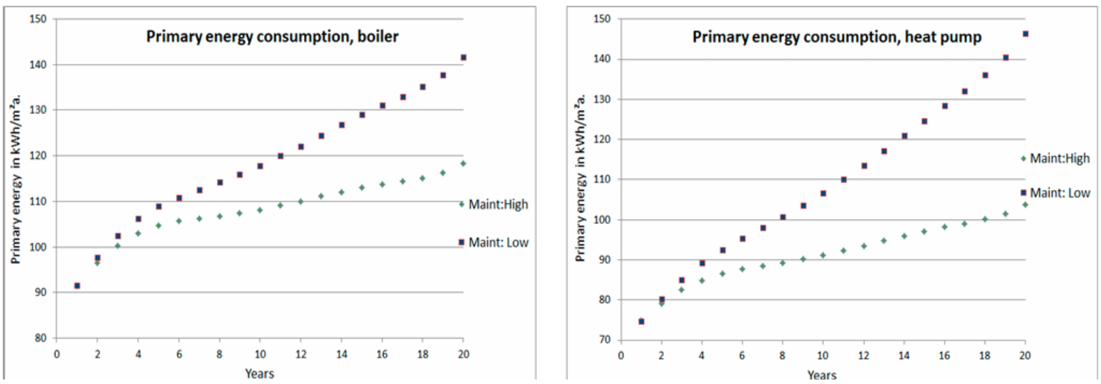


Figure 24 Energy consumption increase as result of heating system (boiler and heat pump) and XPS insulation degradation(Eleftheriadis & Hamdy, 2018)

HVAC and component degradation can lead to up 47% additional primary energy consumption.

Another case study of interest was conducted by Taki & Zakharanka (Taki & Zakharanka, 2023) where they simulated the building degradation by developing dynamic models of two different case study buildings and by conducting initial thermal dynamic simulations to establish baseline energy efficiency indicators, primarily focusing on heating energy requirements. Based on the analysis of existing data and additional research, probable scenarios for the performance deterioration of various building components were created. These scenarios included degradation in insulated glass units, increased thermal conductivity of insulation, reduced airtightness, contamination of heat recovery system filters, and performance drops in photovoltaics.

Table 4. *Cont.*

The Operational Stage	Windows (↓Filling with Inert Gas)	Insulation (↑Thermal Conductivity)	Air Permeability (↑)	MVHR (↓Heat Recovery Performance)		PV Modules (↓Performance)
	Scenario 1.2	Scenario 2.1	Scenario 3.2	Scenario 6.1 (Detached)	Scenario 6.1 (Apartments)	Scenario 5.1
5 years	89	102.5	110	75	50	82.5
10 years	88	105	110	75	50	80
15 years	87	107.5	112.5	75	50	77.5
20 years	86	110	115	75	50	75
25 years	85	112.5	117.5	75	50	72.5

↑—increase; ↓—decline.

Table 5. Stages of building components' degradation (the worst-case scenario) (based on [14]).

The Operational Stage	Windows (↓Filling with Inert Gas)	Insulation (↑Thermal Conductivity)	Air Permeability (↑)		MVHR (↓Heat Recovery Performance)		PV Modules (↓Performance)
	Scenario 1.3	Scenario 2.2	Scenario 3.4	Scenario 3.1	Scenario 6.2 (Detached)	Scenario 6.2 (Apartments)	Scenario 5.2
As design	100	100	100	100	90	65	100
As built	90	100	100	100	90	65	90
1 year	89	102	130	120	50	50	85
5 years	85	110	130	120	0	0	80
10 years	80	120	130	120	0	0	75
15 years	70	130	132.5	122.5	0	0	70
20 years	50	140	135	125	0	0	65
25 years	0	150	137.5	127.5	0	0	60

↑—increase; ↓—decline.

Figure 25 In order to model building energy performance, 268 dynamic simulations were first run, taking into account the specific deterioration of each component. 32 simulations were then run to assess the cumulative deterioration effects of several components. The best-case (lowest degradation) and worst-case (highest degradation) scenarios, which are shown in Tables 4 and 5, were taken into consideration for simultaneous deterioration.(Taki & Zakharanka, 2023)

Dynamic thermal simulations were then performed using DesignBuilder software based on the EnergyPlus engine. These simulations applied the degrading characteristics to the dynamic models to observe changes in energy performance, specifically heating energy consumption. A total of 268 simulations were conducted for individual component degradation, and 32 for combined degradation effects, with scenarios reflecting both minimal and maximum degradation levels. The results were compared against initial performance indicators to identify patterns and assess the impact of degradation. The study concluded that more airtight and insulated buildings are more sensitive to component degradation, with faster increases in heating energy consumption observed in these buildings compared to less airtight ones. This methodology provides insights for improving building design and maintenance by accounting for degradation over time.

3.3.3. Insulation Materials

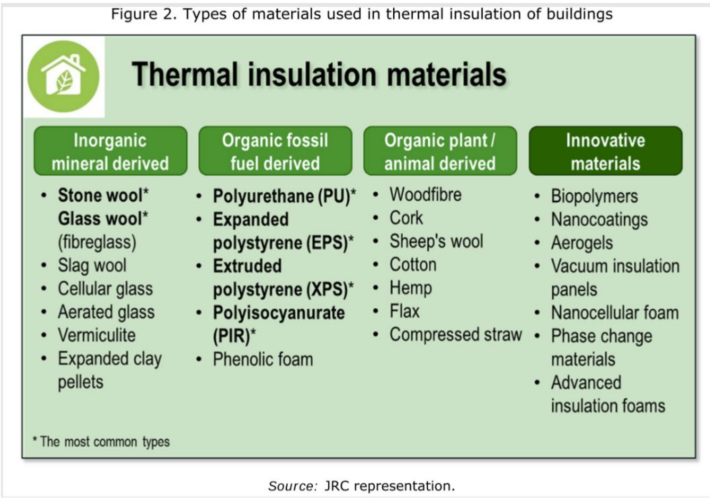


Figure 26 Types of Insulation materials (Pavel & Blagoeva, n.d.)

Thermal insulation comprises materials or combinations thereof that slow down the rate of heat transfer through conduction, convection, and radiation when appropriately applied. Employing thermal insulation aids in diminishing reliance on HVAC systems. According to (Pavel & Blagoeva, n.d.), these insulation materials are utilized across various building components such as walls, roofs, ceilings, windows, and floors. The majority of available thermal insulation materials fall into four general categories:

Inorganic Materials:

Mineral wool encompasses various inorganic insulation materials such as rock wool, glass wool, and slag wool. These materials exhibit low thermal conductivity, are non-flammable, and highly resistant to moisture damage. However, they may pose health risks such as skin and lung irritation.(Hung Anh & Pásztory, 2021).

Organic Materials:

Conventional materials like polyurethane (PUR), polyisocyanurate (PIR), extruded polystyrene (XPS), and expanded polystyrene (EPS) are widely preferred in many buildings and thermal energy storage applications due to their combination of low thermal conductivity and affordability.

Natural-Based Organic Materials:

Organic insulation materials are derived from natural resources and are increasingly utilized in buildings due to their appealing attributes, including renewability, recyclability, and environmental friendliness. (Hung Anh & Pásztory, 2021).

Advanced Materials:

Advanced insulation materials include vacuum insulation panels (VIPs), gas-filled panels (GFPs), aerogels, and phase change materials (PCM). VIPs, exhibit exceptionally low thermal conductivity values, have a long lifespan exceeding 50 years (Hung Anh & Pásztory, 2021).

Inorganic materials like glass wool and rock wool, which constitute 60% of the market, while organic materials make up 27%. Conventional materials like polyurethane (PUR), polyisocyanurate (PIR), extruded polystyrene (XPS), and expanded polystyrene (EPS) are favored in many constructions due to their low thermal conductivity.

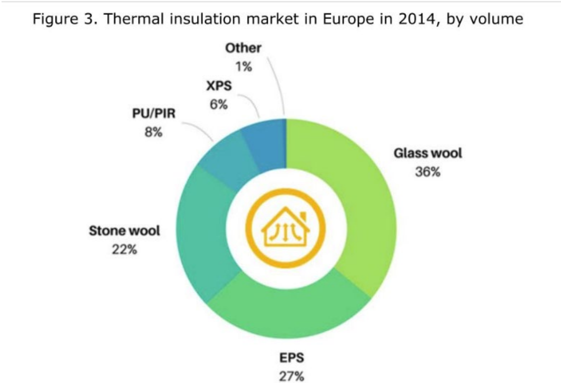


Figure 27 European market demand of thermal insulation

Glass and stone wool collectively constitute 58% of the European thermal insulation market. Expanded polystyrene (EPS) foam stands out as the preferred material for external wall insulation due to its affordability and superior performance characteristics. Polyurethanes (PU and PIR) offer distinct advantages, particularly in scenarios demanding higher thermal insulation requirements, as they can achieve the same insulation efficiency with thinner layers compared to other insulation materials(Pavel & Blagoeva, n.d.).

3.3.4. Influential Factors on Insulation Material Conductivity:

Most of commonly used building insulation materials considerable influenced by the environmental conditions due to their porous structure and the proportional of air or other gas filling up the cells. The heat conduction of an insulator is strongly influenced by several factors: temperature, moisture content, density, aging time, along with secondary factors such as raw material, cell gases, nature and microstructural of solid component, air surface velocity, pressing, and sample thickness(Hung Anh & Pásztory, 2021).

Temperature:

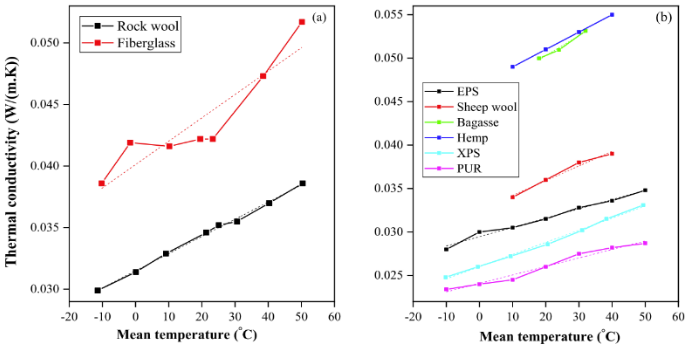


Figure 28 Thermal conductivity change of different insulations based on temperature changes (Hung Anh & Pásztory, 2021)

Due to the fact that molecule movements is the basis of heat conduction, the temperature has a huge impact on thermal conductivity of insulation materials

Fibrous insulation materials such as fiberglass, hemp fibers, flax fibers, cellulose fibers, sheep wool are more affected by temperature than other insulation materials.

Moisture Content:

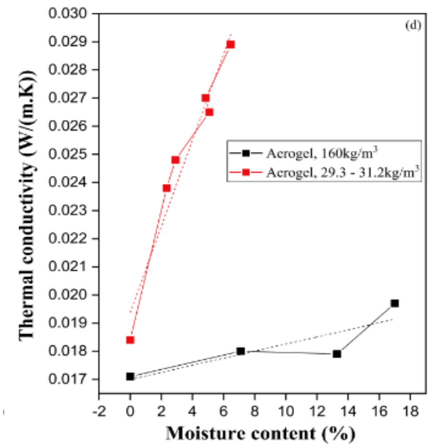


Figure 29 Thermal conductivity change in aerogel insulation based on the moisture amount (Hung Anh & Pásztory, 2021)

Excessive moisture contributes to deteriorating habitation quality, reduced thermal resistance, additional mechanical stresses, salt transport, and material decay. Moisture can compromise the effective thermal properties of building envelopes, insulated walls, and roofs. Since water conducts heat about 20 times more effectively than stationary air, water absorption invariably leads to increased thermal conductivity. The ability of moisture to penetrate into the internal open pore system at increased relative humidity significantly affects the temperature distribution as well as the thermal conductivity. The rate of change in thermal conductivity with moisture content is higher at higher initial moisture content. The lower the density of open-cell insulation materials, the higher the effect of moisture content on the thermal conductivity.

Density:

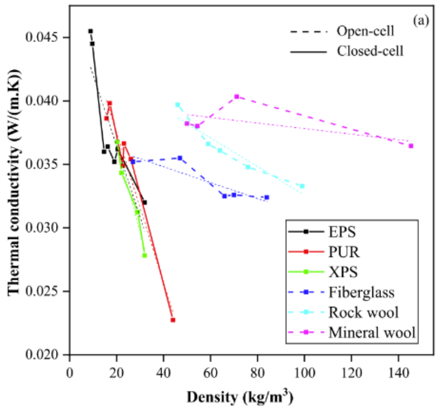


Figure 30 Insulation conductivity changes because of density changes [36].

Density can impact the conductivity of certain materials. Increasing foam material density reduces air content and the size of air inclusions, leading to a decrease in air convection and conduction, resulting in decreased thermal conductivity. Generally, higher density yields lower thermal conductivity, with specimens of lower densities experiencing faster increases in thermal conductivity.

Thickness:

While it's commonly believed that thicker insulation reduces heat transfer, it's important to note that thermal conductivity isn't thickness-dependent; rather, thickness affects thermal resistance.

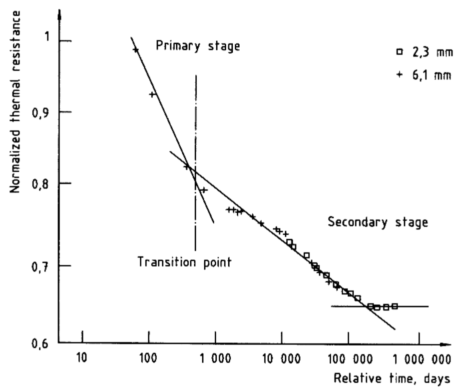


Figure A.3 — Normalized thermal resistance of two thicknesses of a rigid closed-cell plastic foam after application of the scaling factor

Figure 31 Normalized thermal resistance of two thicknesses of a rigid closed-cell plastic foam after application of the scaling factor [37]

Aging:

According to (Choi et al., 2018) and (British Standards Institution. & British Standards Institution., 1999) , aging alters material performance over time, with different mechanisms influencing the aging process of various insulations.

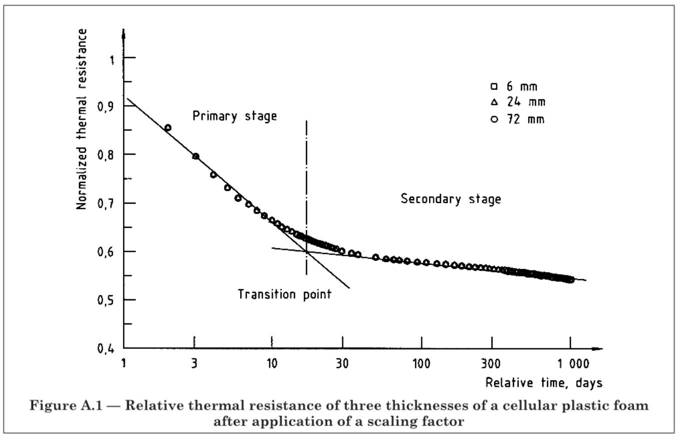


Figure 32 Relative thermal resistance of three thicknesses of cellular plastic foam after application of a scaling factor [37]

Foam materials typically undergo three aging stages. Closed-cell materials exhibit lower thermal conductivity at higher densities, with EPS, XPS, and PUR insulation materials showing a decreasing trend in thermal conductivity as density increases from 20 to 40 kg/m^3. This is because higher density implies smaller pores and less air volume, causing heat flow primarily through solid particle conduction, rendering convection and radiation effects insignificant. The variation in thermal conductivity values of EPS and PUR is attributed to differences in microstructure, porosity, and pore dimensions.

The aging process of foam materials can be approximated by separate linear fits to primary and secondary stages, with a transition zone playing a crucial role in analyzing thermal performance. The primary stage involves rapid diffusion of atmospheric gases into the foam, leading to a rapid decrease in thermal resistance. In the secondary stage, the blowing agent diffuses out of the foam at a slower rate, further decreasing thermal resistance, albeit at a slower pace than the primary stage. The final stage represents equilibrium, where the gas composition stabilizes, maintaining relatively constant thermal resistance. The duration of each stage depends on factors such as foam type, blowing agent, cell size, density, and environmental conditions.(British Standards Institution. & British Standards Institution., 1999).

Available data

The most typical way of testing the performance material over time is through slicing. The reason is to accelerate the aging process for testing purposes. By slicing the foam into thin pieces, the diffusion path for the gas is shortened, which speeds up this aging process significantly. This allows researchers to simulate long-term performance in a much shorter timeframe. Even though these are the standardized tests there is a lack of field

Another way of checking the material's performance is by measuring their performances over time. A study in Korea examined the long-term aging variation of building insulation materials, specifically expanded polystyrene and rigid polyurethane providing valuable insights into their durability and performance. It was found that the thermal resistance of these materials decreases over time, falling below Korean Standards (KS) performance standards within 50-150 days for polystyrene and about 1000 days for polyurethane. Expanded polystyrene showed a decrease in thermal resistance by 25.7% to 42.7%, and rigid polyurethane by 22.5% to 27.4%, over a period of approximately 5000 days.(Choi et al., 2018)

Specimens				Thermal resistance ($m^2 \cdot K \cdot W^{-1}$)					Density ($kg \cdot m^{-3}$)	Deterioration ratio (%)
				Initial value	100 days	1000 days	4000 days	5000 days		
Window	Expanded polystyrene	Type 1	Special	2.485	1.965	1.493	1.469	1.424	35.4	42.7
			1 class	2.221	1.687	1.368	1.367	1.338	32.3	35.6
		Type 2	Special	2.157	1.860	1.694	1.588	1.575	33.2	23.4
			2 class	1.984	1.698	1.566	1.471	1.472	30.5	20.6
	Rigid polyurethane	40 K		2.656	2.486	2.128	1.992	1.929	36.3	29.3
		50 K		2.613	2.476	2.128	2.048	2.024	46.8	23.7
Wall	Expanded polystyrene	Type 1	Special	2.498	2.004	1.497	1.454	1.452	35.4	42.1
			1 class	2.222	1.725	1.370	1.346	1.322	32.3	36.2
		Type 2	Special	2.158	1.825	1.486	1.535	1.535	33.2	25.1
			2 class	1.968	1.671	1.578	1.453	1.450	30.5	20.9
	Rigid polyurethane	40 K		2.661	2.522	2.094	1.913	1.902	38.9	30.6
		50 K		2.579	2.445	2.044	1.903	1.897	49.4	27.5

Figure 33 Samples of insulation degradation taken by the paper(Bae et al., 2022)

PART 2

5. IMPLEMENTATION METHODOLOGY

Throughout the literature review, the theoretical background needed to start with the development of this thesis was presented.

In the first part, the basic theory of renovation techniques was investigated, and the conclusions showed that the exterior envelope insulation, together with the windows, constitute an important factor to the overall energy performance of the building. The change of the building's performance was also investigated through research of the factors that affect the conductivity performance of the plastic foam insulations. It was determined that even though there are various reasons why the conductivity can change, one important aspect was the aging of the material. By assuming that the insulation material properties are the only thing that will change through time, it is possible to speculate how the building's performance will change as a relation with time. However, the other aspects should also be considered and implemented in later research.

Additionally, the theoretical foundations of Markov Decision Process and Planning algorithms were examined.

Building upon these theoretical frameworks, the subsequent sections will integrate these theories to develop a workflow aimed at determining the optimal timing for retrofitting a building.

Value iteration serves as a foundational concept in reinforcement learning without being categorized as a RL method but as a predecessor. It was chosen as the algorithm to be used in the optimization process as it was easy to use, required less hyperparameters and since it was a model-based approach provided a more direct understanding of the dynamics of the environment and the expected results.

Given the investigative nature of this project, various techniques were employed to conduct energy calculations, alongside the iterative development and testing of the Markov Decision Process (MDP) environment. While the outcomes of the initial stages may not directly impact the final results, they offer valuable insights into the methodological challenges encountered, which can inform future research endeavours.

To ensure coherence, the project's development was structured into two distinct stages. The first stage encapsulates the genesis of the project, beginning with the conceptualization of the initial idea, which was presented at P3. Subsequently, the second stage marked the development and refinement of this idea, leading to the creation of the final version which was utilized for testing the environment dynamics.

In the following chapters the final version of the methodology will be presented. However, in case that it is needed, the methodology of stage one placed in the appendix, providing further information of the process that was followed for the development of the project.

4.1. General Workflow

To start solving the problem, the most important aspect was to define it in a way easy for everyone to understand and follow. Even more, from there, the basic workflow needed to solve the problem can be developed. The original question of the thesis was ***"how to optimize the planning of a retrofitting to be cost optimal in the span of the building's lifetime that was assumed to be 60 years".***

Based on the literature review, it was revealed that there are a lot of different factors that have to be considered like the house typology, the retrofitting measure, the energy bills and the method of solving the problem. Based on those facts the problem was formulated in a simple way that could be addressed.

Toy problem formulation:

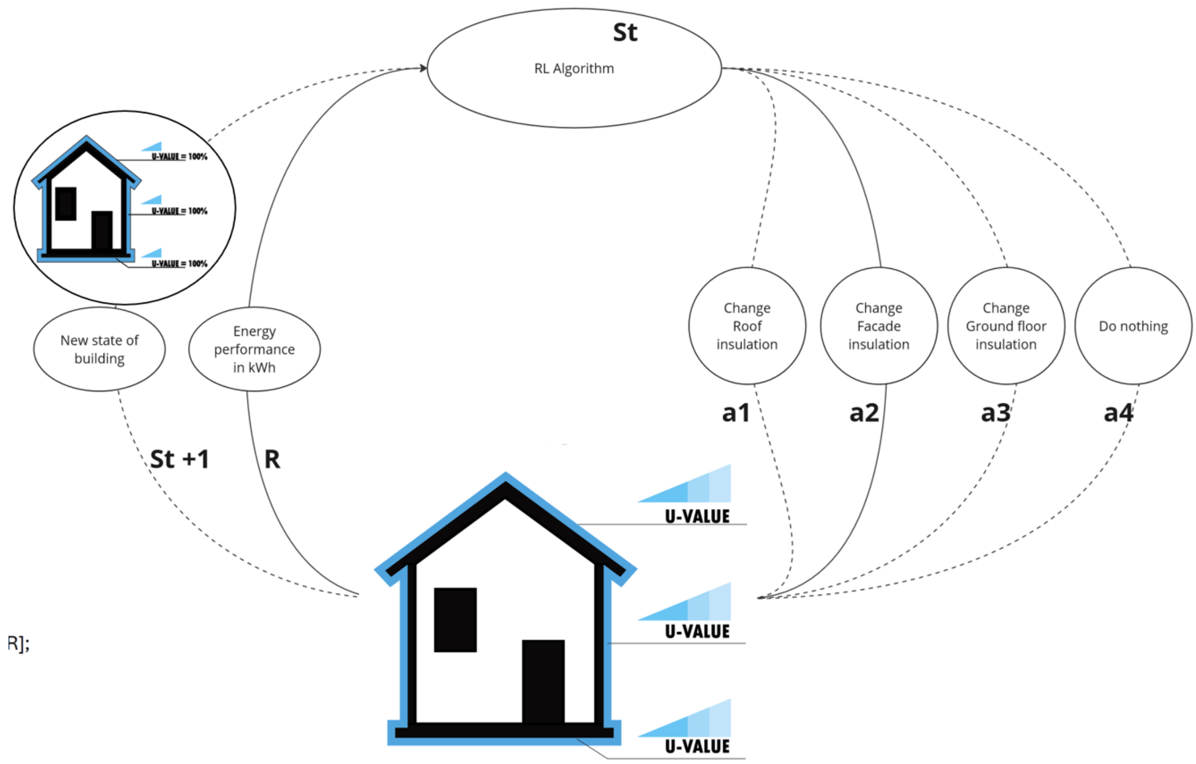


Figure 34 Interaction of the agent with the environment according to the problem formulation (own work)

We have a residential building case study. In this building, 3 main components are degrading over time with uncertain rates. With every degradation percentage, the energy consumption of the building will increase. If the energy performance decreases, the amount spent on electrical bill increases. The owner of the house is trying to spend as little money as possible. By changing the envelop components to new ones, the performance of the house can better. The goal is to choose when to renew and what in order to spend the least amount of money and have the greatest house quality possible.

Objective function

In this case, the objective function must describe the minimization of costs that can accumulate over 60 years. This is based on the investment costs of actions (including labour and other types of costs) and the running energy costs following each retrofitting action. The objective function can be represented as:

$$minCost = \sum_{t=0}^{60}(IC_t + EC_t)$$

where IC_t are the investment costs at time t (including labour and other types of costs) and $EC_t$are the running energy costs after each retrofitting action.

Based on this definition of the problem, the MDP that defined the problem was created as a tuple of (S,A,P,R,γ) where.

- States **S** are all the possible states of deterioration the different components of the building can reach. The state space will be described as the tuple of time, components' degradation states, age of each component.
- Actions **A** are described as the actions of changing the insulation of each component or the combination of actions, in order to bring the building in a better state of degradation.
- Probabilities **P** were defined as all the possible states of degradation that the building would reach given that it was in a certain state and a certain action was taken.
- Rewards **R** was defined as the costs that could be expected given the state of degradation and the action. Basically, that meant that the state of degradation would result to a certain energy demand which would result in energy bills, and the action would be connected to the investment costs of taking an action.
- The discount factor **γ** was decided to describe the growth rate of the economy and the depreciation of the costs that could be expected in the future.

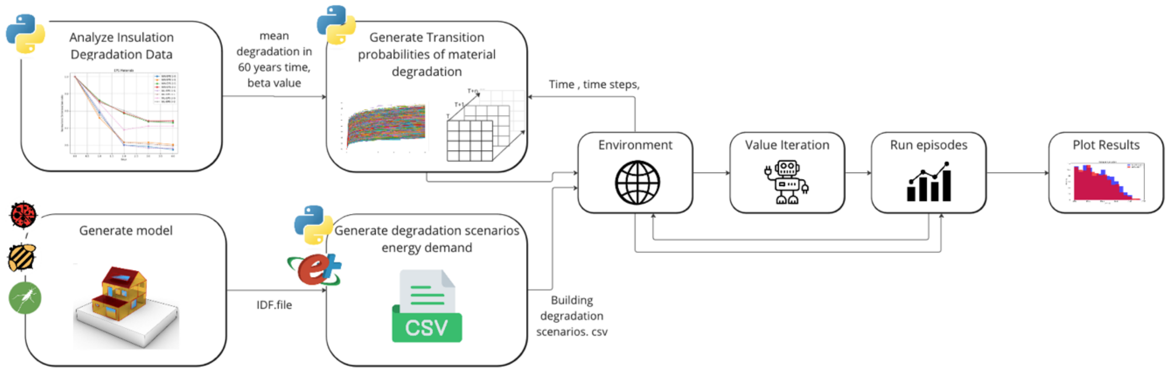


Figure 36 Final workflow (own work)

Based on observations of different stages of trial and error the final methodology was created. The final script workflow, as presented in the figure above was divided into two major parts: The data generation needed to simulate the environment and the optimization. Based on the EPS degradation data that was found during the literature review, the transition probabilities were generated in a separate script and imported into the environment to calculate the state transition matrices. The house typology was also created using Grasshopper and Ladybug, exported to IDF format. Eppy library was used in Python in order to run the different scenarios of degradation. The results were exported in csv format and imported in the environment to calculate the rewards of each state. In order to gain time, the environment pre-generated the states, rewards, transitions into NumPy arrays that were then used to run the value iteration and the episodes scripts. A final script was created to plot the policies and the results of the different episodes in order to be analysed.

4.2. Case study

In order to generate the energy demand, a building case study was needed. The building was a detached residence built in 1971 in Riel.

According (Ministerie van Binnenlandse Zaken en Koninkrijksrelaties, 2022) detached houses built from 1965 to 1974 represent 1.7% of the Dutch housing stock with 128,500 houses. Homes from this period are all owner-occupied. The homes are often traditionally built with sand-lime brick bearing walls and concrete floors.

Since 1965 there have been requirements for the energy quality of homes. Yet by today's new construction standards, the houses were not very well insulated. Natural ventilation was used to ventilate these houses. Central heating was used more and more extensively in this type of home over the years.

The original envelope values for the exterior walls were taken from(Ministerie van Binnenlandse Zaken en Koninkrijksrelaties, 2022). Even though it was stated that some kind of insulation might have been used, it was assumed that the construction details and their thermal resistance didn't contain any insulation.

Figure 39 Ground floor layout [40]

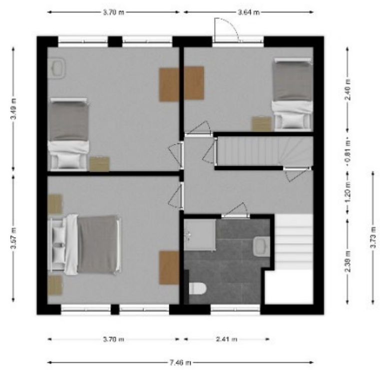


Figure 38 First floor layout [40]

Figure 37 Second floor layout [40]

Detached house 1965 - 1974		current
BUILDING	area [m2]	
first floor	112,90	Rc 0.17
closed facade	126,68	Rc 0.43
pitched roof	94,50	Rc 0.86
flat roof	24,56	Rc 0.86
window	38,08	U 2,90
door	8,10	U 3,40
crack sealing (q;v10)		flat
INSTALLATIONS		
ventilation type		all-natural
heat recovery		no
space heating		HR107 boiler
hot water		Gas combination appliance with gas certification HR/CW
PV panels (m^2)		0,00
ENERGY PERFORMANCE		
standard (kWh/m2)		87,8
heat demand (kWh/m2)		204,9

Figure 40 Aspects of the house typology as taken from(Ministerie van Binnenlandse Zaken en Koninkrijksrelaties, 2022)

Table 8 : Case study's 2 construction details			
Aspect	**Details**	**Area (m^2)**	**R values (m^2K/W)**
Wall	Not stated	267.52	0.43*

Roof	Pitched roof	89.41	0.86*
Ground Floor	Crawl space ground floor without insulation	75.40	0.17*
Windows U-value	Single glazing	37.6	2.90
*non insulated surface values			

Table 9 :EPS Insulation Properties		
Conductivity (W/mK)	**Sp. Heat Capacity (J/kgK)**	**Density (kg/m3)**
0.035	1400	25

4.3. Energy simulations

The energy simulation was done using Ladybug plug-ins for Grasshopper. The process followed the workflow as provided by Philipp Galvan (*Honeybee "ENERGY" Part 1 | Setting the Scene | Ladybug Tools 1.4.0 - YouTube*, n.d.)

The building information was exported as an idf file which was then imported in python. The Eppy library in conjunction with the EnergyPlus software was used in order to run iteratively thought all the possible 27 scenarios of different degradation states (0, 20% and 40%) of the roof, facade and ground floor insulation. The results were saved in a csv dataframe and imported in the environment code for the rewards to be calculated. With this technique, the amount of time required to run through all the scenarios amounted to total execution time of 201.91 seconds and to 118.69 seconds to read and store the results into the csv file.

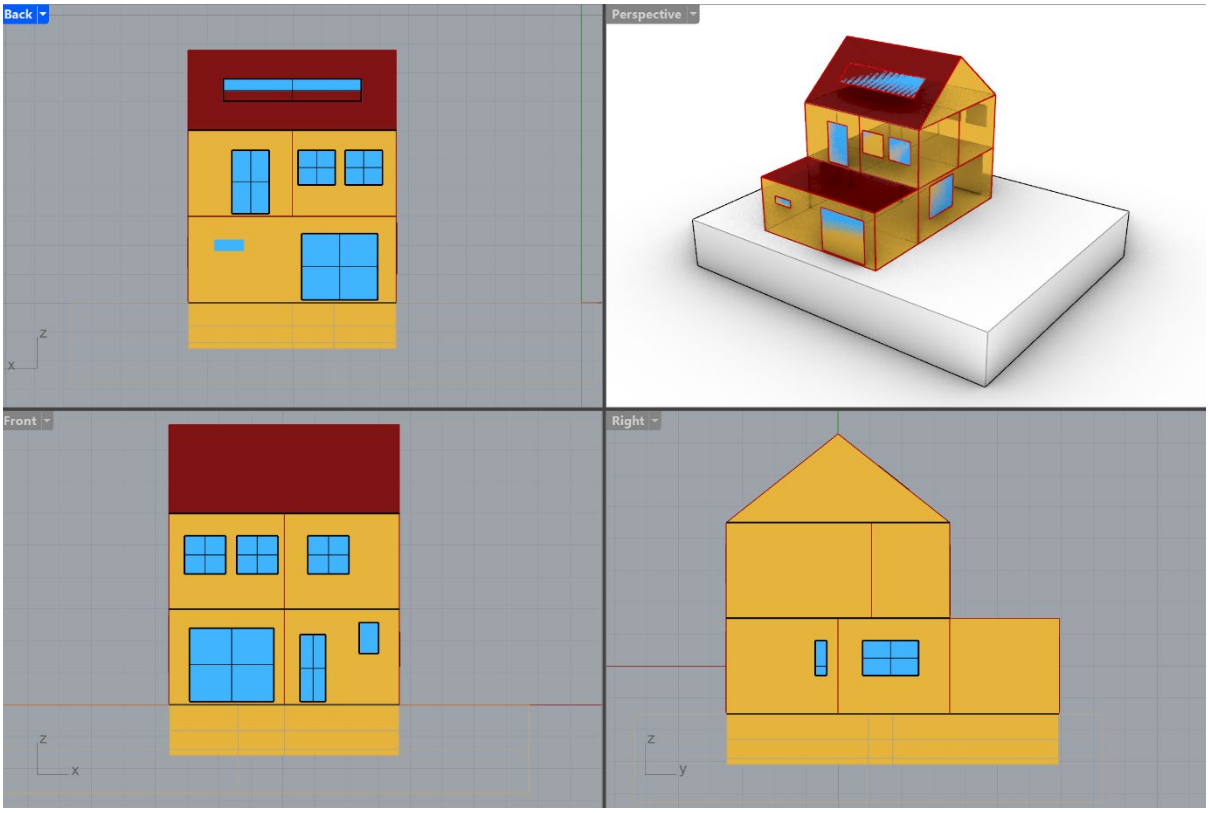


Figure 41 Model of the house developed in Grasshopper (own work)

According to the simulations, the household's difference of energy consumption between the best-case scenario and the worst case scenario amounted to increase of about 5.59% in energy consumption The changes reflected on the financial costs which amounted to 816 euros difference. The small changes in the energy performance can be explained over the fact that the windows and infiltration rate didn't change. Those parameters amounted also on the degradation simulation scenarios observed in literature. The different degradation scenarios can be found in table 10 of the appendix.

4.4. State space

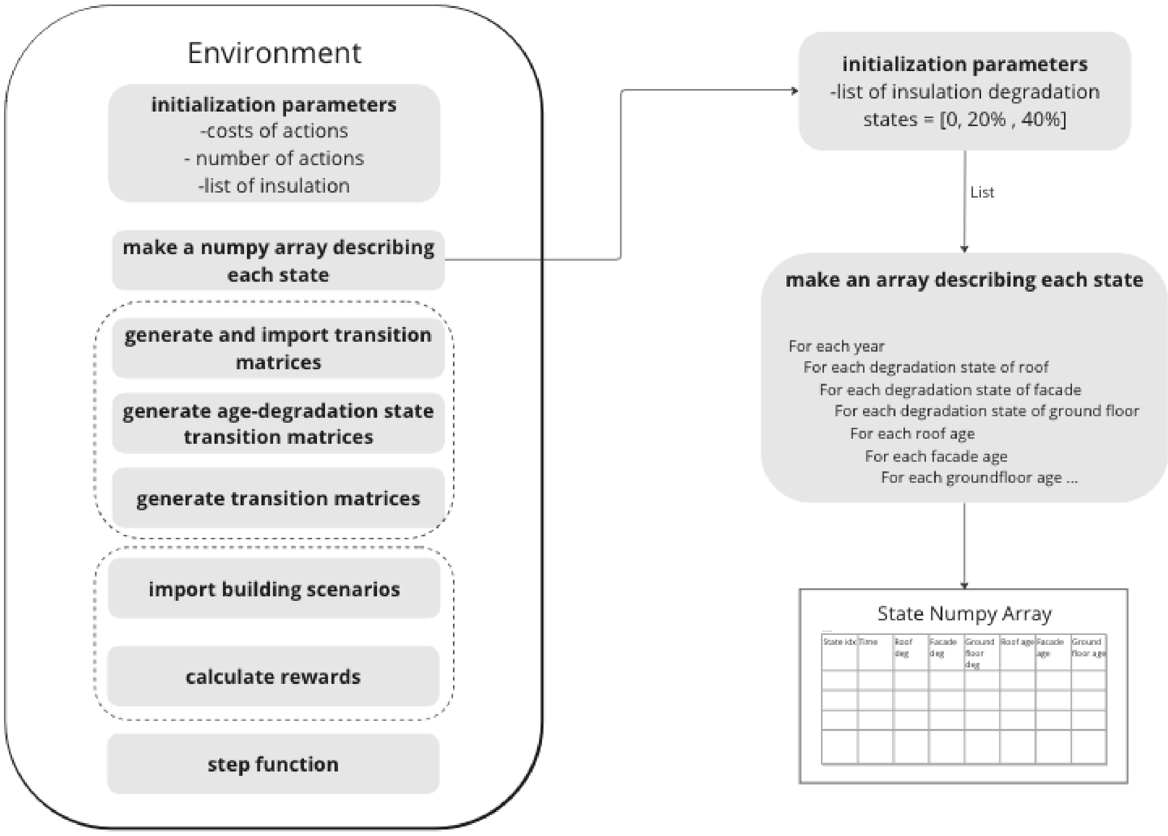


Figure 42 Graph depicting the workflow of creating the state space

The newly constructed state space was delineated as:

[time] x [roof degradation percentage]x [facade degradation percentage] x [ground floor degradation percentage] x [roof age] x [facade age] x [ground floor age]

In a more mathematical way this can be represented as :

$$S = \left\{(t, r, f, g, r_{age}, f_{age}, g_{age}) | t \in [0,5,10, \dots ,60], r, f, g \in \left\{0\%, 20\%, 50\% \right\}, r_{age}, f_{age}, g_{age} \in [0, t] \cap \{0,5,10, \dots ,60\}\right\}$$

In this equation:

- t represents the elapsed years, constrained between 0 and 60, in 5 years time steps.
- r, f, g denote the degradation levels of the roof, facade, and ground floor, respectively, each ranging from 0%, 20%, to 40%.
- $r_{age}, f_{age}, g_{age}$ indicate the ages of the insulation material for the roof, facade, and ground floor, respectively. These ages range from 0 to tt, ensuring they are always equal to or smaller than the elapsed time, and they increase in steps of 5 years up to a maximum of 60 years.

The size of the state space fluctuated according with the defined time and time step parameters[11].

Time needed to be incorporated in order to avoid an infinite horizon problem[12] and in order to be able to pinpoint the time that each action should be taken. Age needed to be incorporated because the transition probabilities were non-stationary (e.g. they changed with age). This meant that at the age 1 the Roof component for example had 15% probability of staying in a degradation state of 0%, however, at the age 10, this probability might have turned to be 60%. In order to be able to connect the probabilities with the state space, it was important that the ages of each material would need to be incorporated in the state space. An example of the state space can be seen in table 11 of the appendix.

A logical argument was used to diminish the state spaces: Since the original states would only be described as states where the building is new or just retrofitted, the insulation material ages could not be greater than the elapsed time. Equally for all the future states, the ages of each material could only be equal or smaller than the time. Based on this argument, all the states that had ages bigger than the year-time were deleted from the state space. This diminished the state space to 223587 states[13].

---

[11] Since it was not important to check the material each year, different time steps were tested to discretize the time and age states and make the state space smaller. Time steps of 10 years provided a quicker run of the code but allowed for a large time to be passed between actions. Time steps of 5 years allowed detailed interpretation of the actions needed to be taken in the span of 60 years. The maximum state space considered encompassed 559,872 states, determined by a 60-year lifespan and a 5-year time step. However, with the number of actions being also raised, this created problems for the value iteration to run. Traversing through more than 500,000 states times 8 actions led to 4,478,976 iterations, demanding substantial computational resources and approximately 111 hours(taken that it would need 7 iterations to converge).

[12] Essentially, the infinite horizon problem arises when the model fails to account for time, making it challenging to adapt to changing environments or unforeseen events. In this scenario, a state initially classified as favorable may deteriorate over time upon reassessment, causing disparities between expected and actual outcomes.

[13] Further arguments could be made that a material could not be in 10% degradation when it had exceeded a certain threshold, however, since there was a probability involved, it was decided to let the states as they were and not diminish them further.

4.5. Actions

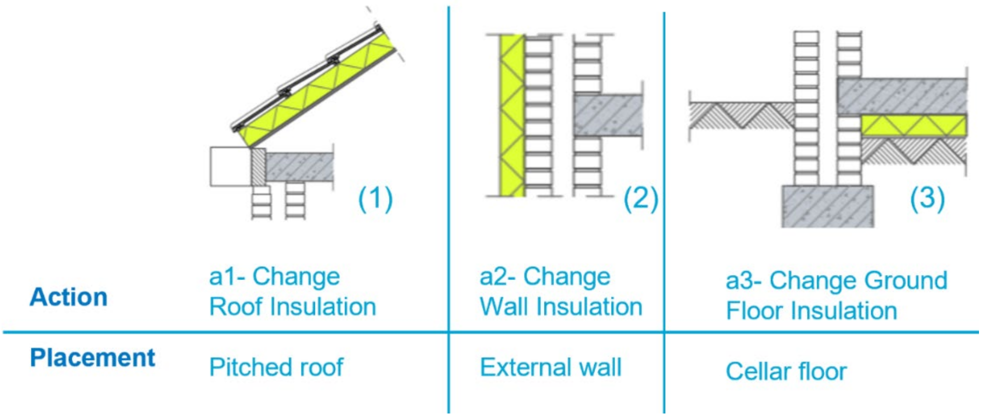


Figure 43 Details of insulation placement on roof, exterior wall and ground floor , taken from (Sewnath, 2024)

In total 8 actions could be done in each state. Still all the actions just amounted to changing the insulation materials. The 'do nothing' action meant that the building was left to deteriorate based on the transition probabilities which described the degradation curve of the materials. The prices for the materials were taken from (*Kostenkentallen | RVO*, n.d.) .

Table 13 : Actions and their costs		
Action	Interpretation	Costs (euros)
0	Do nothing	0
1	Change Roof Insulation	13407
2	Change Façade Insulation	43533
3	Change Cellar Floor Insulation	2614
4	Change Roof and Cellar	16021
5	Change Facade and Cellar	46147
6	Change Roof and Facade	56940
7	Change All	59554

Labor costs plus , costs of installation and costs of the materials were included in the investment costs of each action. Since no EPS insulation was provided for pitched roofs, the costs of PIR[14] insulation were taken into account instead. The measures can be seen below:

Table 12 : Chosen measures of retrofitting						
Name	Info	Placement	Width (mm)	Rd value	RC value (m2 K/W)	Price euros per m2
Roof insulation						
WB212b – PIR renovatie dakplaten		Pitched roof	175	6.45	8.3	149.96
Ground floor insulation						
WB003b -EPS		Crawl space floor	300	-	2.6	34.68
Façade insulation						
WB008b -EPS isolation	Decorative plaster finishing	Exterior wall	100	-	2.6	162.73

4.6. Transition probabilities

Nonstationary probabilities are necessary to accurately model the time-dependent and evolving nature of material degradation, such as the deterioration of insulation in buildings. Traditional stationary models assume that the transition probabilities between states remain constant over time. However, factors like age, heat, moisture, and other environmental conditions significantly influence the degradation process. For instance, a piece of insulation might degrade at different rates depending on its current age and the specific conditions it has been exposed to. By using nonstationary probabilities, we can account for these varying rates and more accurately predict future states of material degradation.

[14] PIR is also plastic foam insulation. In this thesis it is assumed that the material is degrading with the same rate as the EPS. However, a separate research should be conducted for its exact properties.

Since the state space is defined as:

[time]×[roof degradation percentage]×[facade degradation percentage]×[ground floor degradation percent age]×[roof age]×[facade age]×[ground floor age]

We can define the transition from one state to the will be dependent on the state that we are in , the action that will be taken, and the ages of the insulation materials at the current time step. It can be expressed as the joint probability of the state $(\mathrm{t}, \mathrm{d}_r, \mathrm{d}_f, \mathrm{d}_g, \mathrm{a}_r, \mathrm{a}_f, \mathrm{a}_g)$ transitioning to $(\mathrm{t} + \Delta\mathrm{t}, \mathrm{d}'_r, \mathrm{d}'_f, \mathrm{d}'_g, \mathrm{a}_r + \Delta\mathrm{t}, \mathrm{a}_f + \Delta\mathrm{t}, \mathrm{a}_g + \Delta\mathrm{t})$:

$$\mathrm{P}((\mathrm{t}, \mathrm{d}_r, \mathrm{d}_f, \mathrm{d}_g, \mathrm{a}_r, \mathrm{a}_f, \mathrm{a}_g) \rightarrow (\mathrm{t} + \Delta\mathrm{t}, \mathrm{d}'_r, \mathrm{d}'_f, \mathrm{d}'_g, \mathrm{a}_r + \Delta\mathrm{t}, \mathrm{a}_f + \Delta\mathrm{t}, \mathrm{a}_g + \Delta\mathrm{t})$$

- t as the current time
- $\mathrm{d}_r, \mathrm{d}_f, \mathrm{d}_g$ as the current degradation percentages of the roof, facade, and ground floor respectively
- $\mathrm{a}_r, \mathrm{a}_f, \mathrm{a}_g$ as the current ages of the roof, facade, and ground floor respectively
- $\mathrm{t} + \Delta\mathrm{t}$ as the next time step
- $\mathrm{a}_r + \Delta\mathrm{t}, \mathrm{a}_f + \Delta\mathrm{t}, \mathrm{a}_g + \Delta\mathrm{t}$ as the ages at the next time step
- $\mathrm{d}'_r, \mathrm{d}'_f, \mathrm{d}'_g$ as the degradation percentages at the next time step

<u>Nonstationary probabilities generation methodology</u>

To achieve this, the methodology proposed by (Saifullah et al., n.d.) was employed, which involves using a gamma process to derive these nonstationary probabilities.

Nonstationary transition probabilities account for the fact that deterioration rates are not constant but vary based on the material's age. They improve the ability to plan maintenance and rehabilitation activities by providing a more realistic forecast of the envelope's performance over time. Based on (Saifullah et al., n.d.), nonstationary transition probabilities can be created using a gamma process model. More information about the gamma process can be found in the appendix.

4.6.1. Data analysis

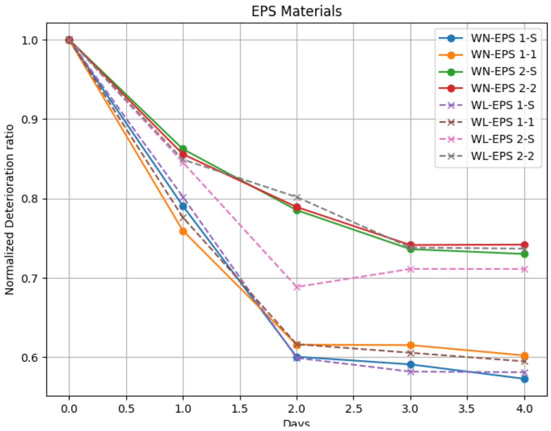


Figure 44 Graph depicting the thermal resistance performance degradation of the different material samples. WN means the material was placed on a window surface, internal side and WL on the internal side of the wall. (Own work)

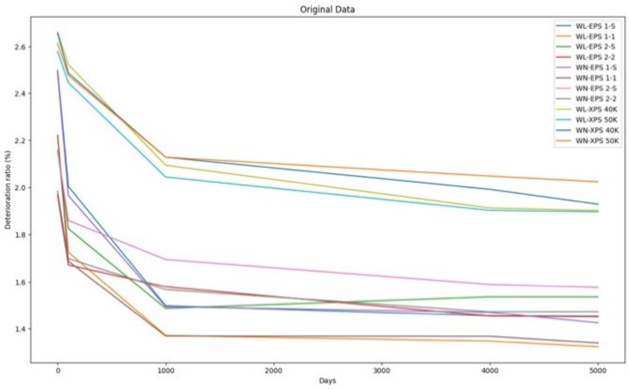


Figure 45 Graph depicting the initial thermal resistance values and their end values in m^2K/W (Own work).

To accurately model the degradation probabilities of insulation materials over time existing data on insulation materials were gathered and analyzed. Long-term data spanning 5000 days from EPS and XPS insulation samples placed on walls and windows showed that EPS exhibited a significant decrease in thermal resistance, between 25.7% and 42.7%(Choi et al., 2018). Based on the material's popularity in the European market led to the selection of EPS for the study.

Further analysis indicated that the degradation of EPS insulation did not significantly differ based on placement (wall or window), but the sample size was too small to draw definitive conclusions.

The degradation data was then fitted to a logarithmic curve, which allowed for the extrapolation of mean degradation over extended periods. Based on the extrapolation, the mean degradation of the material in 60 years was found to be 43%

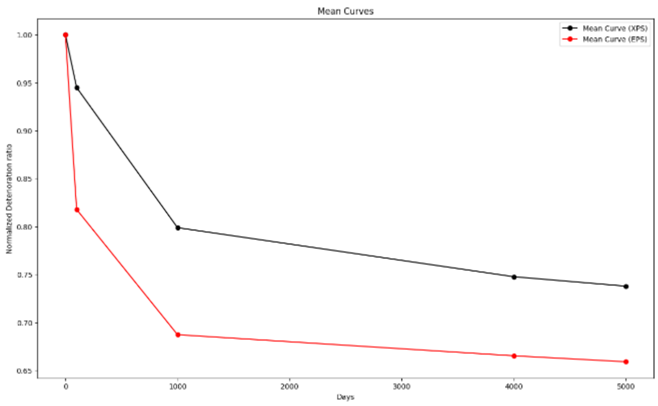


Figure 47 Graph depicting the mean degradation curves for XPS (black) and EPS (red) insulations over 5000 days span (Own work)

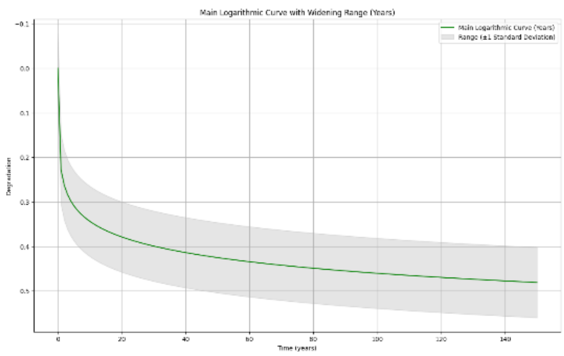


Figure 46 Fitted logarithmic curve of the EPS thermal resistance degradation over 140 years span.(own work)

4.6.2. Transition probability generation using the custom gamma function

In order to generate the transition probabilities the a new script was created which generated random degradation curves based on a gamma distribution curve. This meant that a lot of different samples were generated that followed a certain degradation curvature

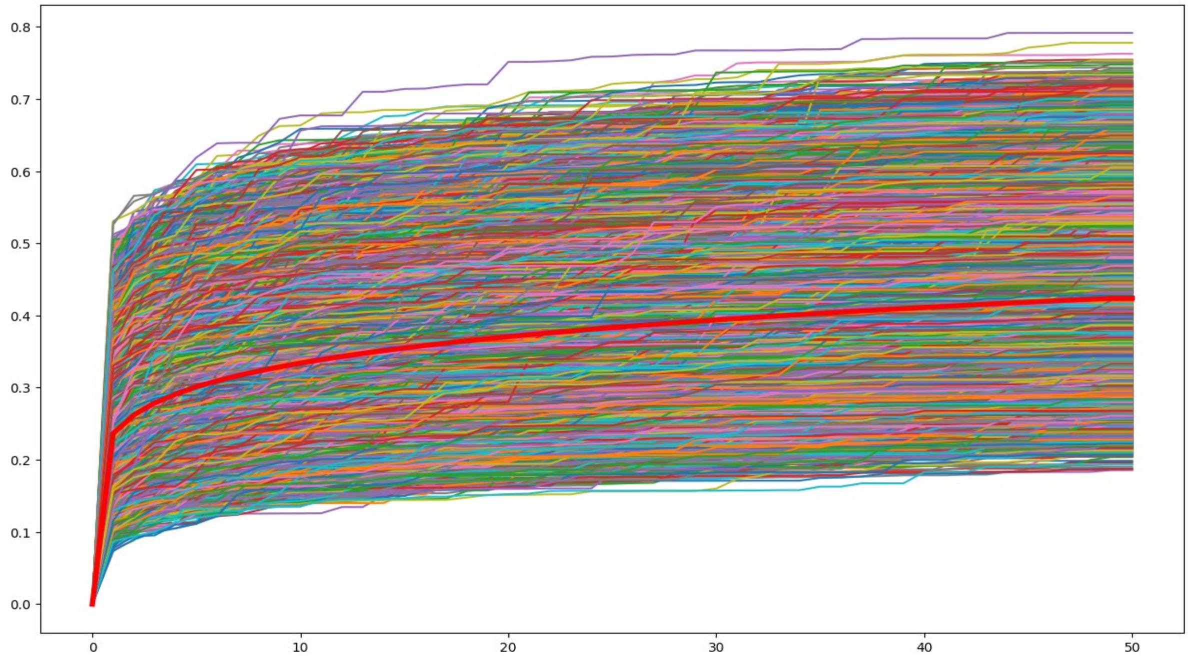


Figure 48 Plot of the sample degradation curves generated using the gamma function. The thick red line depicts the mean degradation curve of the material (own work)

but with some randomness.

A custom gamma curve function provided by Charalampos Andriotis was manually fitted to match the shape of the logarithmic curve. The below parameters were used to fit the gamma distribution to the material degradation data.

```
def custom_gamma(a, b, t, beta):
        x = np.random.gamma(shape=a * t**beta - a * (t - 1)**beta, scale =
b)
        return x
```

Figure 49 custom gamma function that was based in the MATLAB script provided by Charalampos Andriotis and translated into python. , where 'a' represents the shape parameter and 'b' represents the scale parameter. The beta parameter influences the curvature of the degradation process over time, and T represents the total time period for which the degradation is modeled.

Table 14 : Initial parameters of the gamma distribution script		
Parameter	Value	Description
mean	0.4343905423150753	Describes the final degradation that the material will have at the end of the time period T

Beta	0.15	Describes the curvature and direction of the degradation curve. For a steeper curve, β should be lower than 1
std	0.15 * mean	Standard deviation, representing the dispersion of the degradation values
T	60	Number of years (time period)
N	1000000	Number of realizations (simulations)
variance	std * std	Variance, representing the square of the standard deviation
b	variance / mean	Scale parameter of the gamma distribution
a	(mean * mean / variance) / (T^beta)	Shape parameter of the gamma distribution, adjusted by T^{β}

The methodology continued with generating 10 degradation states, each representing a 10% degradation over 50 years. However, for practical purposes, the model was adjusted to explore three states of material degradation with larger time steps (5 or 10 years). The random sampling generated degradation percentage and insulation age pairs, with a million sequences created to determine transition probabilities. The sampled data were categorized into 3 degradation states of <5%,<20% and 100% , and matrices were created to represent transition probabilities for each year.

Based on the 60 years with 5 time step increments twelve 3 x 3 probability matrices were created. The graph depicting the methodology of generating transition matrices can be found in the appendix.

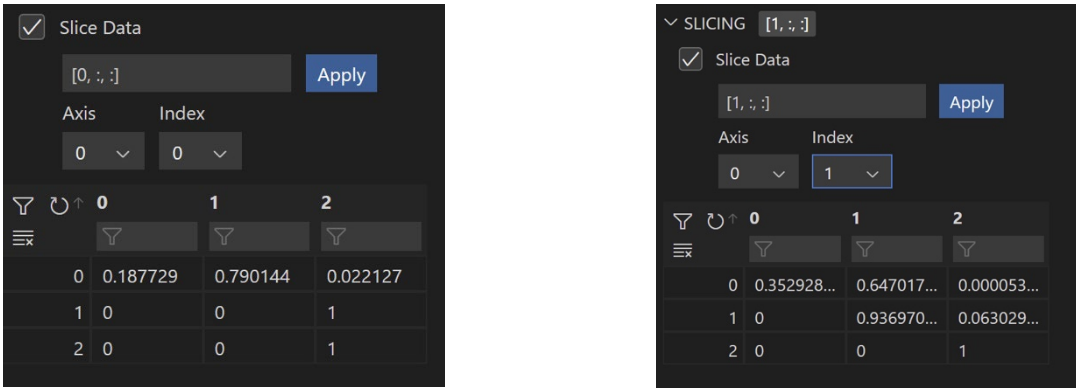
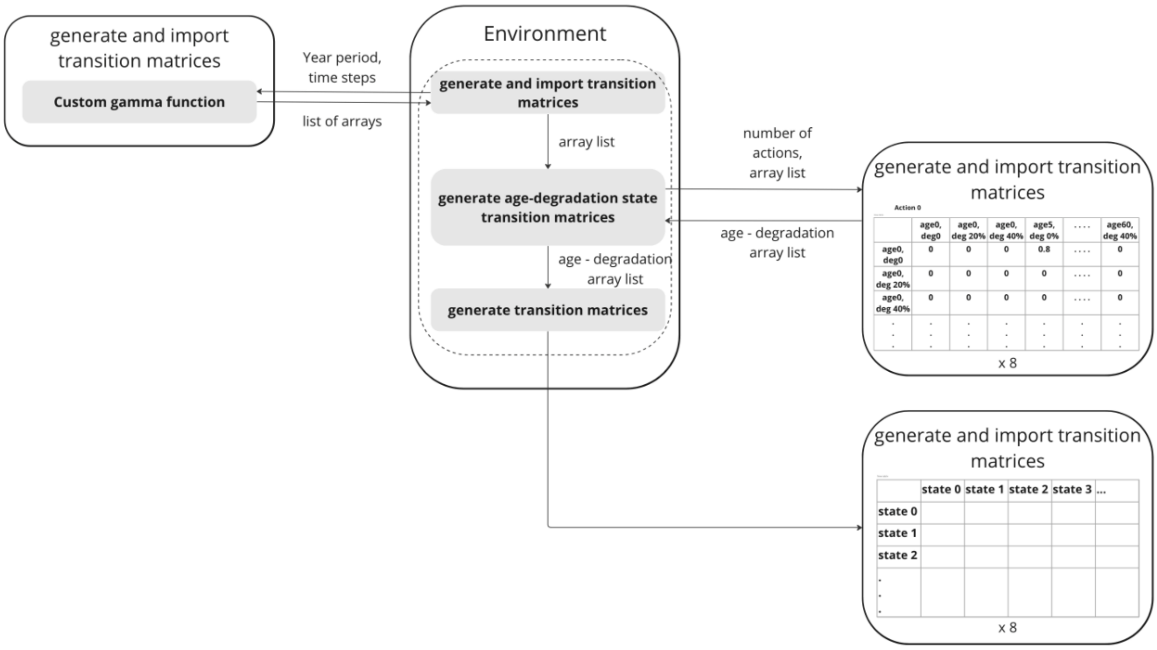


Figure 50 Examples of the NumPy arrays depicting the transition probabilities for the year 0 and 5 respectively

The calculation of state transition probabilities was carried out through a structured three-step approach:

Figure 51 Workflow of generating the state to state transition matrices (own work)



1. The initial step involved importing the pre-generated transition probabilities into the computational environment. These probabilities were stored as NumPy arrays, enabling efficient manipulation and computation.
2. Transition matrices were constructed to represent the probabilities of degradation percentage and age state pairs transitioning to other states. Each matrix element P_{ij} represented the probability of transitioning from state i to state j. The states in this context are defined by a combination of degradation percentage and age.
3. Using the transition matrices, the probability of each state pair transitioning to any other state pair was computed. This process was iteratively applied to all states in the environment. The matrices stored as separate transition matrices for each action into sparse matrices to be used during the training phase.
4. The process of calculating first the degradation – age pairs and then the state-to-state transition plus the use sparse matrices spead up considerably the process. By pre-calculating the transition probabilities, also sped up the value iteration considerably.

4.7. Discount factor

By applying a discount factor of 0.97 per time step, the MDP framework can account for the time value of money, ensuring that future rewards are appropriately discounted and aligned with real-world financial considerations.

However, since states do not align precisely with each year but occur according to discrete time steps, the discount factor was adjusted accordingly. The final discount factor for each time step was calculated using the equation, based in the idea that the growth rate would be 3% each year:

$$\gamma = 0.97^{time\ step}$$

4.8. Environment

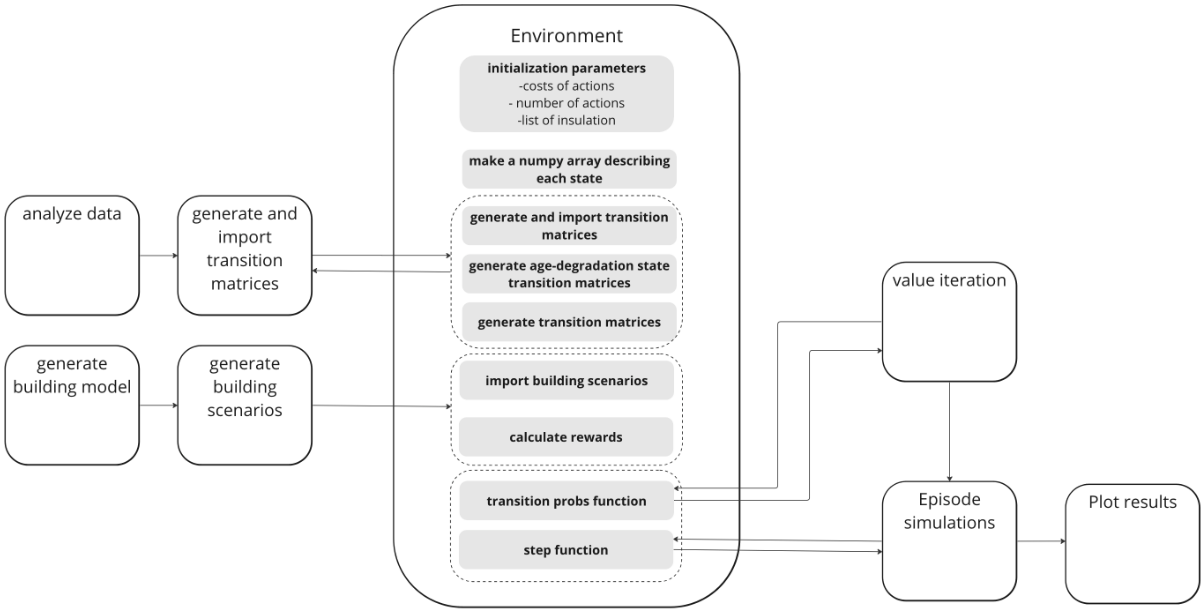


Figure 52 Graph depicting the dynamic relation between the environment and the rest of the scripts

Based on the above-mentioned elements that described the problem as an MDP, a environment was created that simulated the world dynamics. Based on time, time step, beta values and amount of sample inputs from the environment, the transition probabilities were generated and outputted in the form of NumPy arrays.

The building model, exported in idf format was placed as input in a script that run iteratively to create the different degradation scenarios using the Eppy library.

The VI optimization algorithm received as NumPy arrays all the possible states, actions and rewards and calculated the optimal policy before exporting it to the final script that used the policy to run episodes and plot the results.

4.9. Value Iteration

In a computational context, as the state space grows larger, computational complexity increases, affecting the efficiency of algorithms like value iteration. To address the challenges posed by a significant state space, the delta threshold was adjusted, transitioning from e-20 to e-10, to control the precision of value iteration. Additionally, rewards for each state space were precalculated, reducing the computational load by combining calculations within the q-value and value function computation loop, thus optimizing time efficiency. Despite these optimizations, further analysis is deemed necessary to identify and mitigate other bottlenecks hindering the process.

For example, with approximately 230,000 states and 8 actions, the mean time for finding the optimal policy exceeded 96 hours, illustrating the computational demands of the task.

Regarding episodes, their size was augmented to 1000000 to ensure a closer variance between the optimal policy and the benchmarking policy (Do-nothing policy). This adjustment allowed for more accurate evaluation and comparison of policies but added to the computational demand.

Different reward and house model scenarios were simulated to determine the parameters that affect the problem formulation dynamics.

6. TESTS

Based on the definition of the environment, a series of tests were conducted to understand the dynamics and understand the optimal policies that were given. In the beginning the initial results were analysed in order to draw conclusions. However, since the simulation was quite computationally demanding, the state was diminished. If the initial results of the bigger state space can be found in the appendix. Below, the results of the diminished state space can be found, together with tests run to determine different dynamics of the environment.

6.1. Initial test

Since the algorithm required a very big amount of time to run through all the state and action space, creating a bottleneck of conducting more experiments and analyzing the results, the state space was diminished by increasing the time step to 10 years and the episodes were rerun. In order to understand better the dynamics of the environment, the return of the episodes was given without discount.

As result, the algorithm gave as an optimal policy a do-nothing policy with non discounted return of 775.935 euros for a 60 year period. This of course meant that the costs of retrofitting the building overpassed by far the costs of gains that could be expected from the energy performance betterment. In the following diagram the plotted policy provided the actions the percentages of actions to be expected in every time step, with in this case amounted to 100% 'Do nothing' in the span of the observed period.

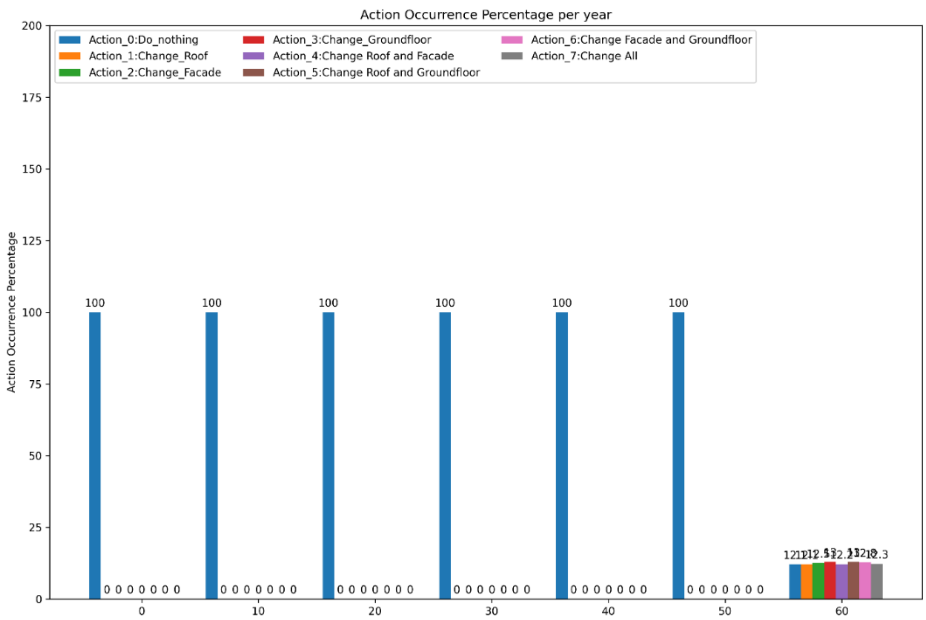


Figure 53 Plot depicting the optimal policy. On the horizontal axis we have the time steps. On the vertical axis we have the occurrence of a specific action given all possible states for that time step. We visualize this for all possible actions per time step. (Own work)

One thing that could be observed is the actions that are recommended at the end of the optimization period (year 60). These states, which are called absorbing states since the agent cannot transition to anywhere from there, receive zero rewards, positive or negative and signify the end of the optimization period. The actions that are recommended by the algorithm in those states are not valid but are one indication that the value iteration is working correctly.

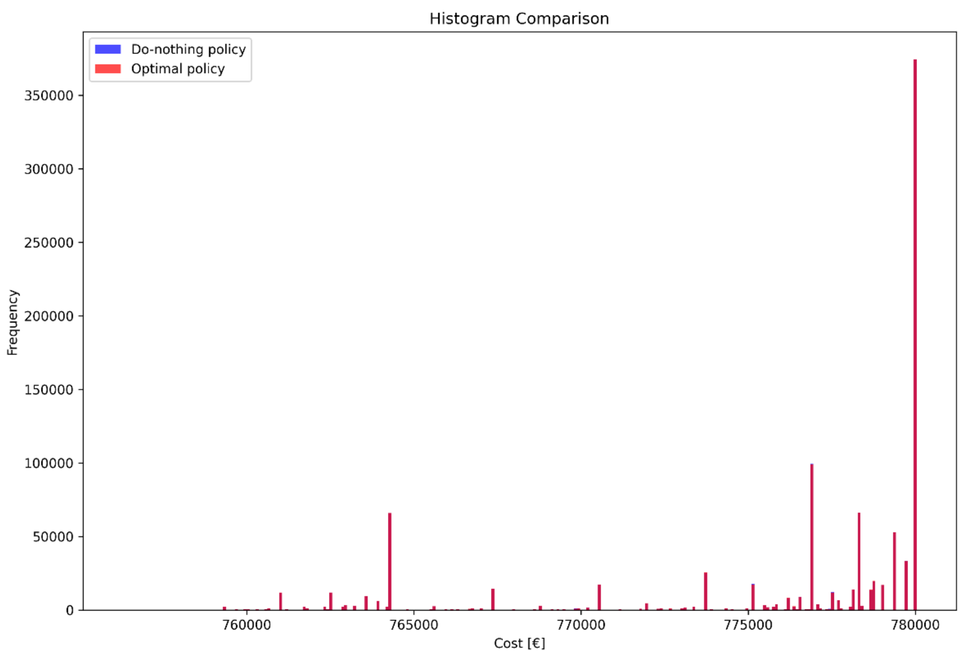


Figure 54 Histogram comparison between do nothing policy an optimal policy depicting the return amount frequency of one million episodes. As the optimal policy matches the do nothing policy, the histograms overlap. (Own work)

In order to understand what the do nothing policy meant and compare it with an optimal policy , a histogram depicting the Return of the generated episodes was created. The plot showed the frequency that the returned costs at the end of the optimization period would appear. In this case, since both policies where the same, the return of the optimal and the do nothing policy closely overlapped.

The do nothing policy meant that the house was let to deteriorate over time. Based on samples from the episodes that were run, it can be observed that the building's energy demand would reach the 174 kWh/m^2 right in the first ten years and require to pay around 150.000 euros for each decade of taking no action[15]. Two points can be observed that do not match the rest: The initial state of 165 kWh/m^2 energy demand, which starts from a much lower expenditure point. This is happening because the costs of the initial state amount only for the first year of expenses. In the rest of the states, the rewards amount for the energy demand that the owner had to pay, assuming that his house would jump the next exact year to the next state. In the last state, the rewards are becoming null, and so the costs are dropped to 0.

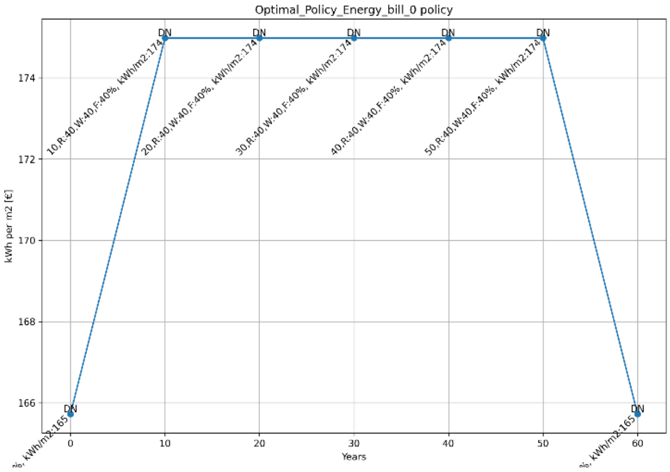


Figure 55 Episode depicting the energy demand states visited by the algorithm (Own work)

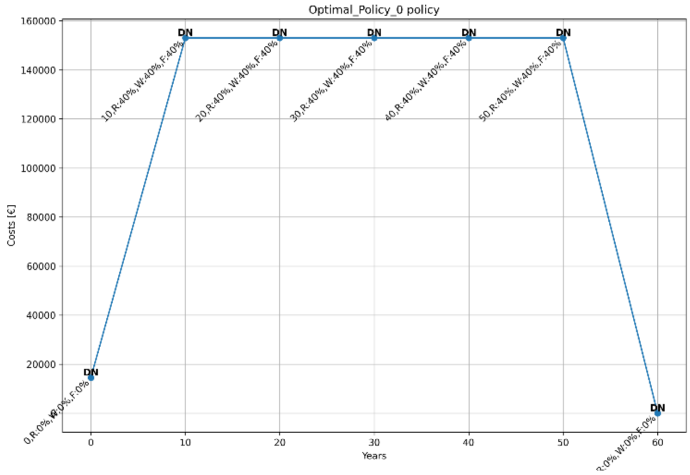


Figure 56 Episode depicting the costs accumulated from every state visited by the algorithm (Own work)

[15] One thing that was not considered on the costs are the different taxations and government policies that might place a cap on the amount of expenses that the house hold might have to pay. For that reason, the costs of energy bill expenses that can be observed here might not be reflecting realistic amount that the residents will be called to pay.

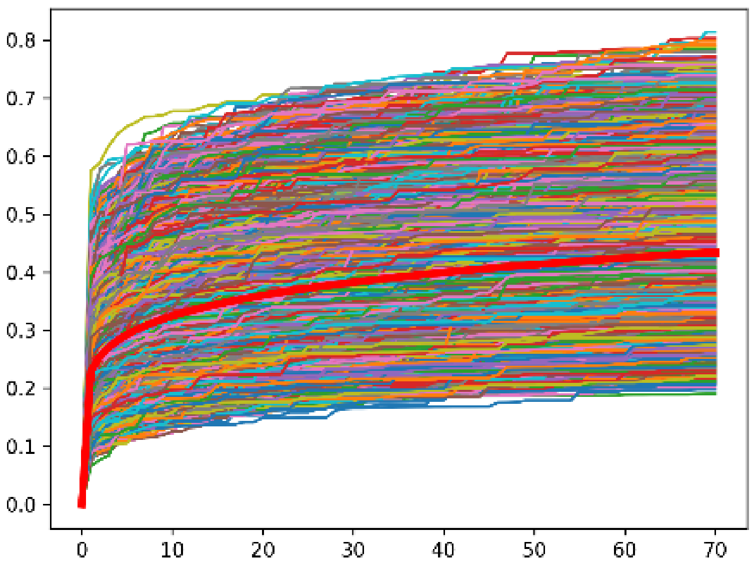


Figure 57 Mean degradation curve and different scenarios of material degradation

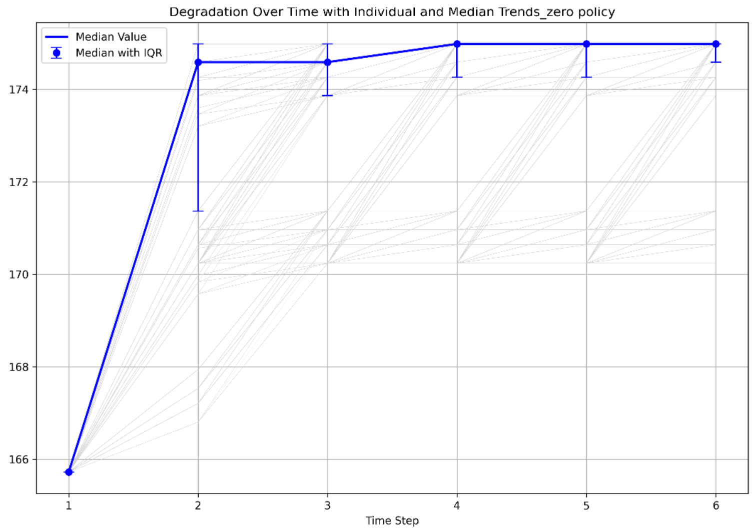


Figure 58 Median energy demand of the building in every time step

The steep rise of the energy demand could be traced back to the steep curvature of EPS insulation's degradation. As seen on the graph of the generated samples of different degradation scenarios of the material, in mean, the material was expected to reach around 30% degradation in the first 10 years of its lifespan and before settling in a very much flatter degradation curve for the rest of the time.

From the below graph, the median rise of energy demand per square meter based on one million episodes can be observed. The building is expected to reach the state of max energy demand at second time step (year 10) and from there a small rise can be expected in year 40.

The distribution of values depicted underneath also points out that in most cases the building is reaching the ultimate state of degradation however because of the distribution of the scenarios, the graph indicates a lower median energy demand, especially in the second time step. The energy demand of the building is matching the degradation probability plot quite closely. This indicated that

1) the environment was behaving correctly, matching the building energy demand with the probabilities of material degradation 2
2) based on the given data, the building would degrade its performance quite fast if only the insulation was to account for the annual energy demand.

3) The expected energy demand was quite small compared to the prices of retrofitting actions.

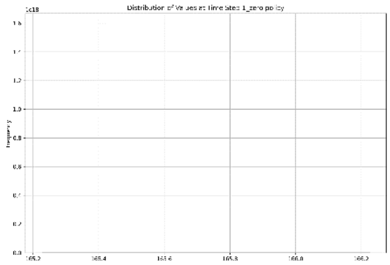


Figure 63 Distribution of energy demand in time step 1

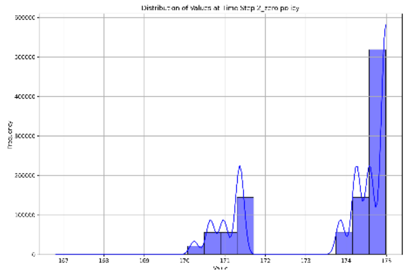


Figure 62 Distribution of energy demand in time step 2

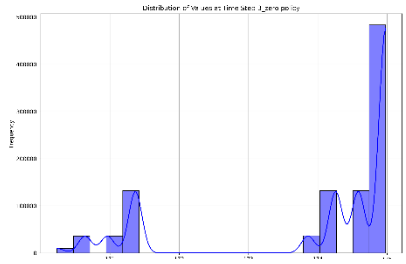


Figure 61 Distribution of energy demand in time step 3

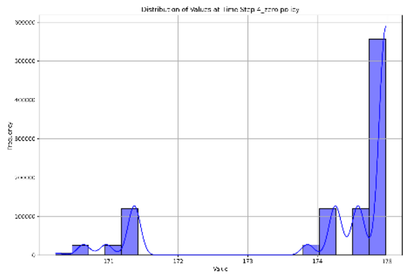


Figure 64Distribution of energy demand in time step 4

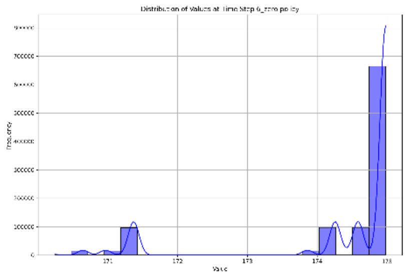


Figure 60 Distribution of energy demand in time step 5

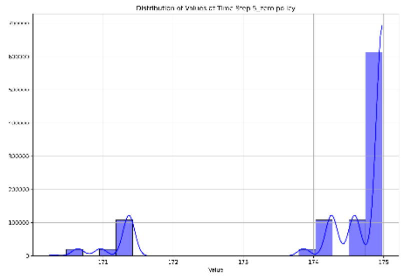


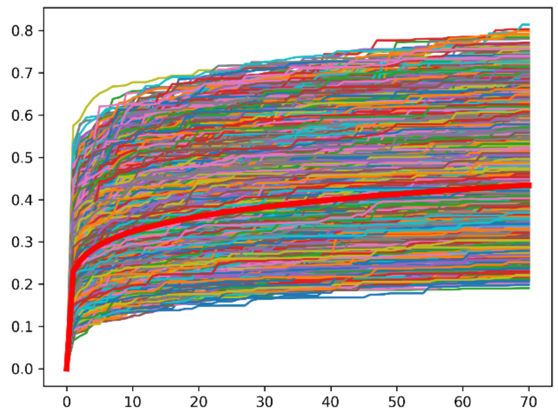
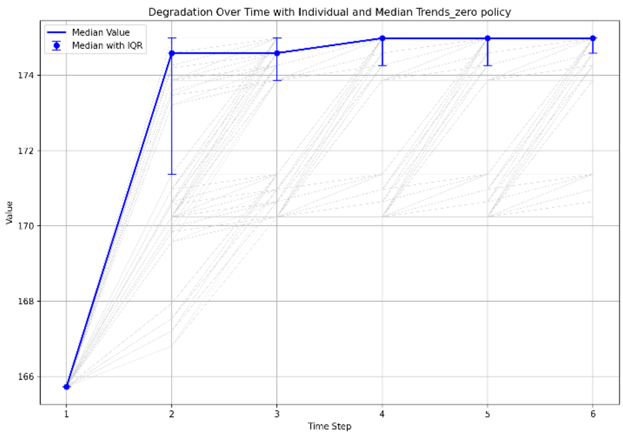
Figure 59 Distribution of energy demand in time step 6

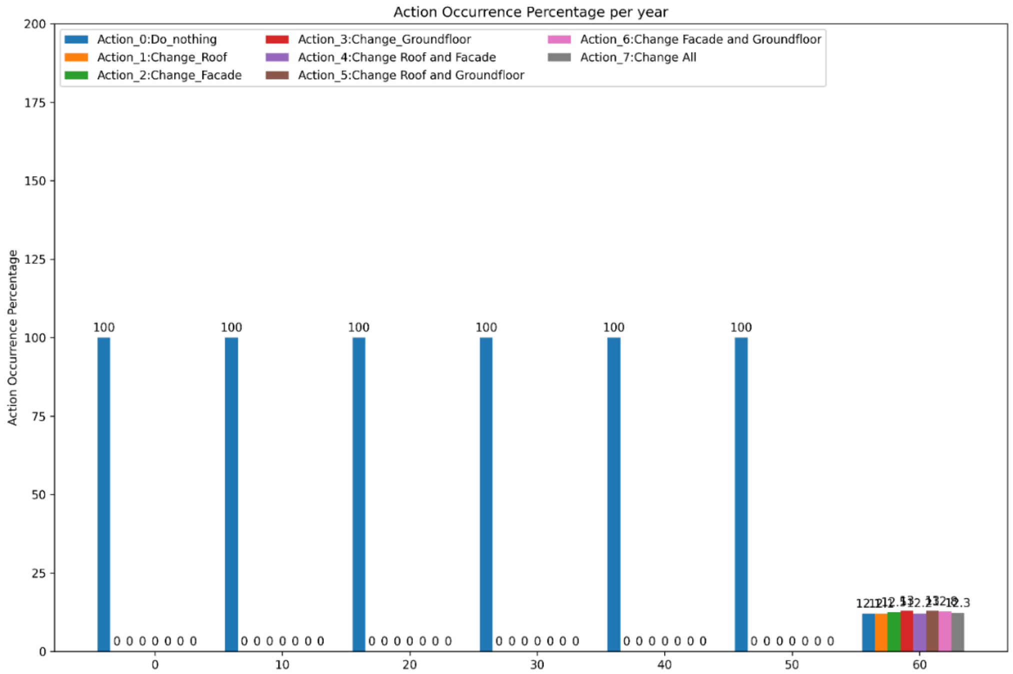
6.2. Test Comparisons

Since the environment seemed to be working correctly and the policy according to the set problem formulation was proving to be same with the benchmark policy, a series of tests were conducted to explore the policies that would be given if the transition probabilities or the rewards were changed. This would allow to not only get a better understanding of the environment and the policies given by the algorithm but also draw conclusions of future changes that should be done in the project.

6.2.1. Comparison 1: Penalty vs No penalty policies

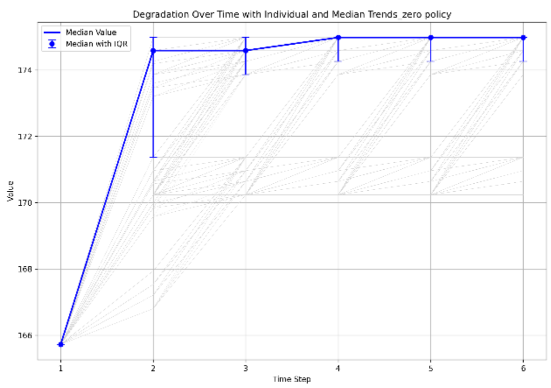
The initial test involved the introduction of a penalty in the case that a certain state of energy demand was reached. This test was based around the idea that different stakeholders are involved in the problem of building degradation. In the beginning of the thesis, the goals of EU trying to better the CO_2 production of the union were explored. In order to reach those goals, EU is introducing policies and incentives requiring the energy demand of the building stock to be recorder and the buildings to be retrofitted to be thermally more efficient by covering a part of the investing costs through different programs. In the case of office buildings, the Dutch government has already set a requirement that the building energy label should be equal or above energy label C. Based on this information, an arbitrary policy was introduced to test the algorithm's results. The policy indicated that the energy bills would double each time that the building would reach energy demand of 174 kWh/m².

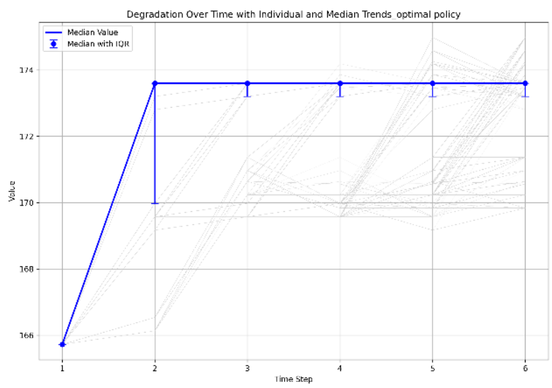
Table 15: Comparison between pernalty and no penalty policies	
No penalty scenario	
	
Material degradation curve, β = 0,15	Median energy demand change over each time step under do nothing policy



Optimal policy when reaching a certain state of energy demand doesn’t result in a penalty

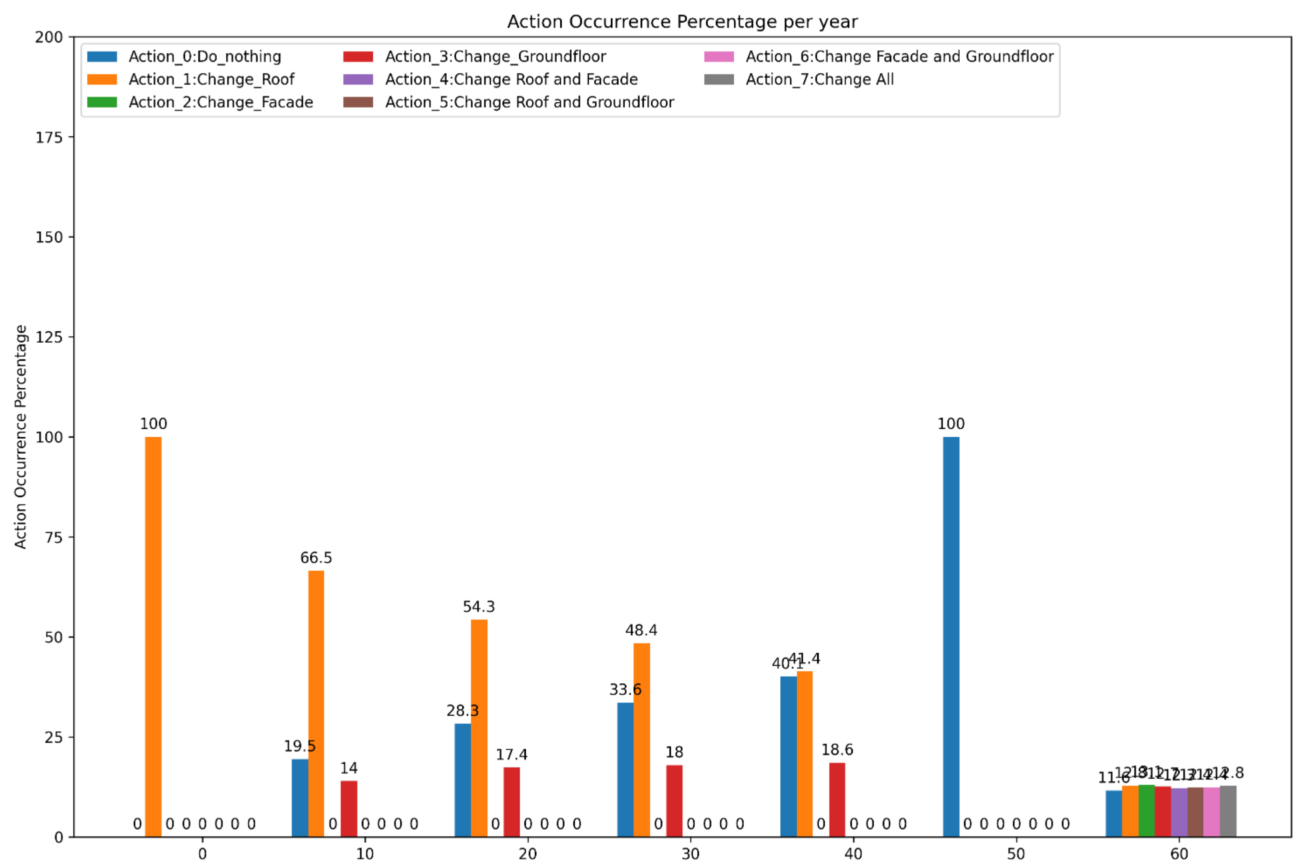
Penalty Scenario





Median energy demand change over each time step under do nothing policy. State of 174 kWh per m^2 are reached in the first 10 years

Median energy demand change over each time step under do nothing policy. The penalty prevents any state equal or bigger that 174 kWh per m2 to be reached



Optimal policy when reaching a certain state of energy demand results in a penalty (doubling of energy bills)

In this case optimal policy indicated changing the roof insulation right from the start of the optimization period as measure. The retrofitting actions were then diminished as the time closer to the end of the simulation period was closing in. This was something that could be expected as nearing the end, there was less of a reason for an action to be taken, since the episodes would finish. The action taken right in the beginning of the analysis period could be explained based on the degradation diagrams of the material degradation and the overall system degradation: since the degradation curve of the materials was so steep, the material was probable to degrade on mean more than 20% in the first 10 years. Based on that, it was expected that the simulation of the building degradation scenarios would reach one of the worst building degradation scenarios in the first-time step after the beginning point.

For that reason, the algorithm chose to take action right from the beginning, to avoid reaching a state of degradation in the next step.

One question that was brought was "why actions were taken since the system is as good as new at time step 0?".

More tests were conducted to see if this behaviour would remain or change in different scenarios.

6.2.2. Comparison 2: Introducing higher energy demand

Trying to understand how different parameters changed the results, the test was rerun, with the transition probabilities having been returned to their original form. This time, the building energy demand simulation was transformed. A simple rule was integrated: change the infiltration rate by one point whenever a degradation percentage was raised by 20%.

```
P.Flow_Rate_per_Exterior_Surface_Area = 0.0007
# Sum the conductivity values
sum_conductivity = roof_conductivity_percentage +
  wall_conductivity_percentage + floor_conductivity_percentage

# Check the conditions in descending order
if sum_conductivity >= 1.2:
    P.Flow_Rate_per_Exterior_Surface_Area = 0.0013
elif sum_conductivity >= 1:
    P.Flow_Rate_per_Exterior_Surface_Area = 0.0012
elif sum_conductivity >= 0.8:
    P.Flow_Rate_per_Exterior_Surface_Area = 0.0011
elif sum_conductivity >= 0.6:
    P.Flow_Rate_per_Exterior_Surface_Area = 0.0010
elif sum_conductivity >= 0.4:
    P.Flow_Rate_per_Exterior_Surface_Area = 0.0009
elif sum_conductivity >= 0.2:
    P.Flow_Rate_per_Exterior_Surface_Area = 0.0008
```

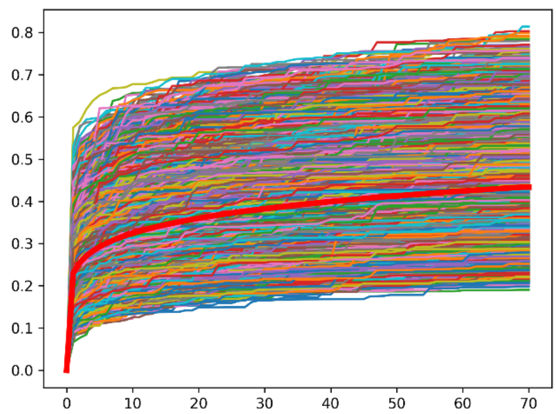
Figure 65 Code snippet depicting the rules of infiltration increase in the code

As the buildings go through cycles of temperature fluctuations and come in contact with different factors that can increase their degradation rate, cracks and other types of infiltration points can appear. These cracks can carry moisture, air and other components in the internal layers of the envelop and affect the performance. In this case, the degradation of the envelop components meant that the insulation performance would drop because of the cracks that would appear in the envelop. With each rise of insulation degradation, the infiltration was assumed to also rise.

The results were a new set of scenarios where the energy demand difference reached 60% from the initial state (initial state 165 kWh/m2, worst case 265 kWh/m2). The table 10 with all the new degradation scenarios can be found in the appendix.

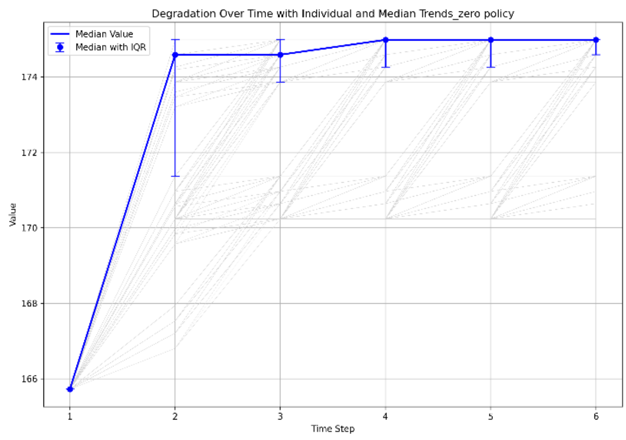
Table 16: Comparison between original degradation and degradation with rise of infiltration

Original simulation with 5.6% change in energy demand from best to worst case

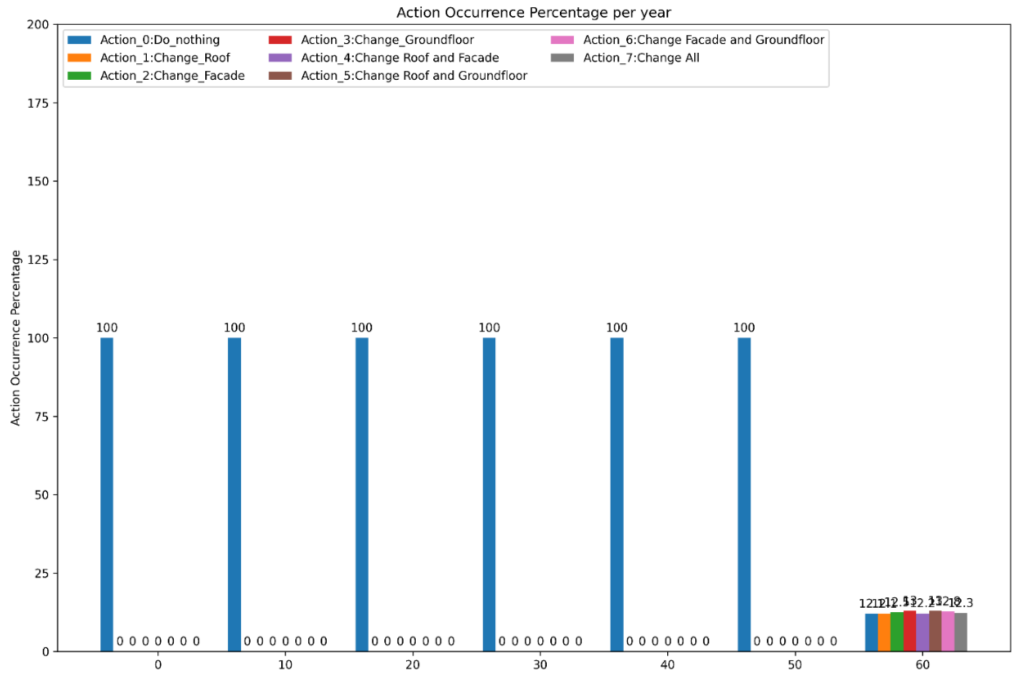


Material degradation curve,

β = 0,15

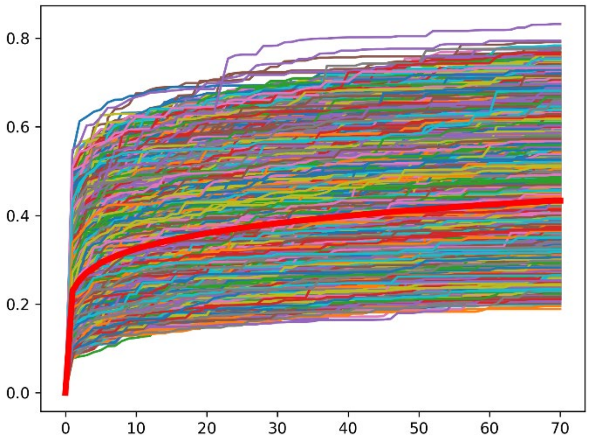


Median energy demand change over each time step under do nothing policy



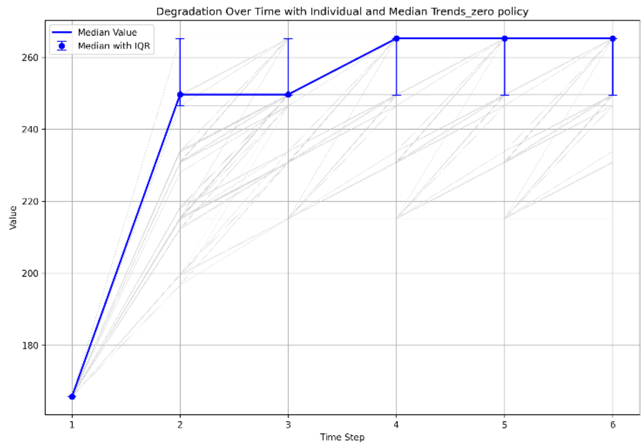
Optimal policy given with 5.6% change in the energy demand between initial and worst case scenario

New simulation with 60% change in energy demand from best to worst case

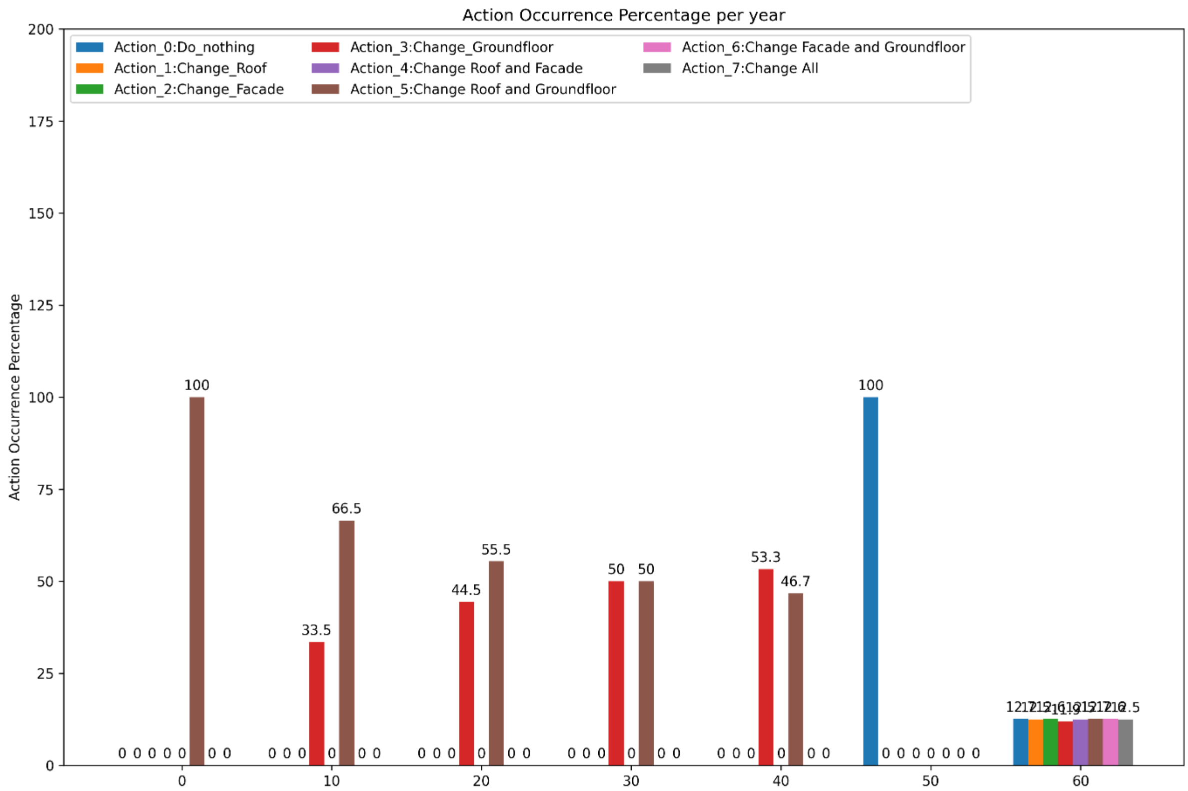


Material degradation curve,

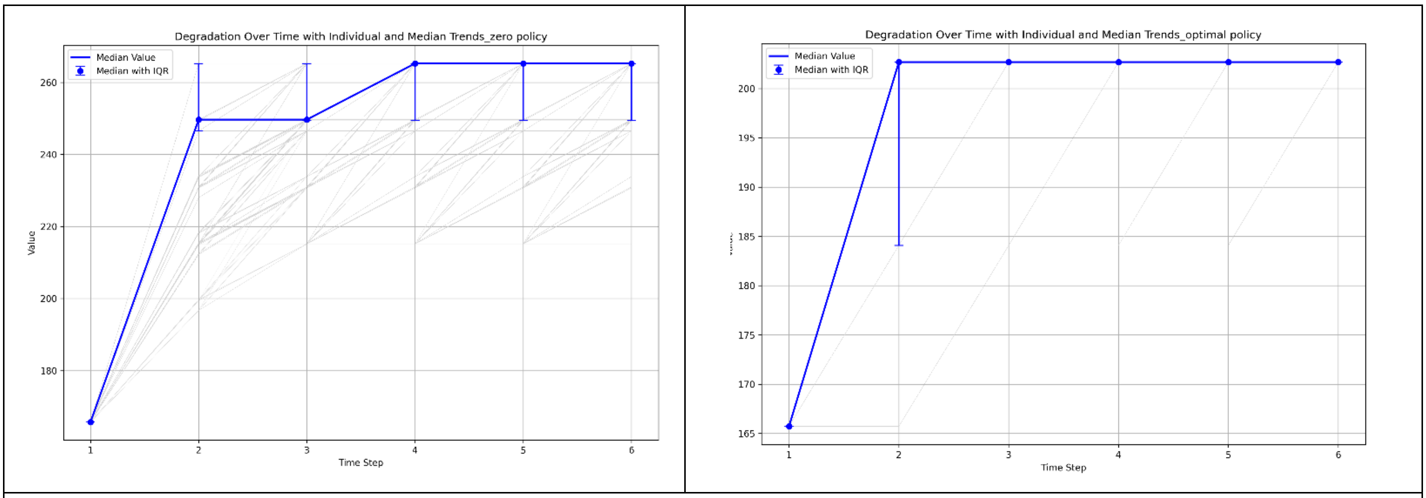
β = 0,15



Median energy demand change over each time step under do nothing policy



Optimal policy given with 5.6% change in the energy demand between initial and worst case scenario



Comparison between median energy demand change over each time step under do nothing policy (left) and optimal policy (right). In the do nothing policy, the energy demand will rise to 265kWh/m^2 while in the optimal policy , the algorithm will try to keep the buidlign demand from raising above 210 kWh/m^2

The results from the second test showed a new policy being introduced. The algorithm tried to keep the building from reaching above 210 kWh/m^2. This probably meant that at that point , the energy bills for the next 10 years would overpass the costs of taking an action. The action change roof and ground floor insulation appeared most prominently throughout the optimization period. The reason of that happening could be reflected to three factors:

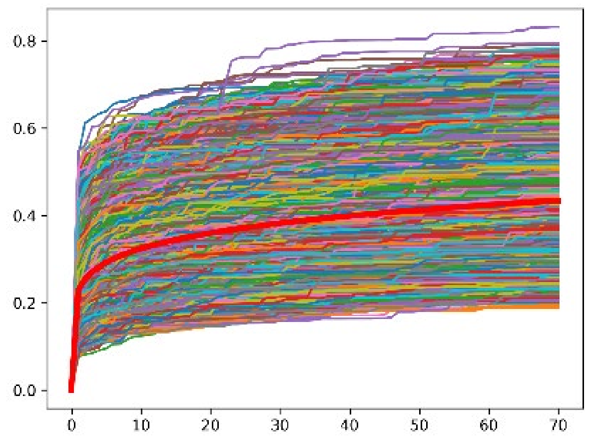
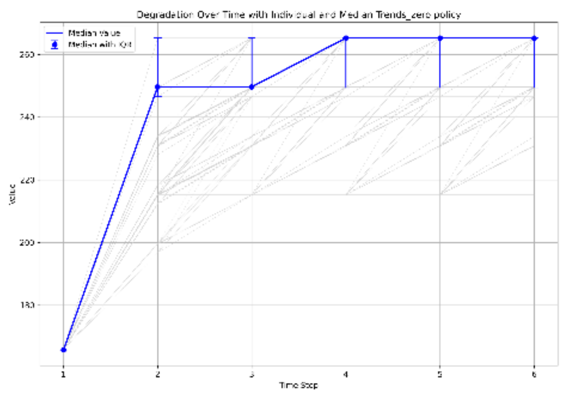
1) the costs of changing any of those two components was quite below the costs of changing the exterior wall insulation. The changing of roof insulation amounted to 13.407 euros and the changing floor insulation to 2.614 euros. The combination of the two actions would still be quite cheaper (by 27.512 euros) than changing the wall insulation.
2) The degradation model in this case changed the infiltration rate when the degradation of any component was raised by 20%. That prompted the algorithm to try to change as many components as possible in order to keep the infiltration in lower levels. In real life, the raise of the infiltration rate might correlate to the total surface area of the component. In that case, the energy demand might be higher when the wall component (assuming is the biggest surface area) is degrading.
3) Even though a policy is given , it appears quite deterministic with only two actions allowed to be taken, with great frequency from the beginning and almost to the end of the buidling's lifespan. This is assumed to be connected to the steep degradation that can be expected. Basically , the steepness enforced an action to be taken in every time step in order for the building to not reach a state of close to total degradation in the next time step.

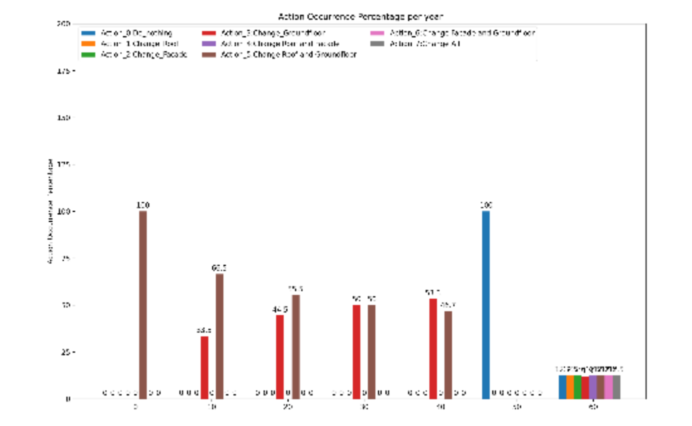
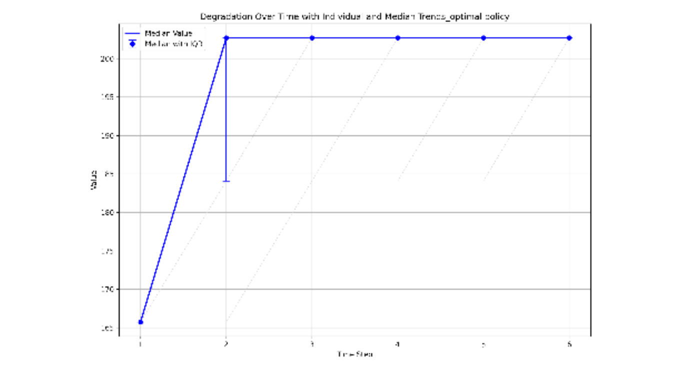
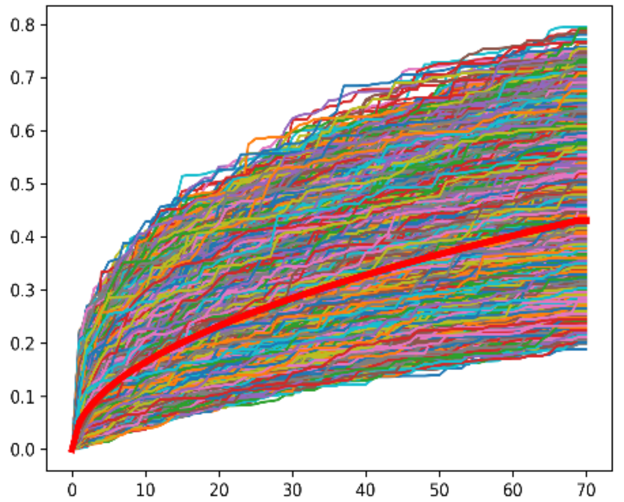
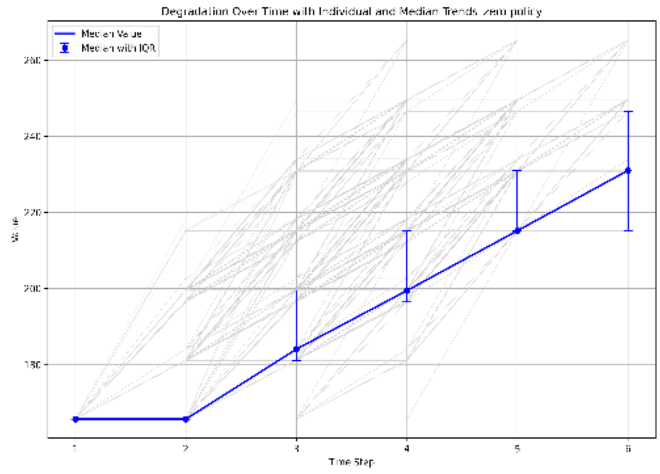
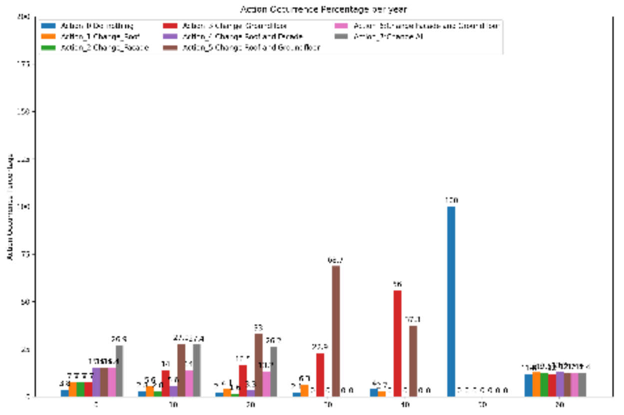
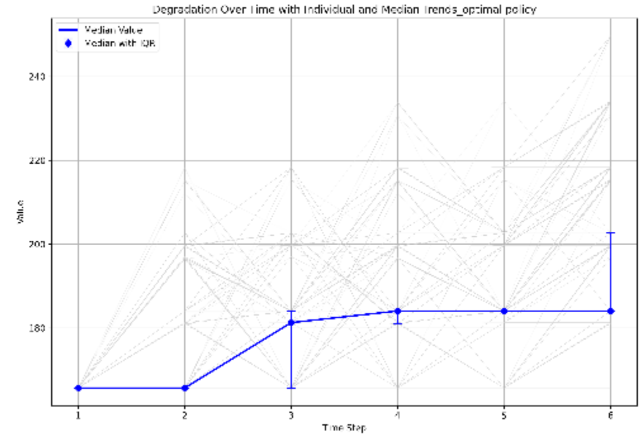
6.2.3. Comparison 3: Different transition probabilities

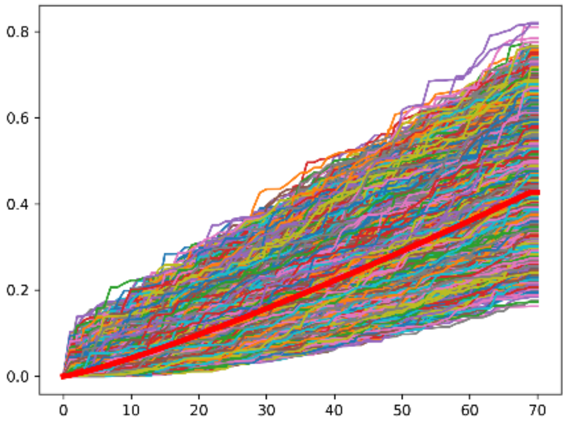
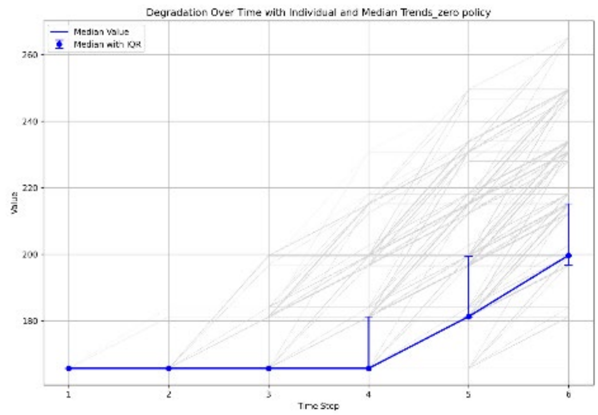
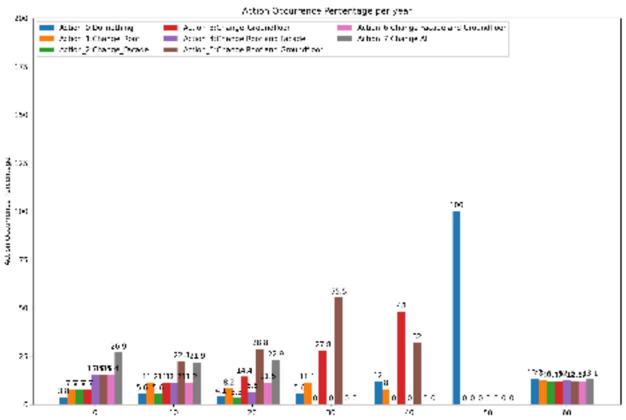
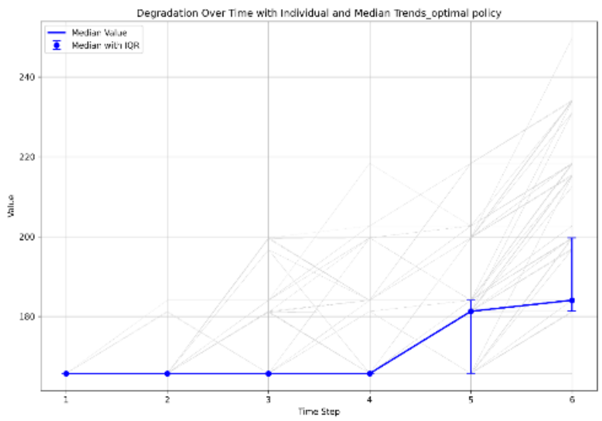
Based on the results of the previous tests, another set of tests was conducted to see the influence of transition probabilities to the policies. Even though both the initial and the higher energy demand scenarios were simulated, only the later where chosen to be depicted here as they present bigger interest. Plots of all the simulations can be found in the appendix.

The change in the transition probabilities of the material degradation brought also a change in the optimal policy. When the β (beta) value that described the curvature of the degradation was changed from the initial 0.15 to 0.5, making the degradation less steep, the policy proved to be more “rich” in actions than before. Even though the actions “change roof and ground floor” and “change ground floor” still prevailed especially in years 30 and 40 they were becoming less apparent in the initial years.

What is interesting to see is that the policy, with small number differences is quite similar in both cases with beta 0.5 and 1.2. This could give some pointers that the median policy could be expected in cases were a more stable model of degradation was introduced.

Table 17 : Comparison between different transition probabilities	
Case 1	
	
Material degradation curve, β = 0,15	Median energy demand change over each time step under do nothing policy

	
Generated optimal policy	Median energy demand change over each time step under optimal policy
Case 2	
	
Material degradation curve, β = 0,5	Median energy demand change over each time step under do nothing policy
	

Case 3	
Generated optimal policy	Median energy demand change over each time step under optimal policy
	
Material degradation curve, β = 1,2	Median energy demand change over each time step under do nothing policy
	
Generated optimal policy	Median energy demand change over each time step under optimal policy

7. CONCLUSIONS

In this chapter the overall conclusions derived from the methodology and results of this study are presented and commented. The conclusions are divided into two parts: the first part covers the general methodology, its bottlenecks, and insights for future researchers; the second part delves into the conclusions from the value iteration trials, addressing specific issues and proposing future researches.

6.3. Methodology and Literature review

The project progressed in two main phases. Initially, an extensive literature review was conducted to establish a theoretical foundation. This review covered the theory of heat balances to understand the energy behavior of buildings due to changes in the envelope and the behavior of insulative materials. It also explored different methodologies for simulating building degradation over time. Additionally, different MDP methods were analysed, particularly focusing on Dynamic Programming's Value Iteration and RL's Q-learning. Even though thorough, the literature review didn't cover all the important aspects. The case studies (Ferreira et al., 2023) and (Mavromatidis & Petkov, 2021) that worked also with different approaches on building stock maintenance were not depicted in the case studies of the literature review. Even more, the exploration of the reinforcement learning methods remained in analysing only two methods of optimization and didn't explore more options.

Based on the literature, a basic methodology was developed. Key references included the works of (Maia et al., 2021, 2023) and (Ferreira et al., 2023) which provided a framework for retrofitting actions and objective function theory. Despite limited access to complete data and documentation, these works guided the problem's construction. The development of the optimization technique to represent the problem as a Markov Decision Process, and the basic theory of the Reinforcement Learning principles relied heavily on (Sutton & Barto, 2018) and online tutorials .

A significant challenge was the lack of actual data regarding building degradation, necessitating considerable time to gather or generate necessary information. Research on the thermal resistance and conductivity changes in insulation materials provided insights into one mechanism affecting building behavior, though future studies should also consider moisture, heat, and airflow, especially given climate change impacts. This study focused on plastic foam insulation, but future research should compare various insulation types.

6.4. Literature Review Conclusions

The Energy Performance of Buildings Directive (EPBD) is an important framework established by the European Union (EU) to reduce energy consumption in buildings and promote environmental sustainability. Buildings account for 40% of energy usage in the EU, and the EPBD aims to ensure that by 2050, all buildings in Europe will use significantly less energy. To achieve this, the EPBD collaborates with other directives, such as the Energy Efficiency Directive, to enhance energy efficiency across the continent.

Despite the potential for significant energy savings, many buildings in the EU were constructed before 2000 and exhibit poor energy performance. Renovation rates remain low, which hampers progress towards energy efficiency goals. To address this, the EPBD emphasizes the need to increase renovation rates, particularly for buildings with poor energy performance.

By proactively addressing potential issues and implementing modern technologies, retrofitting ensures that buildings remain functional, efficient, and compliant with current regulations, thus maintaining their overall integrity and value.

Maintenance policies can be broadly categorized into planned and unplanned types. Planned maintenance involves scheduled activities to prevent failures and improve performance, while unplanned maintenance is reactive, addressing issues only when they occur, often leading to prolonged downtimes and user dissatisfaction. Planned maintenance can be further divided into preventive, corrective, and improvement strategies, with preventive maintenance focusing on addressing potential issues before they cause failures, thereby enhancing the reliability and lifespan of building components (Ferreira et al., 2023).

Preventive maintenance includes several sub-strategies: predetermined, condition-based, opportunistic, and predictive maintenance. Predetermined maintenance schedules tasks at regular intervals regardless of the condition of components. Condition-based maintenance relies on regular inspections and specific criteria to decide when maintenance should be performed. Opportunistic maintenance takes advantage of planned downtimes to conduct additional tasks, minimizing disruptions. Predictive maintenance is the most advanced form, using data analysis to predict failures and schedule maintenance accordingly. This approach is especially relevant for retrofitting buildings. Unlike traditional corrective maintenance, which reacts to failures, predictive maintenance anticipates potential issues and addresses them proactively, reducing downtime and overall maintenance costs while extending the operational life of building components (Ferreira et al., 2023).

Concerning the planning of retrofitting, existing methodological frameworks such as EnerPHit(Jan Steiger & Eva Vahalova, 2019), iBroad(*IBROAD - European Commission*, n.d.), and ALDREN (Sesana et al., 2020) offer general roadmaps for building retrofitting and fall under the planned and condition based types.

However, these types of roadamps are not trying to find an optimal plan but offer generic methodologies of actions that should be taken to improve building performance. Even thought there are a lot optimization tools involving genetic algorithms that provide the optimal retrofitting packages, only a few were detected that involved time and planning optimization of the maintenance and improvement actions. Optimization models have been proposed to determine the best timing and sequence of retrofitting steps, considering budget constraints and the interdependencies between measures, while other approaches tried to consider the physical degradation of the materials in order to determine the optimal timing for maintenance actions(Ferreira et al., 2023).

Based on the literature research on building performance, heat losses through the envelop can greatly affect the energy performance of the building. There are different components that might affect the building energy demand as well as different factors that might contribute to how fast this loss of performance may occur. Among them, the ageing factor can contribute to the loss of themal performance. Even though this has become apparent, there seems to be a lack of (especially of field) data regarding how much the U-values of the building might change over time.

However, based on the above findings it has become apparent that uncertainties play a crucial role in the formulation of the problem- something that should be taken into account for any future works.

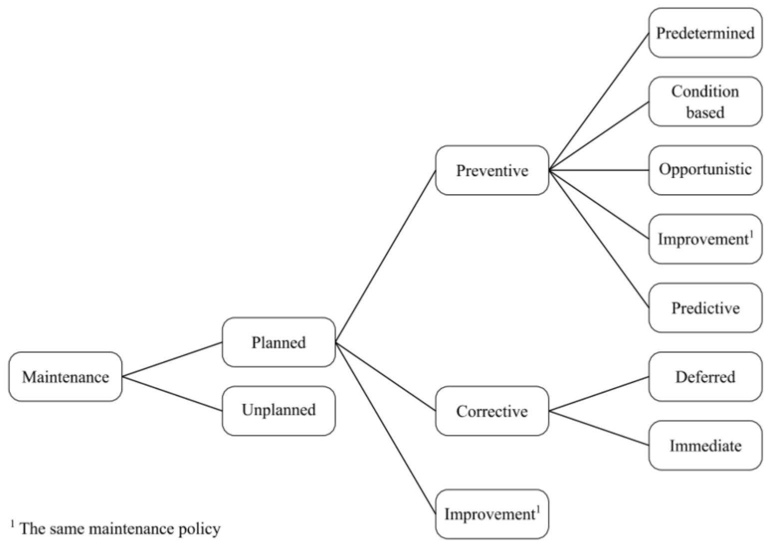


Figure 66 Taxonomy of different maintenance policies (Ferreira et al., 2023)

To address these kinds of problems , algorithms able to work with predictive maintenance should be used in order to reduce unexpected failures and associated costs. By leveraging predictive maintenance, building managers can enhance the performance and longevity of their assets, making it a key strategy in modern building management and retrofitting efforts.

Planning algorithms that fall under a broader category of algorithms known as "Sequential Decision Making" algorithms, which handle problems involving a sequence of actions(Littman, 1996) are emerging as probable ways to work with predictive maintenance problems. Reinforcement Learning (RL), a data-driven optimization algorithm, has gained popularity for its ability to develop effective maintenance policies and address predictive maintenance challenges (Ogunfowora & Najjaran, 2023).

Several conclusions became apparent throughout the literature review:

1. Limited Research on Time-Factor Building Maintenance and Renovation Optimizations:

There is a notable gap in research focused on optimizing building maintenance and renovation plans that consider the time factor. Most existing studies offer optimizations based on genetic algorithms in order to offer retrofitting packages. Throughout the literature review, only a very few papers were discovered that took into account the time factor in their optimization methodologies. The absence of comprehensive time-based optimization models means that opportunities to extend the lifespan of building components and achieve cost savings through strategic, long-term planning are often missed. Future research should prioritize developing methodologies that incorporate time as a critical variable in maintenance planning to enhance both efficiency and effectiveness.

2. Lack of Consistent Methodology in Existing Bibliographies:

Current bibliographies reveal a fragmented approach to building maintenance and retrofitting optimization. There is no consistent methodology that comprehensively addresses all major factors necessary for developing robust models. This inconsistency hampers the ability to compare results across different studies or to build on previous work effectively. To advance the field, a unified framework that integrates economic, technical, and environmental considerations is needed. Such a framework should include standardized metrics and methodologies for assessing building performance, predicting maintenance needs, and evaluating the long-term impacts of various retrofitting strategies in both physical and function level on the building.

3. Neglect of Correlation Between Physical and Thermal Degradation:

No existing research has adequately explored the correlation between the physical degradation of building components and their thermal degradation. This is a critical oversight, as physical wear and tear can significantly affect a building's thermal performance and energy efficiency. Ignoring this correlation may lead to suboptimal maintenance and retrofitting decisions. Incorporating an understanding of how physical deterioration impacts thermal properties into maintenance algorithms could lead to more accurate predictions and better-informed policy recommendations. This integrated approach would help in developing maintenance strategies that more effectively balance cost, performance, and longevity.

4. Underutilization of Reinforcement Learning in Retrofitting Planning:

Despite the known potential of Reinforcement Learning (RL) algorithms to solve complex planning problems, there has been no significant application of these algorithms in the context of retrofitting planning. RL's ability to learn and adapt from interactions makes it particularly suitable for dynamic and uncertain environments like building maintenance and retrofitting. Its absence in current methodologies represents a missed opportunity to harness advanced computational techniques for optimizing retrofitting schedules and strategies. Future research should explore the integration of RL into retrofitting planning to take advantage of its capabilities in handling sequential decision-making and uncertainty, potentially leading to more efficient and effective retrofitting processes.

By addressing these gaps, future research can significantly improve the strategies for maintaining and retrofitting buildings, ultimately contributing to more sustainable and energy-efficient building management practices.

6.5. Development and Implementation:

The second phase involved iteratively realizing the methodology workflow, transitioning from experimental approaches to more concrete solutions. Various methodologies were employed to simulate building energy demand, progressively incorporating non-stationary transition probabilities and additional actions into the environment. Value iteration was chosen as the solver method, to test the environment's dynamics. Different techniques were tried to expedite the convergence process, like utilizing NumPy libraries to pre-calculate states, actions, and rewards. However, the large state space rendered unable to test in detail a 60-year planning, as it required enormous computational power.

Sensitivity analysis tested different scenarios, providing conclusions on the performance of value iteration and decision-making results, which identified optimal retrofitting plans over a 60-year building lifespan for two different scenarios of rewards, concluding the fluent behavior of the environment.

6.6. Methodology conclusions and discussion

6.6.1.1. Energy Demand Simulation:

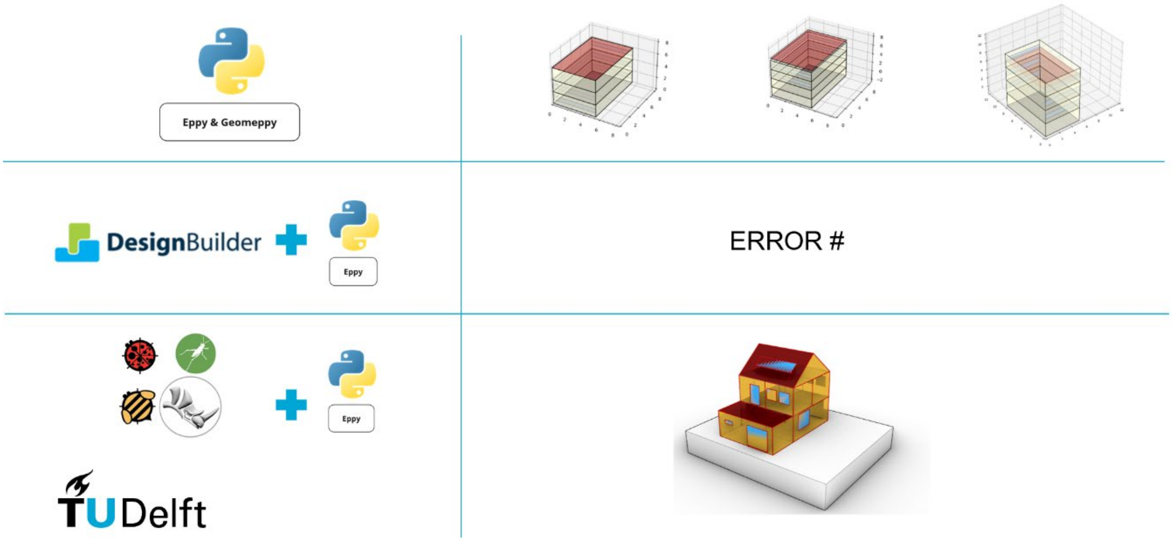


Figure 67 Comparison of results of different methodologies of building modelling and simulation

Three methodologies were employed to simulate the building's energy demand:

Geomeppy: This method allowed for embedding within a Python file, enabling iterative simulations across different building degradation scenarios. Although each simulation ran in 3-5 seconds, overall results from the TBL files took approximately 400 seconds. The basic geometry produced by Geomeppy was prone to bugs, and limited community support hampered problem-solving. If the community grows, Geomeppy could facilitate faster simulations. However, users must have substantial experience with EnergyPlus and handling IDF files, as this process is time-consuming.

Design Builder: Design builder modelling software is broadly used to model and simulate the energy performance of buildings. Even though it is quite known and broadly used, it lacks a user friendly interface. Even more , the software was susceptible to bugs and errors which hindered the generation of multiple scenarios iteratively. Even more, the idf files exported by the software appeared corrupted and unable to be further edited by the Eppy library commands.

Grasshopper: This tool provided more robust support for developing a building model, but each basic simulation took around 1 minute. Running numerous simulations could significantly extend the overall time required. Grasshopper's scripting components often slow down due to data piping complexities, though they simplify scripting for casual users.

One thing that must be noted though was the amount of time needed to set up the building model. While the Geomeppy geometry required more than two weeks to get familiar with the commands and set up a simple geometry, creating and manipulating the building data in this case amounted to 3 days, with a limited already knowledge of the grasshopper interface and the ladybug plug in commands.

Combining both tools, generating a basic IDF file in Grasshopper and simulating scenarios using the Eppy library , allowed for more accurate building modeling results in a total of 400 seconds for multiple files. On average, simulating and storing 27 different degradation scenarios took about 14 seconds per simulation. Large-scale simulations can slow down the value iteration or RL method if scenarios are simulated in every episode. Therefore, pre-simulating all energy performance scenarios is recommended for efficiency.

Even more the current simulation still lacked more realistic interpretation of the building energy degradation. Based on the methodologies of (Taki & Zakharanka, 2023)(Eleftheriadis & Hamdy, 2018) airtightness should also be considered in the building degradation together with fenestration gas loss and HVAC system's loss of Coefficience of Performance in order to simulate more accurately the building's behaviour through time.

6.6.1.2. Environment and MDP problem formulation

The environment for this study was based on the Markov Decision Process (MDP) framework provided by Prateek Bhustali. However, it is not compatible with OpenAI's Gym framework, which is commonly used for Reinforcement Learning (RL) experiments. Calculating transition probabilities was complex process to code and computationally time-consuming, taking up to 45 minutes to compute and store all state transition matrices for 8 actions. Future research should consider developing a more user-friendly environment that is compatible with Gym documentation. This would facilitate the use of RL methods and potentially streamline the process.

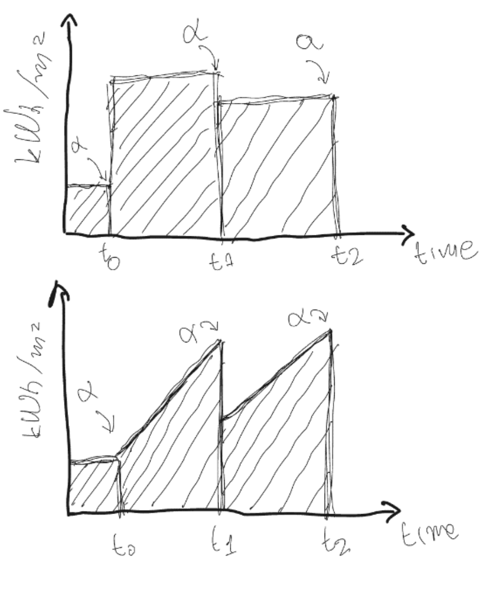


Figure 68 Graph depicting the energy demand in every time step (above) and how it should become (below) in future interpretations for more correct rewards calculations

The tests results also provided some insights and errors with the problem formulation. The way that the problem was formulated allowed for an action to be taken and the outcome of that action would amount to the negative rewards (energy bills) that would be expected in the next 10 years. However , no degradation was expected to happen on the mean time. A future model of the problem should reformulate the problem and take those elements in account for the transition probability matrix and the rewards.

States

The thesis formulated the states as the degradation percentages of three major envelop components and incorporated the ages and time in the state space. A more complete environment would incorporate also the window and HVAC degradation states and possibly their ages , the budget restrictions of the owner , the climate scenarios. Since the components might not actually have the same materials , or different factors might affect their performance, they should be treated separately and different degradation curves should be exploited for each case.

Even more, since there was no clear correlation between the material physical degradation and the energy demand, a maybe more correct state space should be created. The new state space would reflect the different stages of a component's physical degradation (in classes that indicate when it is close to a point that it has to be changed) , and the possible states of energy demand.

New state space → *[component physical degradation]x [expected energy demand]x[budget]x[climate]*

Actions

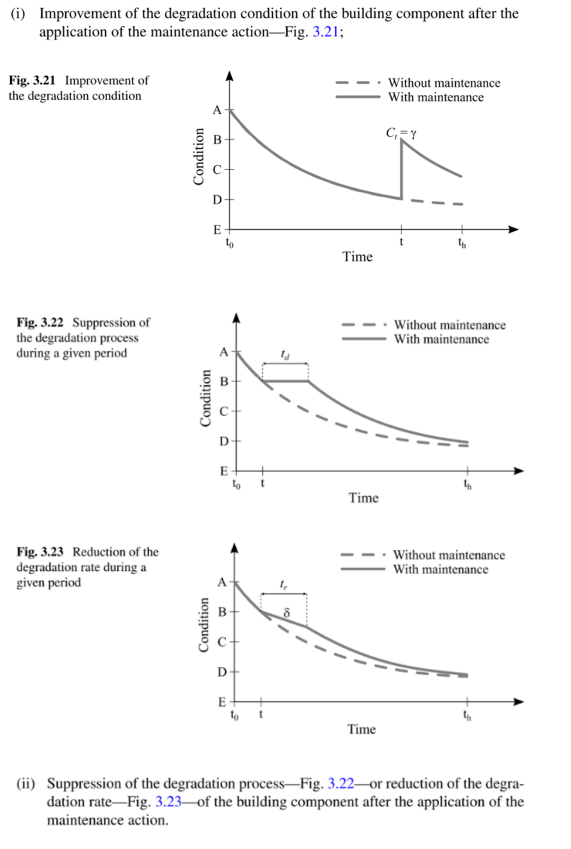


Figure 69 Graph depicting how actions could affect the degradation process in a more realistic model (Ferreira et al., 2023)

This study focused on eight binary actions. Typically, predictive maintenance problems include actions such as 'Inspection', 'Minor Maintenance', 'Major Maintenance', and 'Change Component'. However, this research only examined the 'Change Component' action. Future studies should include these additional actions and define what 'Major Maintenance' entails in the context of retrofitting. Understanding how major maintenance translates to changes in the building environment is crucial. Ferreira et al., (2023)offer some suggestions on addressing these issues.

A more comprehensive investigation should explore how different combinations of measures impact long-term policies. This thesis provided preliminary measures based on recommendations from (*Kostenkentallen | RVO*, n.d.), but a more detailed approach is necessary.

Moreover, this study assumed perfect installation of retrofitting measures, leading to specific changes in building performance. Future research should investigate how building performance evolves over time with different retrofitting techniques.

Rewards

The reward structure in this study was based on the objective function of Maia et al., (2021) . Maia et al., (2023) and adjusted to work with the problem formulation, however, suggests incorporating additional objective functions such as CO2 emissions and global costs, which should be examined in future research. If taxation of CO2 emissions significantly increase, it might affect the energy bills, more actions may be recommended by the policy. Conversely, newer and more natural materials, which have lower lifecycle CO2 emissions, might be preferable.

According to the above-mentioned papers, retrofitting building components in the correct order can lead to significant cost savings by avoiding oversized heating systems for extended periods. This study did not fully explore the 'lock-in effect' due to insufficient data, but this remains an important area for future research. Assuming there is an optimal sequence for changing building components, simulations should be redone with adjusted rewards to encourage the correct order of actions. Maia et al. noted that changes in buildings are often circular, with boilers being replaced more frequently than other components. Future studies should address this by adjusting simulations to reward correct sequences of retrofitting actions. In reinforcement learning, this process can be modeled with periodic checks, recommending changes at intervals (e.g., every 10-20 years), and positive rewards for the correct sequence of actions.

6.6.1.3. VALUE ITERATION

Value Iteration was chosen as the optimization algorithm. Being a model based method with no hyperparameters, like Reinforcement Learning methods, allowed for a closer look at the dynamics of the environment and the produced policies. In that sense it is believed that it was a good starting point to understand the complexities of the problem and draw conclusions for future works to take into account.

However, as the model will become more realistic with additional data (e.g., degradation of boilers and windows), the state and action spaces expand significantly. This increase in complexity causes both Dynamic Programming and basic Q-learning to struggle in finding solutions. One technique that was considered in order to optimize the process is state aggregation based on (Duan et al., n.d.) . State aggregation is a technique that clusters states with similar transitions and rewards and treats them as a single state in value iteration. Based on the fact that a lot of states are similar, if the time factor is excluded, this technique would help mitigate the problem. However, even then, it can be argued that the more realistic the problem becomes the more the state space will expand until it will reach a point that DP and shallow learning will not be able to handle it. Therefore, advanced deep learning techniques should be considered to handle the increasing complexity of the state and action spaces in this study. Deep Reinforcement Learning utilizes neural networks to search through high dimensional spaces. However as stated in (Bhustali, 2023) : "Single Agent Deep Reinforcement Learning approaches do not scale well under practical computational constraints because the joint space of states, observations and action spaces grow exponentially with the number of agents". In their paper, they propose to work on Inspection and maintenance problems by using multiple agents. Each agent, they propose would have the task to handle one component of a system. Based on this paper, the recommendation would be for the problem to be reformulated and solved again using MADRL methods.

6.7. Test results

The tests conducted also indicated that the environment responded accurately to the dynamics of material degradation and energy demand. The correlation between the steep initial degradation of EPS insulation and the corresponding rapid increase in energy demand validated the simulation's alignment with the data taken from the real-world. Additionally, the algorithm's adjustments to the optimal policy in response to varying economic factors, penalties, and transition probabilities demonstrate its robustness and sensitivity. This responsiveness confirms that the environment is effectively modeling the complex interactions between material degradation and energy efficiency, providing reliable insights for optimizing retrofitting strategies.

Overall, the tests demonstrated that the optimal retrofitting policy is highly sensitive to various factors, particularly economic considerations, the imposition of penalties, and the specific model of material degradation employed. Initially, the value iteration algorithm identified a do-nothing policy as optimal in scenarios without penalties, indicating that the retrofitting costs far exceeded the benefits derived from improved energy efficiency. However, this scenario changed dramatically with the introduction of penalties. When penalties were imposed for high energy demand states, the optimal policy shifted towards proactive retrofitting measures. This change underscores the significant impact that economic incentives and penalties can have on decision-making processes related to building maintenance and energy efficiency.

Moreover, the tests underscored the necessity of devising cost-effective retrofitting strategies. When higher energy demand scenarios were introduced, the optimal policy involved frequent and targeted actions, such as changing roof and ground floor insulation. These components were selected due to their lower retrofitting costs compared to other parts of the building, like wall insulation. This approach allowed for managing the overall energy demand more efficiently while keeping costs under control. It also illustrated the importance of prioritizing retrofitting actions based on their cost-effectiveness and impact on overall energy efficiency. One thing that has to be noted in that the simulation of the model's performance degradation didn't follow any indicated data. That meant that there was no correlation between which component was degrading and the expected infiltration.

If the simulation was redone with more detail and observations from real world data, the policies that might be provided will be different from those in this thesis. One expectation will be that components with greater influence on the energy demand will be chosen above others to be retrofitted.

In conclusion, the tests highlighted that optimal retrofitting policies are highly context-dependent, requiring careful consideration of economic factors, potential penalties, and the characteristics of material degradation. Introducing penalties for high energy demand can shift policies from inaction to proactive retrofitting, emphasizing the role of economic incentives in driving maintenance decisions. Additionally, early interventions are crucial in cases of steep material degradation to avoid higher future costs. Finally, adopting cost-effective retrofitting strategies, focusing on components with the highest impact and lowest cost, is essential for managing energy demand efficiently and sustainably. These insights are vital for policymakers, building managers, and homeowners aiming to balance cost, efficiency, and sustainability in building maintenance and energy management.

6.8. Future works Proposals

Several important aspects were not addressed in this research but should be considered for future investigations to provide a more comprehensive understanding of the retrofitting process. These aspects include:

1. Reformulation of the problem and the environment to be solved with Deep Learning, probably mutli-agents approaches.
2. More detailed Building Degradation simulation by incorporating more components of the building, such as windows, airtightness, and HVAC systems, should be considered in the degradation scenarios. Even more , future studies should investigate the potential degradation of PV panels and other sources of energy generation and how those affect the energy demand at the end of the process.
3. More detailed problem formulation with bigger state spaces and more actions. In the new problem formulation other than only the components, budget restrictions of different income groups, climate scenarios and tenant behaviour should also be incorporated in the state space. Even more, the actions for each component should be expanded including minor repairs, retrofitting and renovation packages with more options of measures. The rewards should equally be adjusted to count for life cycle costs, penalties if the owner's budget reaches a certain limit of expenditure and/or if the building drops underneath a performance requirements. Even more, incorporating POMDPs (Partially Observable MDP's [16]) could enhance the

[16] In POMDP's the real state of the environment might not be known. In that case we are working with the probability of the state being in certain condition.

modeling of uncertainties in the retrofitting process, such as incomplete information about the state of the building or the exact impact of interventions.

By addressing these areas, future research can provide a more detailed and practical framework for optimizing retrofitting strategies, ultimately leading to more sustainable and cost-effective building practices.

Another area of interest is for the models to be used for reverse engineering purposes. Reverse engineering can be instrumental in optimizing building retrofitting strategies by dissecting the key factors influencing energy demand and retrofitting costs, such as material degradation rates and economic penalties. By understanding these influences, predictive models can be developed to accurately forecast future states of a building, enabling precise planning of retrofitting actions. This approach can also optimize economic incentives, guiding policymakers to design effective programs that encourage timely retrofitting while balancing costs. Furthermore, reverse engineering can improve material selection and maintenance practices by identifying materials that offer better long-term performance, thus reducing the frequency and cost of retrofitting. Tailoring retrofitting schedules to align with material degradation patterns ensures timely interventions, preventing significant energy losses and cost increases. Additionally, the continuous feedback loop established through reverse engineering can enhance algorithm accuracy and policy robustness, leading to more sustainable and efficient building maintenance strategies.

Last but not least, if a more detailed model is realised, it can be possible to be used to aid in the formulation of more focused European policies, tailored to the cases on each country, building and owner target group. These policies then , is believed, that will greatly affect the rate and efficiency of the retrofitting of the building stock.

Initial Questions and Responses

6.9. Main Question

"How can we optimize staged retrofitting planning using Reinforcement Learning?"

To optimize staged retrofitting planning using Reinforcement Learning (RL), we need to develop a comprehensive RL-based framework that integrates building energy performance simulations, economic analysis, and policy constraints. The framework should involve the following steps:

i. Problem Formulation: Define the retrofitting problem as a Markov Decision Process (MDP) to systematically evaluate different retrofitting actions over time. The formulation of the environment will define the overall process that will be followed to find the results. In that sense it is quite important to understand the mechanics of the various Reinforcement learning methods before hand. This will allow more correct interpretation of the environment.

ii. Simulation Environment: In the absence of data on retrofitting actions for certain building typologies, create a detailed simulation environment that models the building's energy demand. A sensitivity analysis should be conducted to determine the most important factors that affect the particular building's energy performance. Those factors must be take into account. For typical single-family houses, these factors usually include windows, roof, façade, ground floor, and HVAC systems, however each building typology might bring forth different major influences. The simulation should also account for airtightness loss over time. Consider usign a energy simulation tool that you are familiar with and /or has a big community to help in case of things not working. Use existing case studies to test how well the generated model behaves against the case study data.

iii. Transition probabilities creation: In case that no available data exist of the probability of failure or degradation of a component, an extra step will be to create transition probabilities to simulate environmental uncertainties. Depending on the material in question and its degradation process based on the factors influencing this degradation, different probability distributions will need to be modeled. Even though in this case Gamma distribution was used, insulations usually require Weibull distribution. This step involves understanding and incorporating the equations that describe the overall degradation curve and the probability distribution. Available data of component behaviour should be analyzed and fitted in order to create the mean curve. The general process can be found in the appendix.

iv. Optimization tips: Optimize the environment before going any further. Consider if you can diminish the state space somehow, and use numpy arrays which accelerate the computational process. Precalculate , if possible the rewards and the transitions if needed and store them in sparse matrices to save space. Even if deep learning is not going to visit all the states and actions, it will require a lot of computational power. By assuring that the environment has optimized performance , the code will run much faster and the convergence might happen even earlier. Even more, develop the environment, the simulations , the probability generation and other codes in separate scripts and as functions which can be incorporated into the main script. This will allow debugging to happen more easily.

v. Reinforcement Learning Algorithms: Given the increasing complexity of the problem formulation and the expansion of the state space, consider employing Deep Reinforcement Learning algorithms from the beginning. A multi agent approach might be recommended as it is able to handle with more ease and less complexity the different component states.

vi. Evaluate the environment and the policy by conducting a sensitivity analysis and testing different rewards to see if the problem as has been formulated through the environment is performing correctly.

6.10. Sub-Questions

"How do we formulate the staged retrofitting as an MDP (Markov Decision Process) problem?"

Since there wasn't an actual case of literature to base the answer the formulation of the MDP was based on different approaches and using a basic understanding of how Markov Decision Processes work. Inspiration was drowned from the examples of (Ferreira et al., 2023) , (van den Boomen et al., 2020) and the online tutorials of Prateek Bhustali.

To formulate the staged retrofitting problem as an MDP, we need to define:

1. States: Represent the condition of the building, and might include the current state of components (e.g., insulation, windows, heating systems). Incorporate the time in the state space I order to be able to determine the time of each retrofitting action and avoid infinite horizon problems.

2. Actions: Define possible retrofitting actions such as installing insulation, replacing windows, changing heating systems, or conducting maintenance. Those actions must reflect somehow to the state space's possible changes. Define also how the actions will change the status of the environment and what is the cost of each action. Major maintenance might keep the house environment in the same energy performance state for a bit more time.

3. Transition Probabilities: Decide if transitioning from one state to the next will happen deterministically or not stochastically. If stochasticity is involved calculate and incorporate the probabilities of moving from one state to another after taking a specific action.

4. Observability: Define if the states are observable or not. That means , define if the states are completely known to the agent or not.

5. Rewards: Define the rewards associated with each state-action pair, which could include costs, energy costs or savings, budgets etc . Remember that the goal of the optimizers is to find the max reward, and so the rewards should be shaped in a way that reflect the objective function that will be tried to be achieved.

6. Policy: Develop a policy that specifies the best action to take in each state to maximize the cumulative reward over time. For example, in the case that certain threshold over energy performance degradation shouldn't be reached, introduce a big penalty like doubling the energy bills or enforcing a retrofitting action with higher costs that normal.

"Which reinforcement learning algorithm should we consider for solving this problem?"

Even thought there wasn't enough time to delve deeper into the different reinforcement learning methods it has become apparent that the methods that should be considered will require to be able to handle big state spaces. Given the complexity and size of the state and action spaces in the retrofitting problem and based on the points made in (Bhustali, 2023), it is possible that Multi-agent Deep reinforcement learning methods should be considered that can efficiently handle the complexity of instepction and maintenance problems.

"How can we simulate the scenario of the building components (energy) degradation?"

To simulate the degradation of building components and their impact on energy performance tools like EnergyPlus can be used. The use of Grasshopper visual programming and the Ladybug plug in can simulate with quite good accuracy the building performance while provide extensive community support. Eppy can be used to generate different scenarios based on the exported idf file however there is a speculation that the process can be done directly through Grasshopper and the simulation results stored directly in CSV files.

The models have to simulate the different R values or conductivity values degradation scenarios of the different major components that affect the most the performance of the building. These components usually are : the windows, façade, roof, ground floor , HVAC systems and airtightness.

“How do we validate the model?”

Theoretically, after the environment is created, we can do sensitivity analysis to test the behaviour of the algorithm. In this case, the problem formulation was tested by comparing the policies against the do- nothing policy. By changing the rewards, discount factor and the costs of actions, the dynamics of the environment can be assessed.

However, in order to be able to assess the policies correctly, the results of the optimization should be compared against real life scenarios.

6.11. Assumptions

The list of assumptions that were taken in order to work on this problem can be found here:

- Economy is in a stable 3% growth.
- Building’s performance changes only through ageing factors of the insulation and doesn’t depend on other factors.
- The actions will return the component and the system in an original state without any loss of functionality or possible worst state
- All actions can be taken from all states
- The policy in deterministic
- No government policies exist that might affect the investment costs.
- PIR is also plastic foam insulation. In this thesis it is assumed that the material is degrading with the same rate as the EPS. However, separate research should be conducted for its exact properties.
- The energy, retrofitting, material and other prices are certain and will not be changed

8. REFLECTION

Embarking on this project centered on retrofitting planning optimizations using Dynamic Programming (DP) and Reinforcement Learning (RL) techniques had a primary objective: to create a foundation for future projects in this domain. However, navigating this novel terrain was not without its challenges.

A significant issue encountered was the fragmented nature of existing knowledge and terminology. The terms "retrofitting" and "renovation" often overlapped, and a fundamental theory of building degradation proved elusive. Instead, various disparate equations and techniques describing the degradation process of different components were uncovered. A particularly concerning gap was the lack of comprehensive data on the behavior of building envelopes over time, highlighting a crucial area for future research. More extensive data on the evolving behavior of buildings would undoubtedly enhance the accuracy of results obtained through solvers.

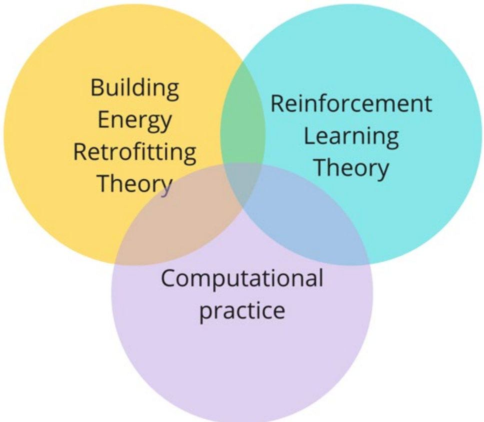


Figure 70 Areas of exploration

Throughout my master's program, I engaged with different scientific domains. Working with RL theories proved particularly challenging due to my lack of prior knowledge in the domain. Understanding the theoretical equations in the literature was one of the most demanding aspects. Expanding my knowledge across such diverse areas stretched my focus thin, preventing deeper exploration in many areas. My biggest regret was not being able to develop a deeper undersanding of Deep Reinforcement Learning approaches. However, the challenges faced with Value Iteration helped me expand my knowledge of optimization techniques and explore additional methods to improve algorithm performance.

Despite rigorous efforts, the development of reinforcement learning algorithms and the environment remained in an early stage due to time constraints. It is clear that further work is needed in these areas to realize their full potential.

Given the nascent nature of the domain, numerous assumptions had to be made, affording considerable autonomy in decision-making. This latitude allowed the project to evolve

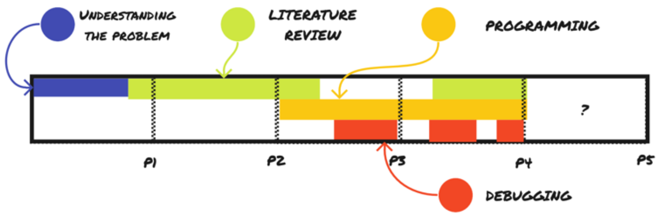


Figure 71 Graph expressing what I did throughout the process

based on prioritized aspects of the process. Emphasis was placed on early methodology, such as building degradation simulations, generating transition probabilities, and formulating the problem as a Markov Decision Process (MDP). Focusing on these basics allowed for realistic goal-setting.

Engaging in this process not only honed my coding skills but also deepened my understanding of sequential decision-making theories and reinforcement learning. This endeavor was a lesson in learning how to learn and develop projects from scratch, relying on logical arguments and resource gathering to make informed decisions.

While the project did not reach full maturity, it established essential guidelines for future research. Exploring different degradation techniques underscored the utility of tools like Geomeppy and affirmed Grasshopper's user-friendly approach to data simulation. Developing the environment from scratch, though challenging, provided valuable insights into its inherent bottlenecks. Furthermore, Value Iteration yielded a comprehensive understanding of the environmental variables that influence optimal policy, guiding future efforts.

In writing this report, I incorporated as much literature, detail, and explanation as possible to facilitate future research in this area. The lack of more equations that might have made this easier to understand stems from my own challenges in formulating them correctly.

In conclusion, this experience was enriching both personally and scientifically. It underscored the importance of perseverance in tackling complex challenges and the value of collaborative mentorship in navigating uncharted territories.

9. THANK YOU NOTE

Conducting a master thesis is not an easy task, especially when it touches on topics at the edge of the architectural field with much unexplored territory. I was fortunate to find professors like Charalampos and Michela who agreed to mentor me on such an interesting and challenging project. I deeply thank them for believing in me.

A big thank you goes to Pablo Morato as well for being my unofficial mentor in all matters regarding Data Science. From helping me understand transition probabilities to analysing algorithm results, he has been an inspiration and a tremendous companion throughout this journey.

Anna Maria Koniari, who started her PhD thesis on a similar topic a few months earlier, generously shared her knowledge with me. Watching her progress motivated me to advance at my own pace.

Lisa Marie, who mentored me on computational processes, was incredibly kind, understanding, and patient with my irregular schedule. Her support extended beyond academics, providing great emotional support to many students this year.

Pratteek Wahi and Prateek Bhustali provided valuable feedback at the beginning and end of this thesis, offering guidance despite their busy schedules with many other students.

Outside of academia, I owe thanks to a few others. My friends and fellow students from the Building Technology track believed in me and provided mental support throughout this process. Thank you, Fieke, Pavan, Tahir, Juan, and Deva. Thanks also to the AI group, the Minions support group: Jair, Kuba, Shreya, and Sashipa.

I am grateful to my brother, Odysseas, for frequently checking on me, trying to understand my project, and helping develop the gamified version of the problem, which made it easier to explain to others.

Lastly and most importantly, to Jacob, my partner. Thank you for your unwavering support, patience, and love. You have been my mentor, inspiration, support, and friend through it all.

Thank you.

10. APPENDIX

15.1. Further Explanations

15.1.1. Transitions example

To give an example, imagine a house with three states of energy demand: Good, Medium, Bad. There is some uncertainty if in the next state the house will have reached 'Medium' energy demand or will have stayed in 'Good'. More accurately it has 60% probability reaching the 'Medium' energy demand state based on the fact that no action is performed to prevent that of happening, and 40% of staying in 'Good' state.

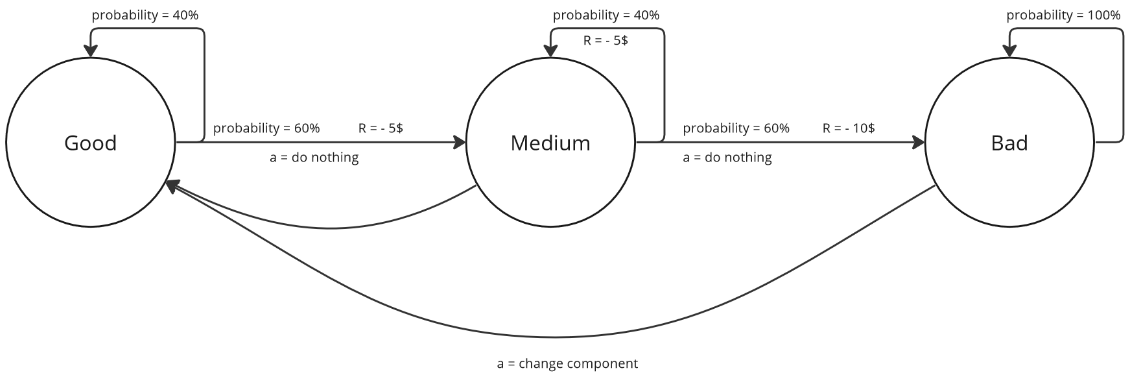


Figure 72 Transition probabilities between states (by author)

A transition matrix represents the probabilities of each state reaching any other state. In the matrix, the rows signify the **now** while the columns signify the **next state transition.** For example, this matrix maps the probabilities of jumping from each state to another if no action is taken based on the above graph.

$$P(\text{if no action is taken}) = \begin{matrix} 0.4 & 0.6 & 0 \\ 0 & 0.\mathbf{4} & 0.6 \\ 0 & 0 & 1 \end{matrix}$$

If we want to find the what is the probability of being in each state after two transitions, say from Good to Bad, passing from an intermediate state, Medium, we can multiply the rows of the matrix with the corresponding columns and then add them up. Matrix multiplication works by multiplying the elements of each row of the first matrix by the corresponding elements of each column of the second matrix, and then adding them up.

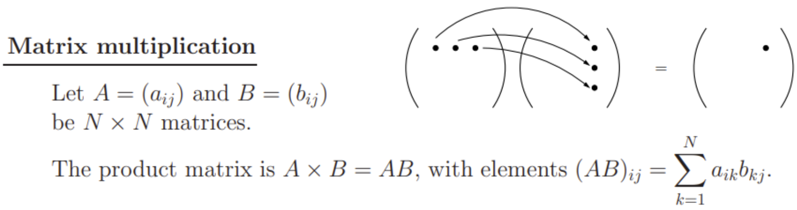


Figure 73 Matrix multiplication graph[17]

15.1.2. Objective function cost equation analysis

The objective function presented in the literature involved a number of different parameters that had to be calculated. Some of them are analyzed below however, for better understanding, reference the paper provided by (Maia et al., 2021) .

$$A_t = (INC_t \cdot s) + A_{t-1}$$

- INC_t = household income (EUR)
- s = allocation factor of total annual incomen on energy related expenses (%)

$B_t \geq IC_t + EC_t + OMC_t$,

with $B_t = A_{t-1} \cdot (1 + l)$

Where:

- B_t = budget restriction [B]
- IC_t = investment cost of retrofitting measures [EUR];
- EC_t = annual running energy costs [EUR/a]
- OMC_t = annual running operation and maintenance costs [EUR/a];
- l = loan [%].

Investment costs for retrofitting steps, such as improving the building envelope or installing active systems, were determined by considering energy-related investment costs,

---

[17] https://www.stat.auckland.ac.nz/~fewster/325/notes/ch8.pdf

maintenance investment costs, the probability of material aging, and a binary control variable indicating whether the measure is performed each year.

$$IC_t = \sum_t [(1 - p_{t,i}) \cdot ICman_{t,i} + ICer_{t,i}] \cdot x_{t,i}$$

where $x_{t,i} = 1\ or\ 0$ and $p_{t,i} > 0.05$

- IC_t = total investment costs [EUR]
- $ICer_{t,i}$ = energy-related investment costs, for each retrofittig step (i) [EUR]
- $ICman_{t,i}$n = maintenance investment cost, for each retrofitting step (i) [EUR]
- $x_{t,i}$ = binary variable (1 or 0) [-], if the step i is performed in the time t
- $p_{t,i}$= ageing process probability of building materials or technical system of step i

In this study, the energy costs were calculated based on final energy demand and energy prices of the corresponding sources. Retrofitting measures lead to reductions in final energy demand, resulting in energy savings that depended on energy-related investment costs.

$$EC_t = \sum_t fed_{t,i} \cdot pr_{t,i}$$

Where:

- EC_t, energy costs [EUR/a]
- $fed_{t,i}$, final energy demand [kWh/a];
- $pr_{t,i}$, energy price [EUR/kWh].

Operation and maintenance costs for active systems were related to investment costs and an operation and maintenance factor.

$$OMC_t = \sum_i IC_{t,i} \cdot f_{OMC,i}$$

Where:

- OMC_t = operation and maintenance costs [EUR/a];
- $IC_{t,i}$ = investment costs of active system [EUR];
- $f_{OMC,i}$=operation and maintenance factor [%]

- Below, the table of the 27 separate states and their energy prices can be seen.

15.1.3. Gamma distribution basics

The gamma distribution is flexible and can model a wide range of shapes depending on the values of (f) and (g). It is particularly useful for modelling processes where the event rate is not constant, which makes it suitable for various applications including modelling waiting times or the time until a certain number of events occur(*Scipy.Stats.Gamma — SciPy v1.13.1 Manual*, n.d.).

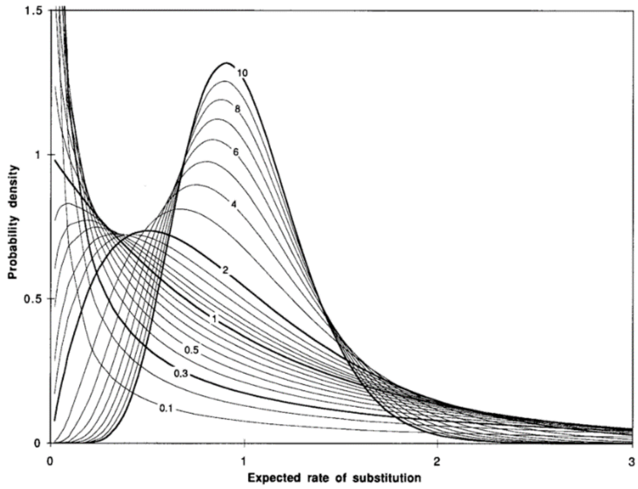


Figure 74 Gamma distribution with different shape parameters(*Deep Variational Inference. Studying Variational Inference Using DL… | by Natan Katz | Towards Data Science*, n.d.)

Probability Density Function (PDF)

The probability density function (PDF) of a gamma distribution for a variable x is given by the equation

$$Gamma(x|\alpha,\beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}$$

Where

- x is the variable of interest (e.g., time until failure, amount of damage).
- α is the shape parameter which determines the shape of the distribution.
- β is the scale parameter determines the scale (spread) of the distribution.

The gamma function, symbolized as Γ(α), plays a crucial role in enabling the gamma distribution to effectively represent continuous events, such as degradation rates that evolve over time. By incorporating time-dependent parameters, this function empowers the model to adapt to changing circumstances, resulting in a more precise depiction of real-world processes. The gamma function is defined as:

$$\Gamma(\alpha) = \int_0^{\infty} t^{\alpha-1} e^{-t} dt$$

Where

t is a variable that goes from 0 to infinity.

$t^{\alpha-1}$ raises t to a power based on α.

e^{-t} is an exponential decay function that ensures the integral converges (the sum doesn't go to infinity).

15.1.3.1. GAMMA DISTRIBUTION , DEGRADATION AND TIME RELATION

The gamma process is a type of stochastic process where each time increment follows a gamma distribution. A gamma process can be used to model the deterioration of the insulation over time. A nonstationary gamma process allows the parameters of the gamma distribution to change over time, which better captures the real-world dynamics of the material's deterioration.

The gamma distribution is a continuous probability distribution that is generally used to model the time until when an event occurs. It is greatly used in medicine and engineering to determine the probability of death or failure of a system.

The gamma distribution's Probability Density Function (PDF) at any time t is given by the equation:

$$Gamma(DI|f(t), g(t)) = \frac{g(t)^{f(t)}}{\Gamma(f(t))} DI^{f(t)-1} e^{-g(t)DI}$$

where DI is the damage index at time t, and g(t) is a non-negative time-varying scale parameter function, and f(t)), a non-negative time-varying shape parameter function. Those functions reflect how a material's deterioration rate changes as it ages. The gamma function $\Gamma(f(t))$ is used to define the distribution.

The parameters f(t) and g(t) can be estimated based on analyzed condition data. This involves fitting a regression model to the observed values to predict the mean and variance of the material degradation over time. The mean damage µDI(t)) at time t is defined as the product of f(t) and g(t), while the standard deviation σs(t)) is the square root of the product of f(t) and g(t).

$$\mu_{DI}(t) = \frac{f(t)}{g(t)}\ , \sigma_s(t) = \frac{\sqrt{f}(t)}{g(t)}$$

As explained by (Saifullah et al., n.d.) for any given time, say, t1 and t2 where t1<t2, the increase in the damage index DI(t2)−DI(t1)) follows a gamma distribution with parameters f(t2)−f(t1) and g(t2)' .

15.1.4. Workflow Overview – Stage 1

The process of formulating the methodology passed through various stages. The stages were then summarized in two parts. In the thesis, the final methodology is described. In order to keep consistency, stage 1 that described the summary of the initial tests and all the results that led to stage 2 were moved to the appendix. You can reference them here in order to understand how the methodology evolved to the final version.

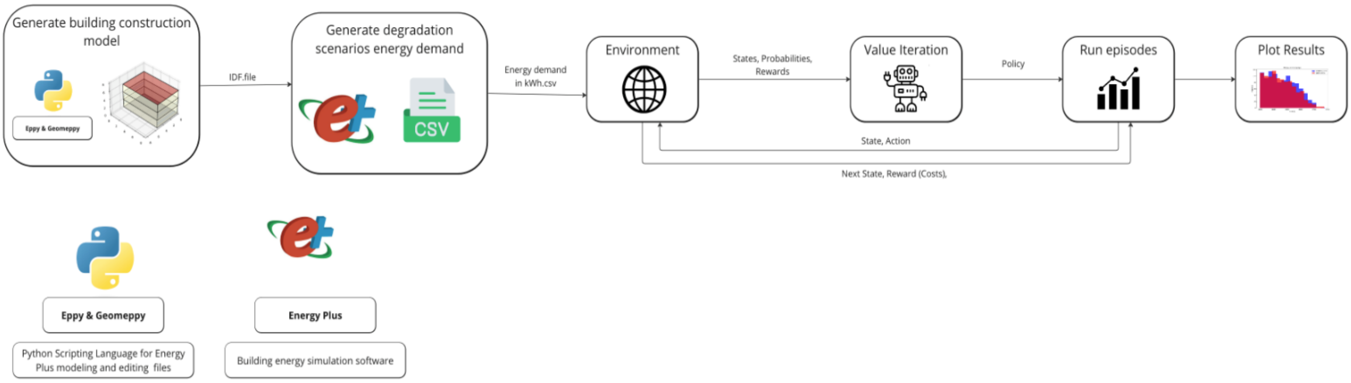


Figure 75 Workflow of the first stage. The problem is defined as an MDP in the environment script. The model and the simulation o fthe buidling degradations are done in separate scripts and imported. The problem is then solved wih a dynamic programming method called value iteration in a separate script. The extracted policy is used to run a series of episodes that compare the behaviour of the policy against a benchmark. The results are then plotted in a separate script.

The initial idea pointed on the translation of the problem into an MDP problem. An environment needed to be build in order for the problem to be solved with a chosen Reinforcement Learning method. As the actions would reflect to the energy performance of the building and the Return, energy simulations of the building's model needed to run to generate all the possible scenarios. After the completion environment , a series of episodes are run to test the model's behaviour by checking the optimal policy against the benchmark criteria. If proved to be correct, the optimal policy could be extracted and assessed.

In this first stage of project development, the initial idea of the Markovian representation of the environment was developed. To simulate changes in energy demand, a case study was chosen, and a model was created using the geomeppy library in Python.

In this investigative stage, for the sake of simplicity, stationary transition probabilities were used with arbitrary numbers. The focus of this stage was to investigate the general methodology needed to address the problem at hand and to decide which techniques needed to be reevaluated.

15.1.4.1. THE PHYSICAL PROBLEM

The physical problem addressed in this study revolves around existing building stock, which is unlikely to undergo significant changes by 2050. Thus, an existing house was chosen rather than a new construction. A pre-1945 Terraced house was selected as the initial case study, with careful consideration given to the chronology of construction due to variations in construction properties such as window to wall ratios and space for insulation.

The characteristics of these terraced houses were derived from information provided by the Dutch Ministry of the Interior and Kingdom Relations regarding typical Dutch home typologies (Ministerie van Binnenlandse Zaken en Koninkrijksrelaties, 2022).Notably, terraced houses from this period were often built without insulation, with those predating the 1930s lacking cavity walls. Typically, window frames were constructed from steel or softwood and featured a single glass layer. These houses constitute 4.5% of the Dutch building stock, with 73% being owner-occupied.

Construction details obtained were utilized for the development of a digital model of the house for energy performance analysis, with the building layout and other characteristics being drawn from the thesis of Naeem Kantawala (Kantawala, n.d.).

Roof insulation was represented by 200 mm EPS insulation plates, while the ground floor was insulated with 300 mm EPS sheets. Since the existence or not of a cavity wall depended on the actual chronology of the house, it was assumed that the house dated pre 1930s. These walls were modelled as two layers of 100 mm with no cavity, resulting to an R value of 0.35 without insulation. This allowed to assume a retrofitting measure of placing the insulation on the interior side of the wall. The thickness of the insulation was assumed to be 100 mm.

Windows were assigned a U value of 2.90, and the window to wall ratio was set at 35%. The thermal conductivity of EPS insulation was derived from available data found at the site of IES[18] entailing detailed information of thermal conductivity, specific heat capacity, and density.

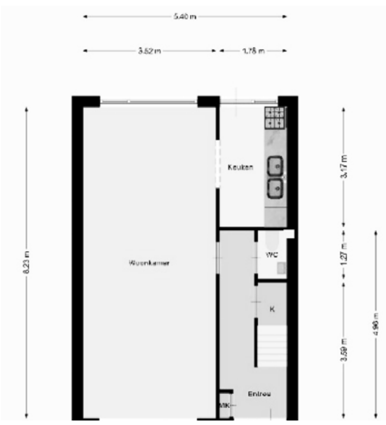


Figure 76 Ground floor layout(Kantawala, n.d.)

Figure 78 First floor layout(Kantawala, n.d.)

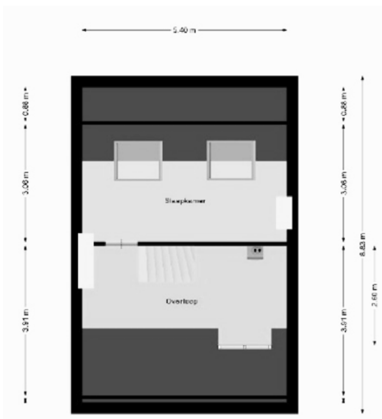


Figure 77 Second floor layout (Kantawala, n.d.)

[18]https://help.iesve.com/ve2021/table_6_thermal_conductivity__specific_heat_capacity_and_density.htm#

Table 1: House model construction aspects (Own work)				
Aspect	**Details**	**Area (m²)**	**R values**	**Measure**
Wall	Two layers of 100 mm bricks with no cavity..	63.8 (non-adiabatic area)	0.35*	100 mm EPS
Roof	Flat roof	44.4	0.35*	200 mm EPS (Expanded Polystyrene) insulation plates
Ground Floor	Crawl space ceiling insulation	44.4	0.15*	100 mm EPS sheets.
Windows U-value	Single glazing,	35% WWR	2.90	-
*Non insulated surface values				

Table 2: EPS Insulation Properties		
Conductivity (W/mK)	Sp. Heat Capacity (J/kgK)	Density (kg/m3)
0.035	1400	25

15.1.4.2. ENERGY DEMAND SIMULATION

To determine the change in energy bills due to retrofitting insulation, it was necessary to analyse a house model and simulate various scenarios of thermal resistance or material conductivity degradation. EnergyPlus, an open-source software widely utilized for building energy simulations, was selected as the primary tool for this analysis.

The simulations needed to consider various states of thermal conductivity degradation for different components. Three modelling tools were initially evaluated for developing the house model:

1. **DesignBuilder**: This tool integrates a user-friendly environment for designing 3D models with EnergyPlus operating in the background. Despite its widespread use, DesignBuilder was deemed unsuitable due to its limited community support, non-intuitive interface, and the requirement for manual execution of simulations since the parametric tools kept crashing.

2. **Ladybug with Grasshopper**: Ladybug offers extensive community support and ample documentation, facilitating the resolution of various modelling issues. However, each simulation run with Ladybug took approximately 15 seconds. Even though it is possible to

connect Grasshopper scripts with Python using Hops components, the cumulative time required to run numerous simulations across various episodes was deemed impractical in that stage.

3. **Geomeppy**: This Python library extends Eppy, a scripting language for EnergyPlus, to enhance capabilities for geometry manipulation. Geomeppy allows the creation of 3D models, and manipulation of the idf.files , incorporating all the building model information, that are executable by EnergyPlus. Geomeppy was considered advantageous due to its speed, with each simulation running in approximately 5 seconds, and the ability to automate simulations through Python scripting.

Table 3: Pros and cons of energy modelling tools

Modeling Tool	Pros	Cons
DesignBuilder	Intuitive 3D modeling environment	Limited community support, manual simulations, unstable parametric tool
Ladybug with Grasshopper	Extensive documentation, strong community support	Slow simulation time, impractical for numerous episodes
Geomeppy	Fast simulation time, automated through Python	Initial learning curve, requires scripting knowledge and good understanding of Energy plus dynamics

Using Geomeppy was considered to be the most efficient approach due to its automation capabilities and faster simulation times., while allowing to work directly through python for the whole workflow.

Following the online documentation, a simple geometry was modeled using parameters from the existing case study. The constructed model was then encapsulated in a function to iteratively run simulations across all possible combinations of insulation degradation scenarios. With this approach 27 distinct deagradation scenarios were simulated and exported in csv form into the environment script.

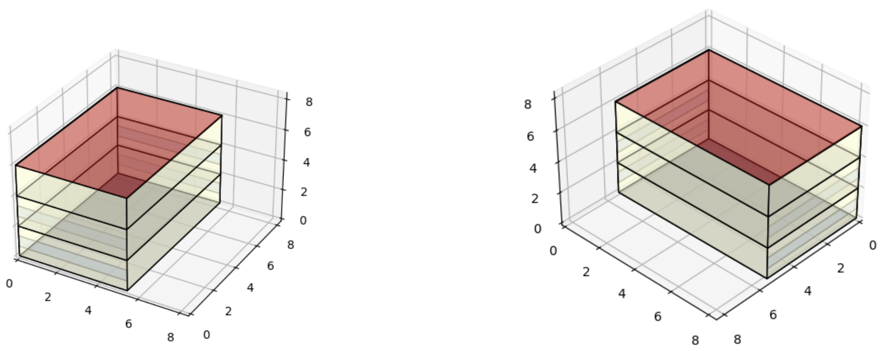


Figure 79 Examples of the simple geometry of a house that was developped using geomeppy library in python (own work)

15.1.4.3. State space (S)

In the context of the retrofitting problem, the states of the Markov Decision Process (MDP) represent snapshots of a house's condition, focusing on key envelope elements: windows, external walls, ground floor, and roof. These states can be either fully observable or partially observable. For simplicity, deterministic states were used where the observed states accurately reflect the actual conditions.

To model this, the three-step renovation strategy was adopted as proposed by (Maia et al., 2023), which targets significant components such as the roof, facade, and ground floor or cellar ceiling surfaces due to their substantial impact on a building's performance and extensive coverage of the envelop. Each decision variable in the MILP model corresponded to a renovation stage occurring in a specific year. In a similar manner, states were defined as combinations of the current statuses of the three major components: roof, facade, and ground floor, reflecting the discrete stages of renovation and binary decisions over time.

Inspired by Ferreira et al., (2023),who proposed uneven discrete degradation stages to quantify performance loss in building components, the degradation of the components (roof, facade, ground floor) in three stages: 0%, 20%, and 50% was defined. The separation was based on the mean degradation of plastic insulation at each degradation stage.

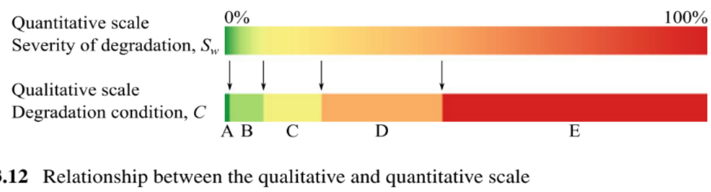


Figure 80 Relationship between the qualitative and quantitative scale(Ferreira et al., 2023)

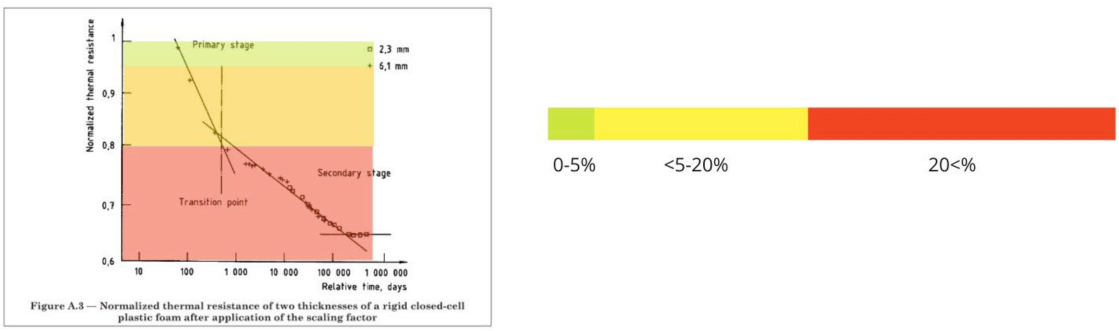


Figure 81 Relationship of quantitative degradation as was assumed based on the available degradation graphs (own work)

With three degradation stages per component, a total of 27 possible state combinations were derived . Each state represented by a tuple indicating the degradation stage of each component, providing a comprehensive overview of the house's condition for effective decision-making in the retrofitting process. For example, a state can be represented as:

(Roof degradation, Façade degradation, Ground floor degradation)

where each component degradation can be 0% (no degradation), 20% degradation, or 50% degradation.

In a more mathematical way this can be represented as :

$$S = \{(r, f, g) | r, f, g \in \{0\%, 20\%, 50\%\}\}$$

Table 4: States example

State number	Roof Degradation (%)	Façade Degradation (%)	Ground floor degradation (%)
0	0	0	0
1	0	0	20
2	0	0	50
. . .	. . .	. . .	. . .
25	20	50	50
26	50	50	50

15.1.4.4. ACTION SPACE (A)

The initial action space in the model was defined as a discrete action space comprising four possible actions:

- **Action 0**: Do nothing
- **Action 1**: Change Roof Insulation
- **Action 2**: Change Façade Insulation
- **Action 3**: Change Ground Floor Insulation

The process of changing insulation is influenced by several factors, including the type and thickness of the insulation material, the method of installation, and the specific surface being insulated. Numerous retrofitting measures are available, often detailed on government websites, offering guidance and support for energy efficiency improvements.

For this initial approach, the retrofitting actions[19] involved replacing the existing EPS (Expanded Polystyrene) insulation with a new, intact version. The following assumptions were made for simplicity and to facilitate a better understanding of the Markov Decision Process (MDP) model:

- **Perfect Installation**: It was assumed that the retrofitting actions were carried out perfectly, with no imperfections that could degrade performance.
- **Restored Performance**: It was assumed that the building's thermal performance would be fully restored to its original state post-retrofitting, implying that the building would function as efficiently as when it was new.

These assumptions, while not entirely realistic, helped streamline the model and provided a clearer perspective on the potential benefits of retrofitting actions

Table 5 : Retrofitting measures for house typology of stage 1

Name	Info	Placement	Width (mm)	RC value ($m^2.K/W$)	Price euros per m2
WB374 – Bio EPS		Flat roof	200	6.5	303.35
WB002d - EPS		Crawl space ceiling	100	2.9	28.91
WB008b -EPS isolation	Decorative plaster finishing	Exterior wall	100	2.6	162.73

Table 6 : Actions of problem formulation in stage 1

Action	Interpretation	Costs (Euros)
0	Do nothing	0
1	Change Roof Insulation	13.468,74
2	Change Façade Insulation	10.382,174
3	Change Cellar Ceiling Insulation	1.283,604

[19] In a lot of predictive maintenance problems, the actions include major , minor maintenance actions. As there is ambiguity about how retrofitting should be categorized , the assumption was made that renovation is major maintenance, retrofitting is considered medium maintenance , minor maintenance being superficial fixes in the envelop skin.

15.1.4.5. Rewards (R)

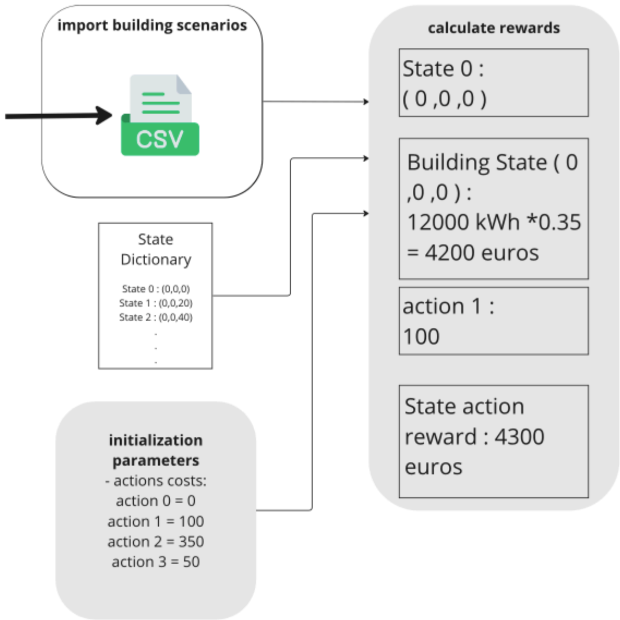


Figure 82 Workflow graph of the calculate rewards function of the script (own)

Based on the developed house typology and the different scenarios of energy demand , 27 variations of energy consumed per square meter were created. The simulated energy demand was multiplied by 0.35, the mean value of euros per kWh in the Netherlands, based on the average provided by GlobalPetrolPrices.com (September 2023)[20]. In summary, our rewards reflect the financial impact of investment costs and energy bills, allowing us to assess the effectiveness of different retrofitting strategies in minimizing expenses over time. The table with all the degradation scenarios can be found in the appendix. Discount Factor

In the context of a Markov Decision Process (MDP), the discount factor is crucial. It correlates to the concept of valuing future rewards relative to immediate ones. In an MDP, decisions are made at each state to maximize the cumulative reward over time. The

[20] According to the information from the official Dutch government , in 2023, the maximum tariff for electricity was €0.40 per kWh for up to 2,900 kWh of electricity used1. This price cap was introduced to protect households and other small-scale users from soaring energy prices(*Netherlands Electricity Prices, September 2023 | GlobalPetrolPrices.Com*, n.d.)

discount factor, denoted by γ reduces the value of future rewards, reflecting the principle that immediate rewards are typically more valuable than future rewards.

15.1.4.6. DISCOUNT FACTOR

To maintain consistency in the analysis, a stable interest rate of 3% was assumed. This interest rate influences the discount factor, which is used to adjust the value of future cash flows to their present value. For a 3% interest rate, the discount factor is calculated as 0.97, indicating that future cash flows are discounted by 3% each year.

$$\gamma = 0.97$$

15.1.4.7. <u>Transition probabilities (T)</u>

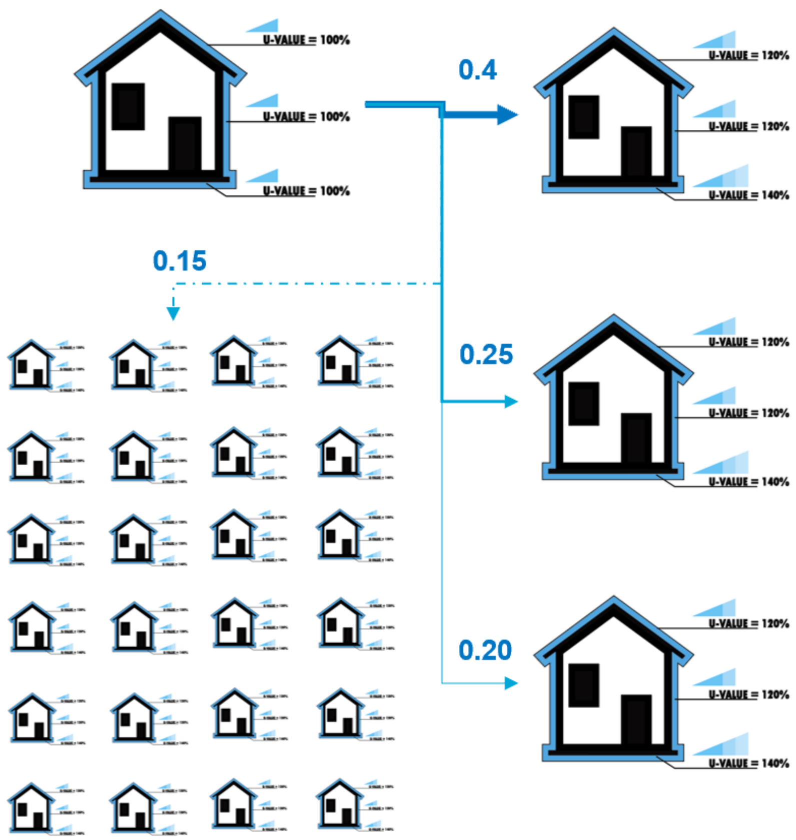


Figure 83 State transitions graph. Each state reaches 27 other states based on the joint probability of each material reaching another state of deterioration (Own work)

To begin with, at an early stage of the analysis, constant transition probabilities were assumed, and a stationary probability was employed for initial testing. However, due to the state being defined as a discrete tuple representing the states of three components, it became necessary to articulate the transition from one state to the next as the joint probability of each component transitioning to a specific next state as outlined in the subsequent state tuple.

$$P = P_{Roof} \cup P_{Facade} \cup P_{Ground\ Floor}$$

Thus, the joint probability of transitioning from one state to another can be expressed as the product of the individual transition probabilities for each component though the equation :

$$P(s'|s,a) = P_{Roof} \times P_{Facade} \times P_{Ground\ Floor}$$

This necessitated the formalization of a stationary transition probability matrix P, where each entry denoted the probability of transitioning from one state to another.

$$P = \begin{matrix} 0.8 & 0.2 & 0 \\ 0 & 0.8 & 0.2 \\ 0 & 0 & 1 \end{matrix}$$

In this matrix, each row corresponds to the current state, while each column represents the probability of transitioning to the next state.

In the same manner, the transition matrix for the actions was denoted as:

$$P = \begin{matrix} 1 & 0 & 0 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{matrix}$$

To compute these probabilities across the entire state space, one hot encoding was employed. This method allowed for the iteration through different transition matrices for each action. Depending on whether a particular component was to be changed or not, the relevant transition matrix was selected. The resulting probabilities of transitioning from one state to another were then multiplied to obtain the joint probability of the entire state transition.

Ensuring the accuracy of the new transition matrix involved summing each row to verify that the probabilities totaled to 1, with a small margin of error accounted for due to numerical precision limitations in Python. This validation process ensured that the probabilities accurately reflected the transitions between different states based on the actions taken.

One hot encoding was employed to represent the various actions taken concerning each component's transition within the state space.

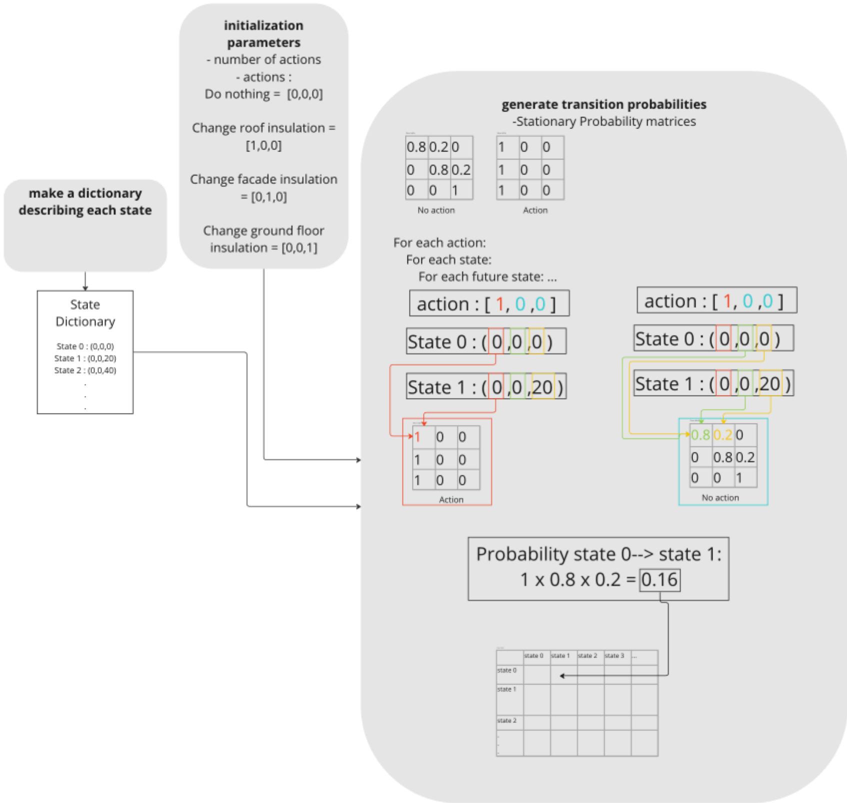


Figure 84 The workflow of the script to calculate the state transition probabilities using one hot encoding (own)

Example:

We start with a state tuple where all components are in perfect condition, denoted as State1=(0,0,0). The subsequent state we aim to determine the probabilities of reaching is State2=(0,1,1). Using this encoding scheme, each action corresponds to a specific binary vector:

- Action 0: [0,0,0]
- Action 1: [1,0,0]
- Action 2: [0,1,0]
- Action 3: [0,0,1]

In this case, action 1 is selected. We examine the probability of transitioning from the current state to the next state for each component:

- Roof state: 0, Future Roof state: 0, Action: 1 → Probability: 1
- Facade state: 0, Future Facade state: 1, Action: 0 → Probability: 0.2
- Floor state: 0, Future Floor state: 1, Action: 0 → Probability: 0.2

The joint probability is computed as the product of these individual probabilities, resulting in 1×0.2×0.21×0.2×0.2. These binary vectors serve as indices to select the relevant transition probabilities from the transition model matrix. By utilizing this one hot encoding scheme for each component and action, efficient computation of joint probabilities for transitioning between states across the entire state space is achieved.

Bottom of Form

15.1.4.8. ENVIRONMENT CREATION

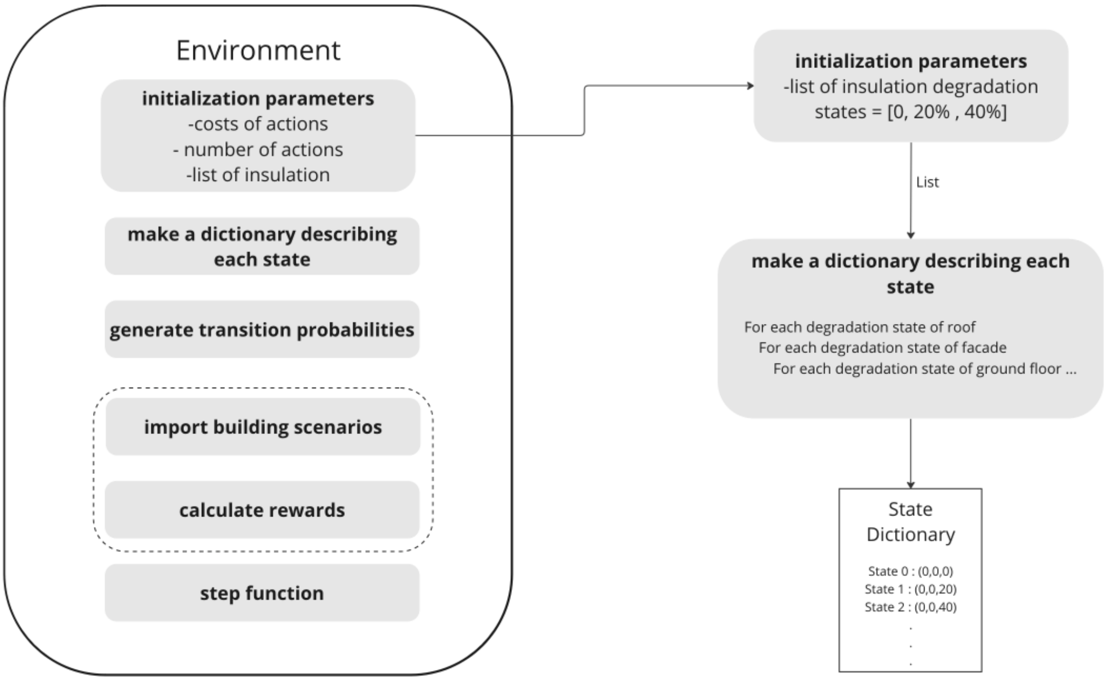


Figure 85 Workflow depicting the initial environment with its functions and the creation of the states (own work)

Initially, an environment was constructed based on a toy environment outlined by Prateek Bustali. However, since Gymnasium libraries, which are typically utilized for running Deep Reinforcement Learning agents, did not support value iteration, adjustments to the environment were necessary.

The environment simulated represented a house undergoing degradation over time, with the goal of minimizing renovation costs and energy bills. This flowchart illustrates how the environment functions, from initialization to episode termination, and how actions influence state transitions and rewards. The basis code for the environment and the value iteration method was created by Prateek Bhustali [21].

Inspiration for the environment was also drown from Bechir Trabelsi[22]

Initialization parameters: The environment is initialized with parameters such as house size, number of damage states, and action space. These include actions like fixing the roof, wall, or facade, or doing nothing.

State Space Generation: The state space of the house is generated based on the number of damage states for each component (roof, wall, cellar). This creates a discrete set of possible states for the house.

Transition Model Creation: A transition model is built to determine the probabilities of transitioning from one state to another based on actions taken. This is done by calculating joint probabilities for each component's transition using a one-hot encoding scheme.

Reward Calculation: When an action is taken, the environment calculates the associated reward. This includes renovation costs and energy bills, which are influenced by the degradation of house components.

State Transition: Upon taking an action, the environment transitions to a new state based on the transition probabilities calculated earlier. The next state is chosen randomly according to these probabilities.

Episode Termination: The episode continues until a certain time limit is reached, when the lifespan of the house ends. At this point, the episode terminates.

15.1.4.9. OPTIMIZATION METHODOLOGY

Different Reinforcement Learning methodologies were considered based on the formulation of the problem. Reinforcement learning involves exploring potential scenarios and devising an optimal policy based on the mean return. Value iteration, on the other hand, is not a dynamic learning method but rather falls under dynamic programming. Dynamic programming serves as the foundation of reinforcement learning. Unlike the reinforcement learning approach, which is model-free, value iteration is a model-based approach: The agent must examine all possible states and actions to create the optimal

---

[21] (https://gitlab.tudelft.nl/pbhustali/mdp_tutorials//blob/main/Inspection_Maintenance_Example/MDP.ipynb?ref_type=heads)

[22] (https://bechirtr97.medium.com/finding-the-optimal-mainteance-policy-via-markov-decision-process-and-policy-iteration-algorithm-c7b604d16fb1)

policy to follow. Value iteration was deemed a suitable starting point for two primary reasons:

1. due to the manageable size of the state space at the time
2. because it provided a clear understanding of the outcomes.

The value iteration process was segmented into three distinct parts.The first part involved the creation of the environment as described above.

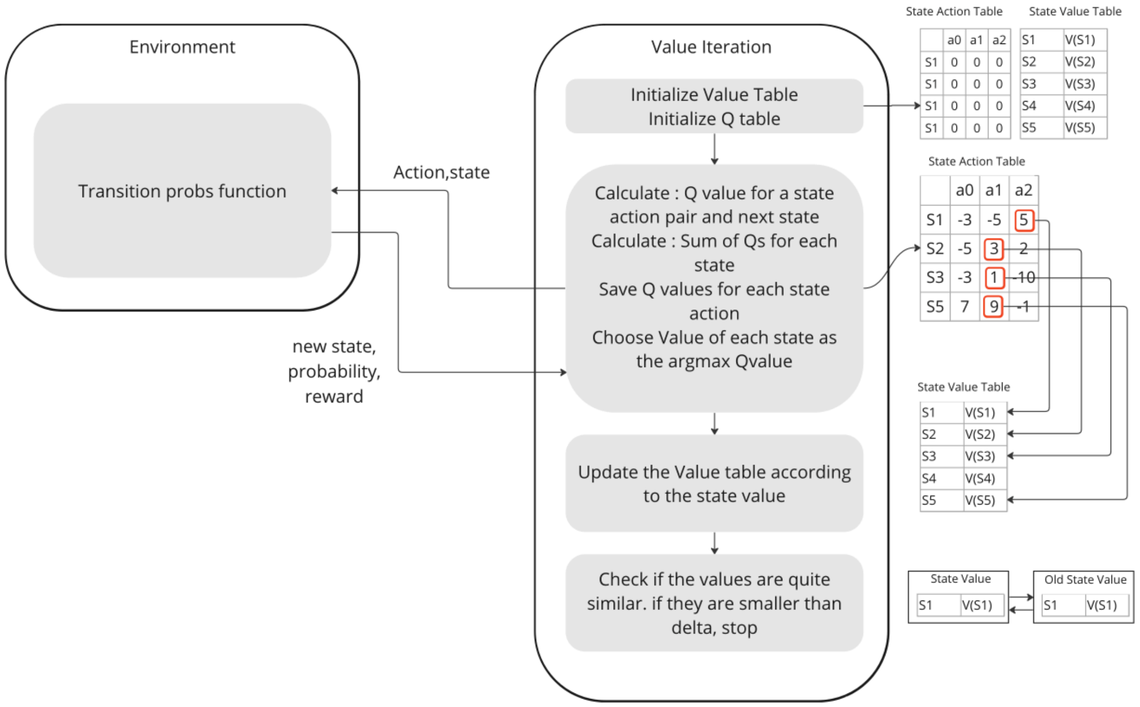


Figure 86 Workflow depicting the value iteration funtion and its relation to the environment

The second part of the process involved the actual implementation of the value iteration algorithm. In this script, the value iteration was set with a convergence delta of e-20 to ensure precise results. This script was responsible for iterating through all states and actions, updating the value function until convergence was achieved, thus determining the optimal policy.

The third part consisted of running a series of episodes to test the efficacy of the value iteration-derived policy. This script was designed to execute 1000000 episodes, where the performance of the optimal policy was benchmarked against the do-nothing policy. By comparing the mean return of the optimal policy with that of the do-nothing policy, it was possible to evaluate the effectiveness of the value iteration approach in addressing the problem.

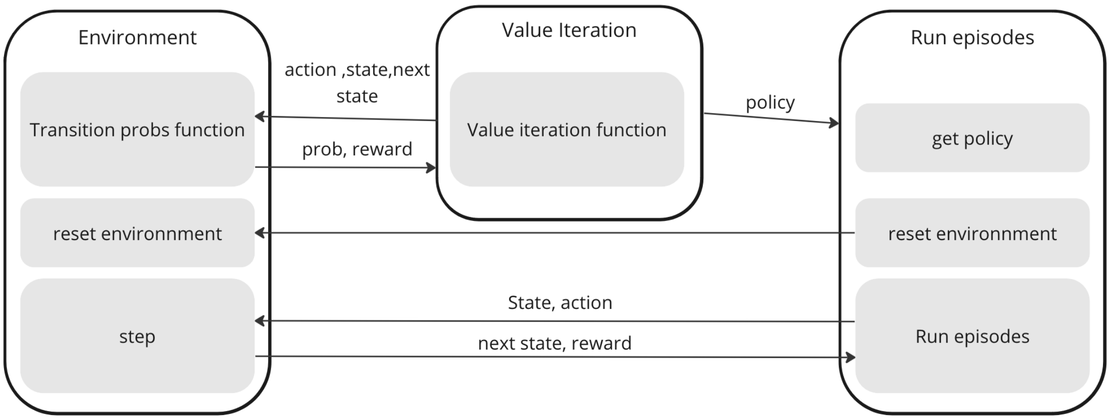


Figure 87 Workflow depicting the relationship between the scripts used for the optimization process (own work)

This three-part methodology ensured a systematic and comprehensive approach to implementing and validating the value iteration technique. By separating the environment creation, value iteration code, and episode simulations into distinct scripts, clarity and modularity were maintained, facilitating easier debugging and refinement of each component.

15.1.4.10. RESULTS

The value iteration algorithm was executed over 700 iterations. The building's lifespan was assumed to span 60 years, with time steps of 5 years each. Initial observations suggested that the optimal policy matched the benchmark policy. However, upon conducting various tests with the prices, it became apparent that the results were not as expected. The problem was detected to lie with formulation of the problem as it will be analyzed below.

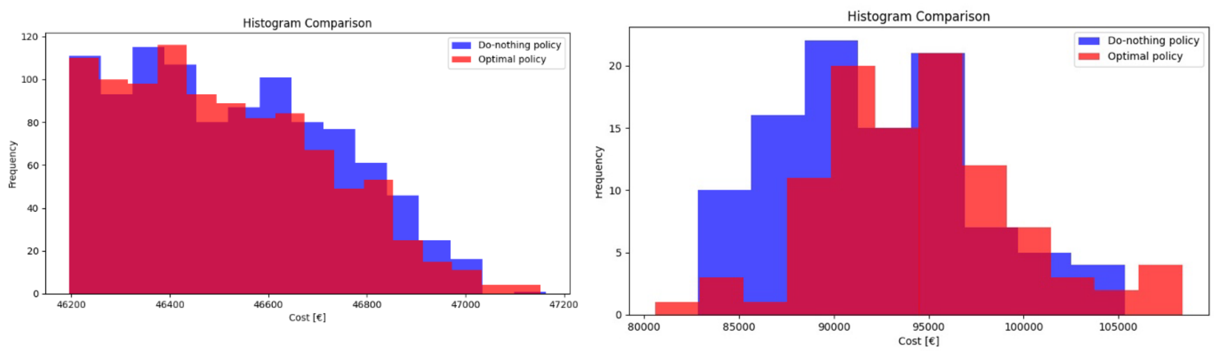


Figure 88 Histograms depicting the optimal policy and the best policy using different action costs . The red parts signify the costs accumulated in the different episodes by the optimal policy while the blue signify the do nothing policy.

15.1.4.11. CONCLUSIONS

Value iteration

The absence of time within the state space was identified during problem formulation, leading to the emergence of the infinite horizon problem. This issue stemmed from overlooking the temporal dimension of the problem, resulting in unrealistic outcomes.

It became evident that the exclusion of time from the state space led to flawed predictions and unreliable results. Despite initial attempts to model the problem without temporal considerations, the outcomes did not align with reality. This discrepancy was attributed to the model's inability to capture the evolving nature of the problem over time, resulting in unrealistic mean returns. The infinite horizon problem exacerbated this situation by assuming a consistent framework indefinitely, impeding the model's adaptability to changing circumstances.

To address this issue, it was crucial to integrate time into the state space, as suggested by (Morato et al., n.d.) to account for the dynamic nature of the problem. By introducing time as a variable, each state could be evaluated within its temporal context, providing a more accurate representation of the problem. Additionally, adjusting transition probabilities to reflect material states' evolution over time was necessary to ensure the model's validity. Furthermore, refining the modeling approach to emphasize temporal dynamics and fine-tuning transition matrices were essential steps in mitigating the infinite horizon problem and enhancing the model's predictive accuracy. This is particularly significant as the problem was framed with a finite horizon (60 years).

Energy modeling

The underperformance of the building model created using Geomeppy was mostly attributed to the lack of comprehensive documentation regarding its modeling capabilities. This led to a simplistic representation of the building geometry, resulting in unrealistic simulations. Moreover, the absence of consistent documentation made it challenging to incorporate crucial details such as infiltration rates, heating system specifications, further diminishing the accuracy of the simulations. Geomeppy, akin to Eppy scripting language, operates as a parametric tool, necessitating specific parameters and data connections for generating desired results. For instance, integrating a boiler into the heating system might require additional output files. While resources like the documentation provided by Big Ladder Software[23] offer extensive information, the absence of a step-by-step approach can be daunting, particularly for new users. Despite the fact that it can be usable by experienced EnergyPlus users, the current lack of documentation and community support poses significant challenges to its widespread adoption and further development.

[23] (https://bigladdersoftware.com/epx/docs/22-1/essentials/essentials.html)

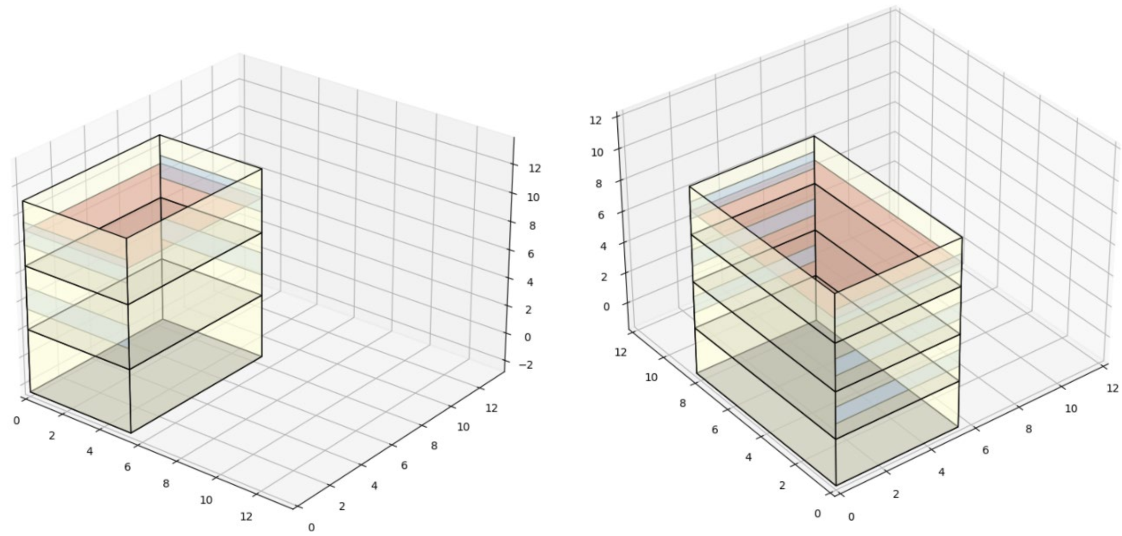


Figure 89 Examples of modelling problems with geomeppy. The house models when underground storey is incorporated resulting to the roof reaching the middle of the upper floor.

As a result of these shortcomings, the initial version of the problem formulation had to be revised. New transition matrices incorporating updated probabilities needed to be generated, and a more accurate building model had to be employed. Furthermore, the minimal impact of facade insulation changes on the terraced house's performance emphasized the need for nuanced adjustments in the simulation approach.

15.1.5. Results from original state space

In the first test, the Reward function was used without a 'penalty', in the sense of restricting a certain state to be reached. This was done to determine the policy over a 60-year period, as there was speculation that energy bills might eventually become more expensive than retrofitting the house.

The value iteration ran for more than 96 hours, completing 13 iterations across all possible scenarios. The results showed the optimal policy matched the "do nothing" policy, where no action is taken to change anything. Based on this policy and various episode simulations, the expected costs over the next 60 years (excluding the final year) were analyzed.

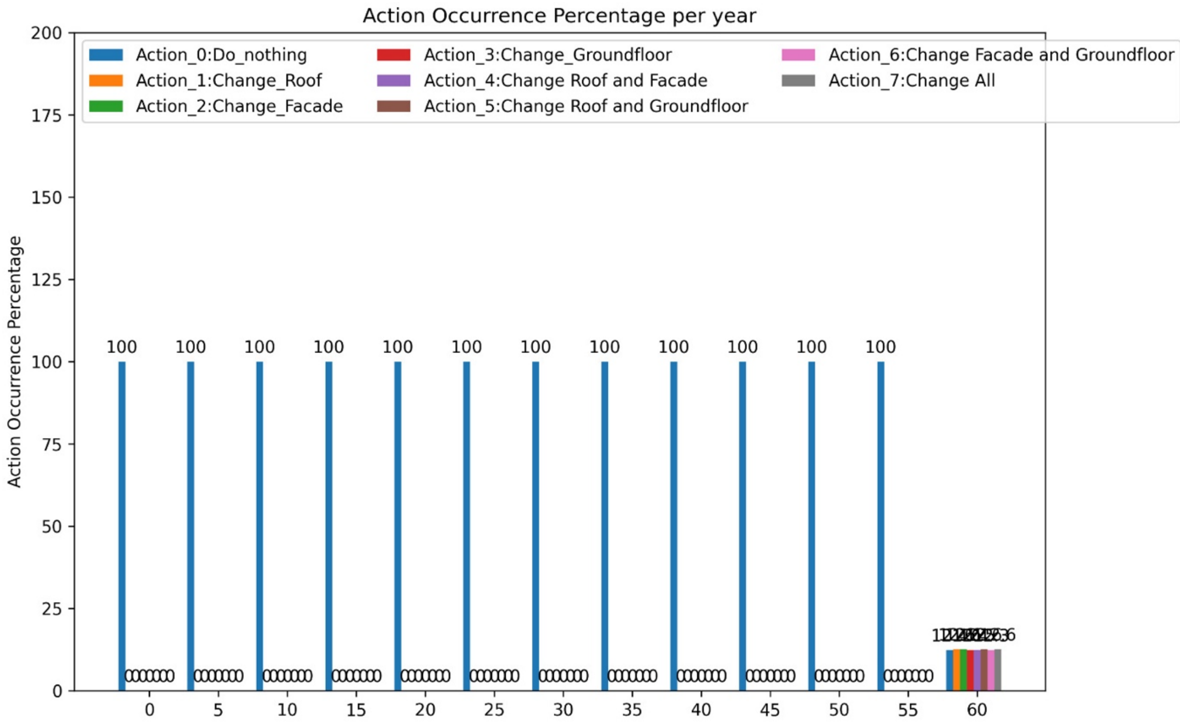


Figure 90 Plot of the optimal policy for the problem formulation without penalty. Year 60 shows all actions being considered. This is happening because the rewards at year 60 are zero. These are considered absorbing states , meaning that once reached there , there is nowhere else you might go.

The "do nothing" policy yielded a mean expected cost of 854,242 euros over 60 years, which was very close to the optimal policy's return of 854,240 euros. The variance between the different return outcomes across the episodes amounted to 4,420 euros. This variance and the minor differences in costs are expected due to the randomness involved in transitioning from state to state. The use of 1,000,000 episodes ensured that the variances between the policies would not differ significantly.

In the following schemes, the plots of the "do nothing" policy are shown. The initial state starts with a reward representing the first year's energy cost, amounting to 14,501 euros based on an energy requirement of 41,500 kWh for the house model. The house area is 250 m², and the total energy demand is multiplied by the price per kWh of 0.35 euros. In the next time step there is a sudden increase in costs, where the rewards start accounting for the five-year lapse between states. In the final time step, the rewards return to zero as that state signifies the end of the episode. The building does not reach the final state of degradation but remains in a state of Roof degradation 40%, Façade 20%, and Groundfloor 40% from year 20, as shown in the right diagram representing kWh per m² per state.

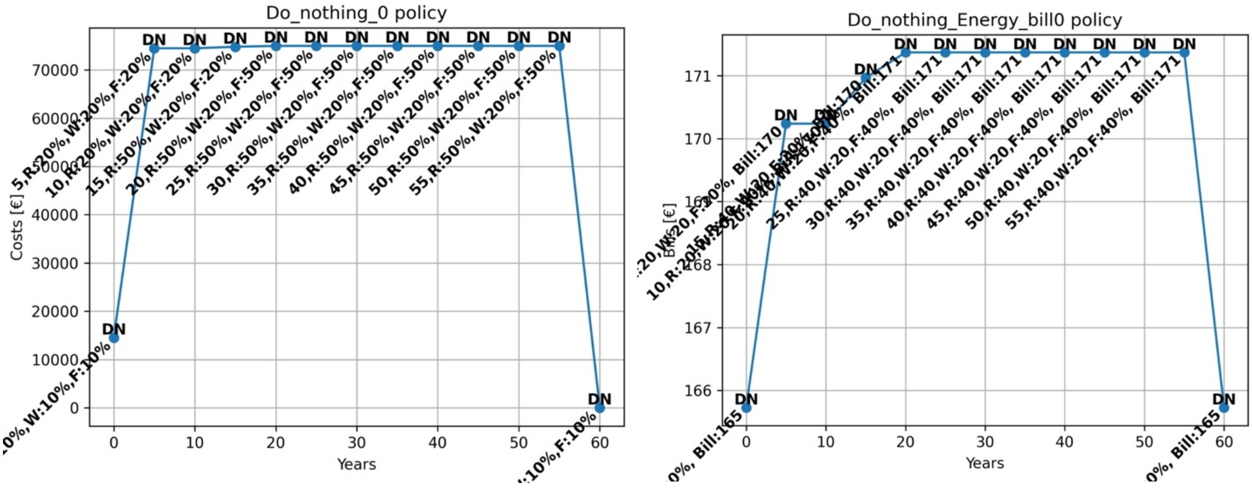


Figure 91 Episode samples depicting the costs of each time step (left) and the energy demand (right) under do nothing policy (own work)

Similarly, the simulated episode for the optimal policy shows the house degrading over time and reaching the worst degradation stage by year 40.

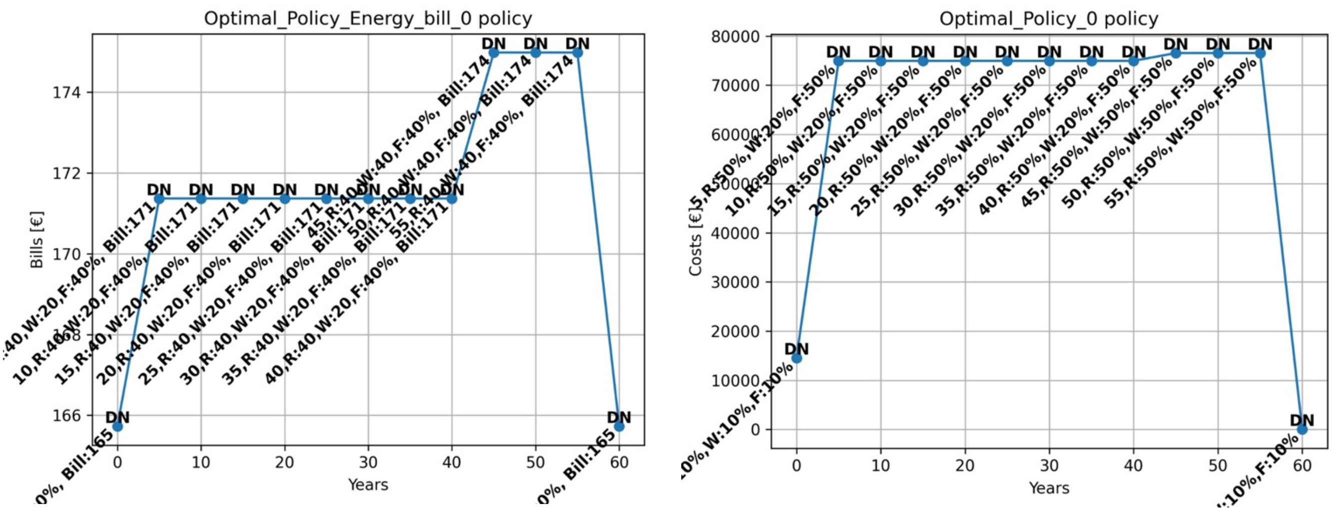


Figure 92 Episode samples of optimal policy (same with do nothing policy) depicting the costs (left) and the energy demand (right) in each time step

The histogram comparison of the total return per episode showed that in most scenarios, the total return amounted to 855,000 euros, indicating that the building would likely reach the worst case of degradation quite early.

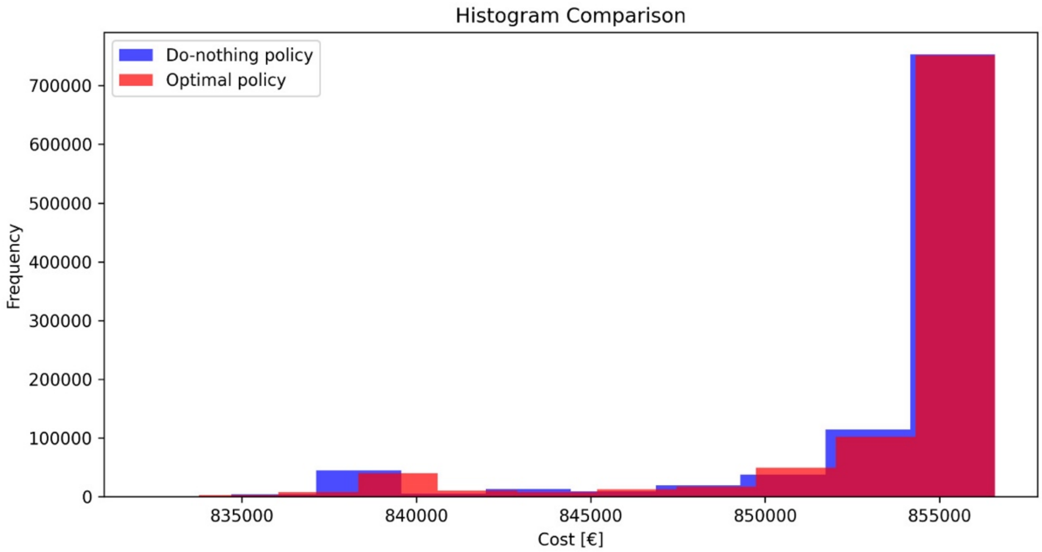


Figure 93 Return comparison between do nothing policy and optimal policy (own work)

The variances between the mean return of the optimal policy (4424) and the mean return of the do nothing policy(4420) were quite similar thanks to the large amount of generated episodes with their difference being only an integer of 4 which can be considered acceptable difference in this case.

```
212000
214000
216000
218000
220000
222000
Value_iteration_finished:
16908.97767853737
13
Number of iterations for optimal policy: 13
The mean return of optimal policy is: 854240.772362554. The variance is:4424.839191998787
The mean return of zero policy is: 854242.7703424009. The variance is:4420.636979023655
[0 0 0 ... 4 6 3]
```

Figure 94 Snippet of mean return over one million episodes for both policies

Based on the results and analysis, several key conclusions can be drawn:

First of all, the fact that the optimal policy matched the "do nothing" policy suggests that, under the current model and assumptions, no retrofitting actions provide a significant advantage. This implies that the costs of potential interventions are not justified by the savings in energy bills over the 60-year period considered (something that was anticipated since the difference in prices between the best and worst case scenario amounted to only 5.6% increase in energy demand).

However, it must be noted that the close match between the return indicates that the model is stable and consistent in predicting long-term costs.

Even more, the uncertainty emphasized both in the variance outcomes and the different degradation scenarios that the episodes simulated showed that the model that was formulated was reliable. This can be assumed based on the histogram comparison and the graphs simulating the energy demand per time step , both of which point that the building is likely to experience early degradation to its worst state, driving up costs quickly.

The value iteration algorithm effectively converged after 13 iterations, despite the high computational demand requiring over 96 hours, indicating its reliability in optimizing expected cumulative rewards over a 60-year period. However, the significant computational resources required suggest potential scalability issues for larger models. For future optimization efforts, incorporating penalties and incentives within the reward structure could provide a more nuanced evaluation of policies, particularly for retrofitting or other proactive measures. Given the possibility of future policies mandating buildings to maintain certain energy performance standards, such as the label C requirement for office buildings in the Netherlands[24], repeating the experiment with an introduced penalty was essential to simulate these scenarios accurately and inform long-term decision-making effectively.

[24] (https://natlawreview.com/article/energy-label-c-obligation-all-office-buildings-netherlands-2023-few-exceptions).

15.2. Tables

Table 11: States example							
State number	Time (years)	Roof Degradation (%)	Façade Degradation (%)	Ground floor degradation (%)	Age Roof Insulation (years)	Age Façade Insulation (years)	Age Ground floor Insulation (years)
1	0	0	0	0	0	0	0
2	0	0	0	1	0	0	0
3	0	0	0	2	0	0	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
27	5	0	0	0	0	0	0
28	5	0	0	0	0	0	5
29	5	0	0	0	0	5	0
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
78304	45	2	1	2	30	10	45
78305	45	2	1	2	30	15	0
78306	45	2	1	2	30	15	5
. . .	. . .	. . .	. . .	. . .	. . .	. . .	. . .
223584	60	2	2	2	60	60	50
223585	60	2	2	2	60	60	55
223586	60	2	2	2	60	60	60

Table 7: Degradation scenarios and their costs in energy bills of building simulated in Geomeppy, Stage 1				
state	description	energy[kWh/m2]	energy[kWh]	Energy bills
1	(0, 0, 0)	183,3	24500	8575,1
2	(0, 0, 0.2)	185,1	24734	8656,8
3	(0, 0,0.4)	186,7	24956	8734,6
4	(0,0.2, 0)	183,6	24536	8587,5
5	(0, 0.2, 0.2)	185,3	24769	8669,2
6	(0, 0.2, 0.4)	187	24991	8746,9
7	(0, 0.4, 0)	183,9	24573	86006
8	(0, 0.4, 0.2)	185,6	24806	8682,2
9	(0, 0.4, 0.4)	187,3	25027	8759,6
10	(0.4, 0, 0)	184	24586	8604,9
11	(0.2, 0,0.2)	185,7	24818	8686,3
12	(0.2, ,0 , 0.4)	187,3	25039	8763,7
13	(0.2,0.2, 0)	184,2	24621	8617,4
14	(0.2, 0.2, 0.2)	186	24853	8698,6
15	(0.2, 0.2, 0.4)	187,6	25074	8775,9
16	(0.2, 0.4, 0)	184,5	24658	8630,3
17	(0.2, 0.4, 0.2)	186,2	24890	8711,5
18	(0.2, 0.4, 0.4)	187,9	25110	8788,6
19	(0.4, 0, 0)	184,6	24669	8634,1
20	(0.4, 0, 0.2)	186,3	24900	8715,1
21	(0.4, 0, 0.4)	188	25120	8792,1
22	(0.4, 0.2, 0)	184,8	24704	8646,3
23	(0.4, 0.2, 0.2)	186,6	24935	8727,3
24	(0.4, 0.2, 0.4)	188,2	25155	8804,1
25	(0.4, 0.4, 0)	185,1	24741	8659,3
26	(0.4, 0.4, 0.2)	186,8	24971	8740

27	(0.4, 0.4, 0.4)	188,5	25191	8816,7

Table 10: Degradation scenarios of model generated in stage 2 without infiltration and their costs in energy bills			
description	energy[kWh/m2]	energy[kWh]	Energy bills
(0, 0, 0)	165.72	41772.9	14620.5
(0, 0, 0.2)	166.13	41876.0	14656.6
(0, 0,0.4)	166.54	41978.3	14692.4
(0,0.2, 0)	169.17	42641.0	14924.4
(0, 0.2, 0.2)	169.57	42742.7	14960.0
(0, 0.2, 0.4)	169.97	42844.3	14995.5
(0, 0.4, 0)	172.80	43556.4	15244.8
(0, 0.4, 0.2)	173.19	43656.6	15279.8
(0, 0.4, 0.4)	173.59	43756.9	15314.9
(0.4, 0, 0)	166.39	41941.6	14679.6
(0.2, 0,0.2)	166.80	42044.7	14715.7
(0.2, ,0 , 0.4)	167.21	42147.0	14751.5
(0.2,0.2, 0)	169.83	42809.0	14983.2
(0.2, 0.2, 0.2)	170.23	42909.9	15018.5
(0.2, 0.2, 0.4)	170.64	43011.6	15054.1
(0.2, 0.4, 0)	173.46	43723.0	15303.1
(0.2, 0.4, 0.2)	173.86	43823.2	15338.1
(0.2, 0.4, 0.4)	174.25	43923.4	15373.2
(0.4, 0, 0)	167.12	42126.5	14744.3
(0.4, 0, 0.2)	167.53	42228.9	14780.1
(0.4, 0, 0.4)	167.94	42331.2	14815.9
(0.4, 0.2, 0)	170.56	42992.5	15047.4
(0.4, 0.2, 0.2)	170.96	43093.4	15082.7

(0.4, 0.2, 0.4)	171.36	43195.1	15118.3
(0.4, 0.4, 0)	174.18	43905.8	15367.0
(0.4, 0.4, 0.2)	174.58	44006.0	15402.1
(0.4, 0.4, 0.4)	174.98	44105.5	15436.9

All possible measures involving plastic foam insulation

Roof insulation						
Name	Info	Placement	Width (mm)	Rd value	RC value	Price euros per m2
WB373 – Bio EPS		Flat roof	160 mm	5.0	5.2	295.67
WB374 – Bio EPS		Flat roof	200	6.3	6.5	303.35
WB375 – Bio EPS		Flat roof	250	7.9	8.1	305.82
WB006a – EPS isolatie		Flat roof	100	-	3.2	231.63
WB007a – EPS isolatie		Flat roof	100	-	3.1	287.04
WB004 – PIR renovatie dakplaten		Pitched roof	81	3.6	3.9	132.49
WB212a– PIR renovatie dakplaten		Pitched roof	142	6.45	6.8	145.06
WB212b – PIR renovatie dakplaten		Pitched roof	175	6.45	8.3	149.96
Ground floor insulation						
WB003b -EPS		Crawl space floor	300	-	2.6	34.68
WB002d - EPS		Crawl space ceiling	100	2.8	2.9	28.91
Façade insulation						

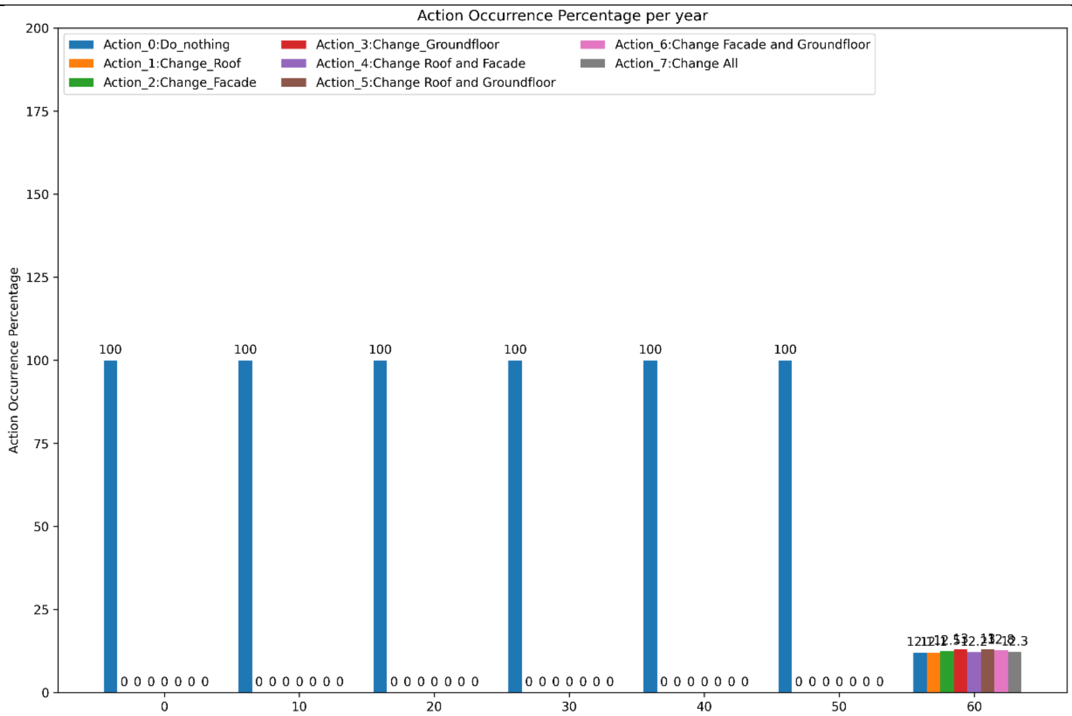
WB009c – EPS beads		Cavity wall	50	-	1.6	21.96
WB008b -EPS isolation	Decorative plaster finishing	Exterior wall	100	-	2.6	162.73
WB224 – EPS isolation	Decorative plaster finishing	Exterior wall	220	-	6.3	175.25
WB270 – EPS100	Façade insulation and walls	Exterior wall	120	3.5	3.9	210,17

Table 10: Degradation scenarios and their costs in energy bills			
description	energy[kWh/m2]	energy[kWh]	Energy bills
0%, 0%, 0%	165.73	41773.00	14.620,55
0%, 0%, 20%	181.06	45638.49	15.973,47
0%, 0%, 40%	196.48	49525.86	17.334,05
0%, 20%, 0%	184.08	46400.01	16.240,00
0%, 20%, 20%	199.49	50282.44	17.598,85
0%, 20%, 40%	214.96	54182.52	18.963,88
0%, 40%, 0%	202.68	51088.43	17.880,95
0%, 40%, 20%	218.14	54983.56	19.244,25
0%, 40%, 40%	233.64	58892.10	20.612,24
20%, 0%, 0%	181.31	45700.60	15.995,21
20%, 0%, 20%	196.72	49585.85	17.355,05
20%, 0%, 40%	212.20	53487.33	18.720,57
20%, 20%, 0%	199.72	50342.43	17.619,85
20%, 20%, 20%	215.19	54240.39	18.984,14
20%, 20%, 40%	230.70	58151.05	20.352,87
20%, 40%, 0%	218.37	55041.43	19.264,50

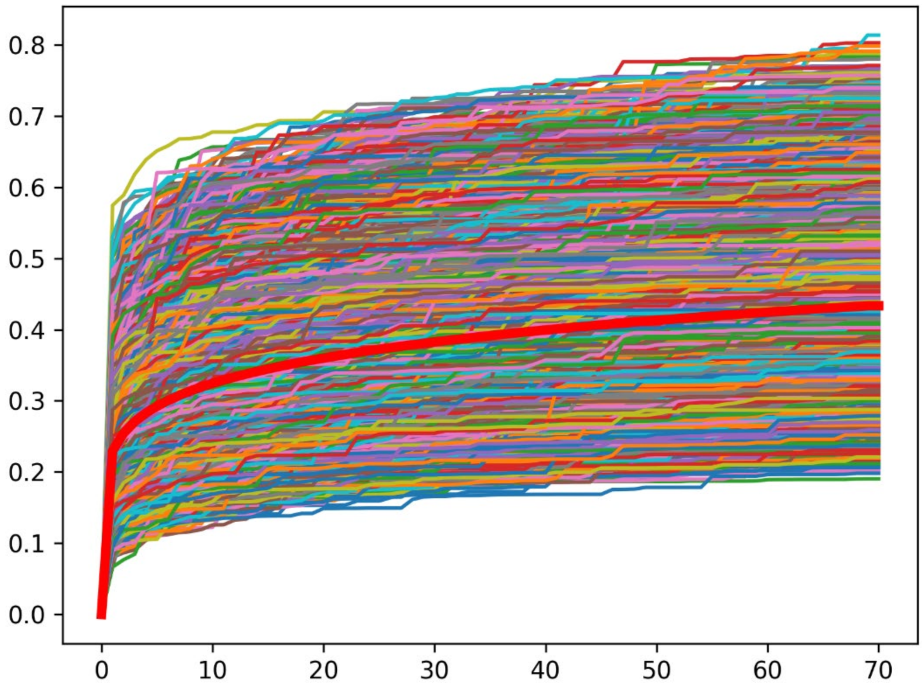
20%, 40%, 20%	233.86	58947.86	20.631,75
20%, 40%, 40%	249.41	62866.28	22.003,20
40%, 0%, 0%	197.02	49661.37	17.381,48
40%, 0%, 20%	212.49	53560.03	18.746,01
40%, 0%, 40%	228.02	57473.51	20.115,73
40%, 20%, 0%	215.48	54313.79	19.009,83
40%, 20%, 20%	230.99	58222.33	20.377,82
40%, 20%, 40%	246.54	62142.87	21.750,00
40%, 40%, 0%	234.15	59019.85	20.656,95
40%, 40%, 20%	249.68	62935.45	22.027,41
40%, 40%, 40%	265.26	66860.93	23.401,33

15.3. Plots and Diagrams

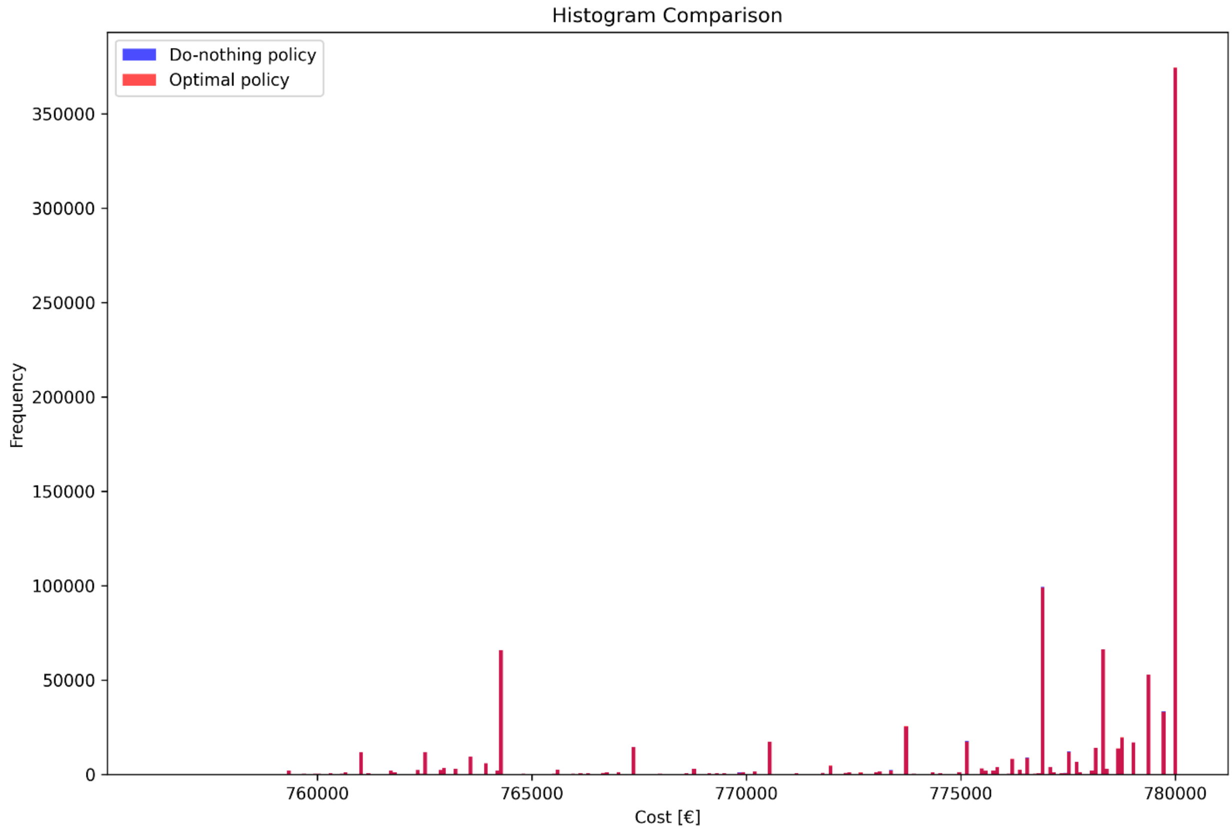
Simulation 1: No penalty, no infiltration simulation, beta 0.15
Optimal policy plot



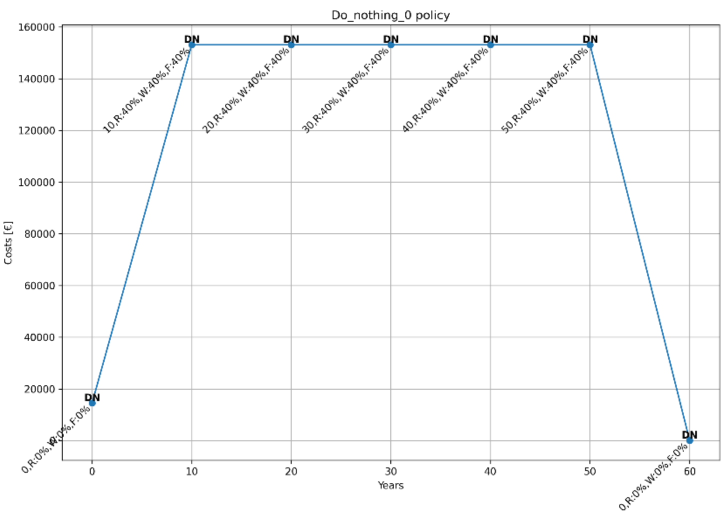
Material degradation scenarios , 1000000 realizations

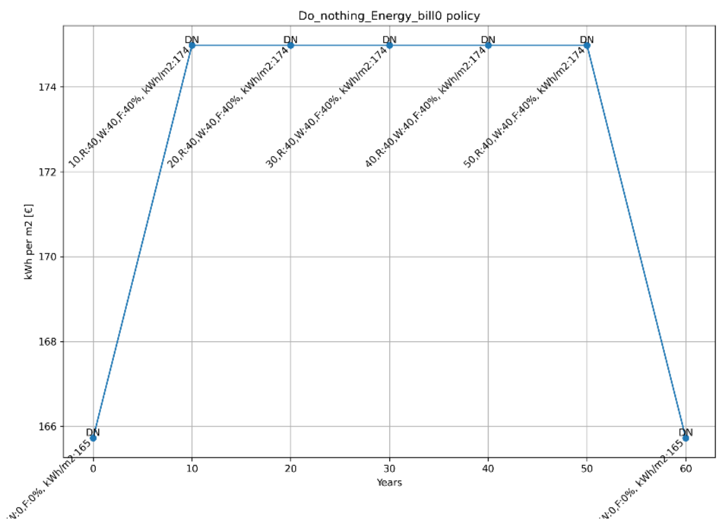


Histogram comparison of return between do nothing policy and optimal policy

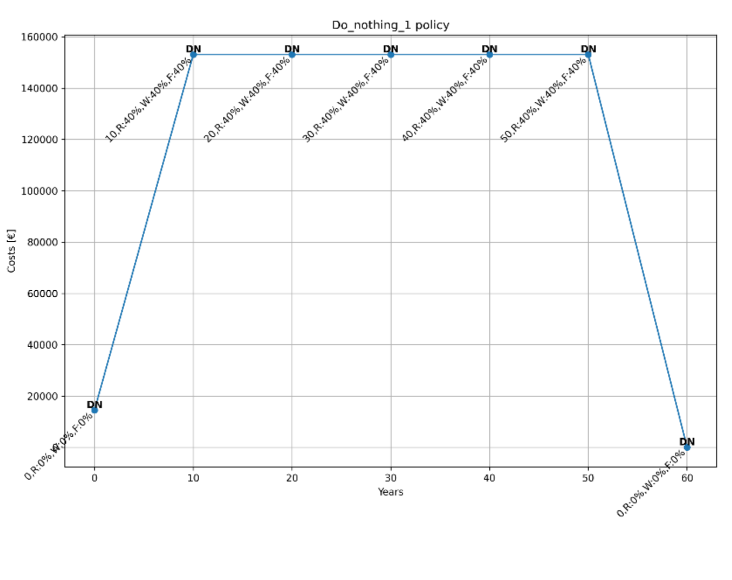


Do nothing policy: Sample Episode 1 Costs	Do nothing policy: Sample Episode 1 States of energy demand

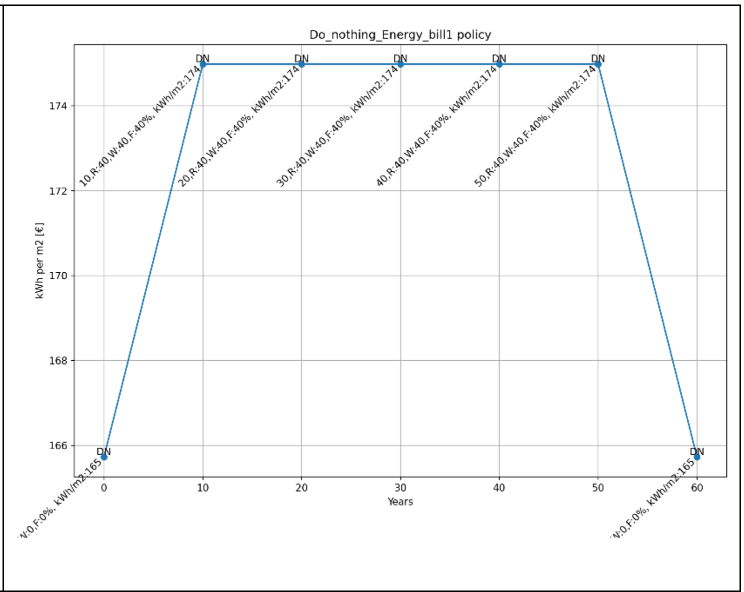




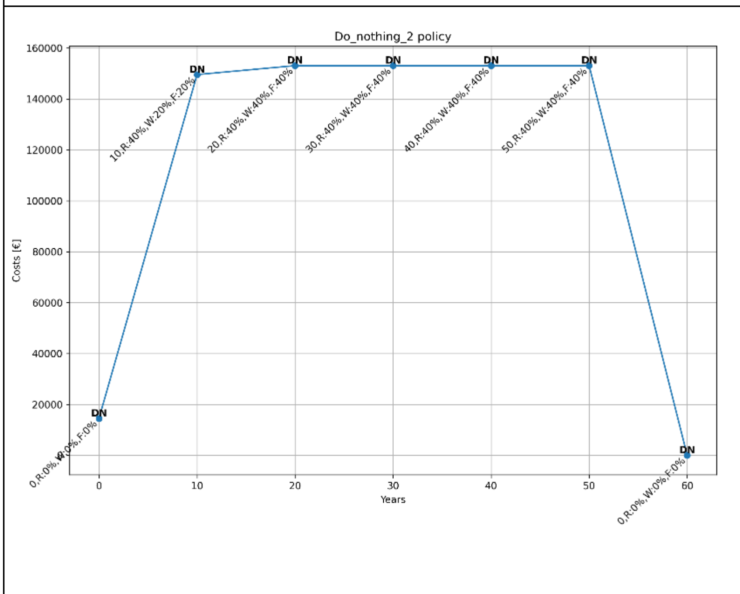
Do nothing policy: Sample Episode 2 Costs	Do nothing policy: Sample Episode 2 States of energy demand



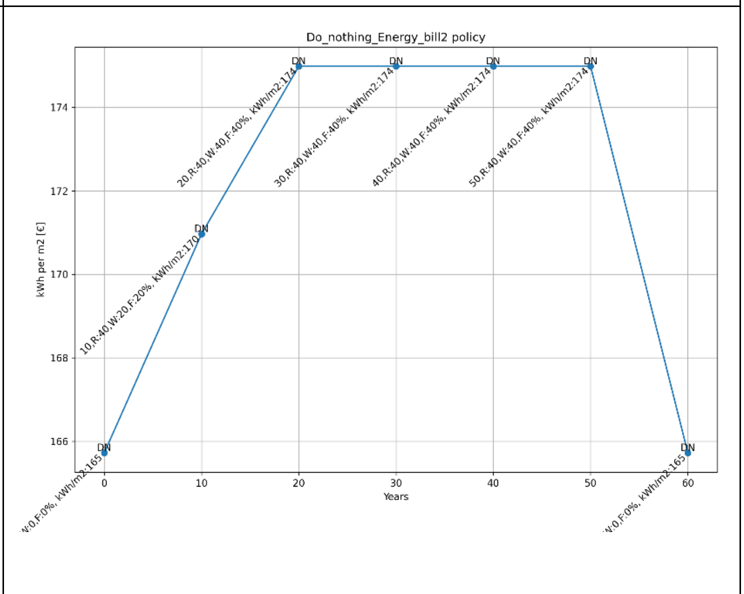
Do nothing policy: Sample Episode 3 Costs



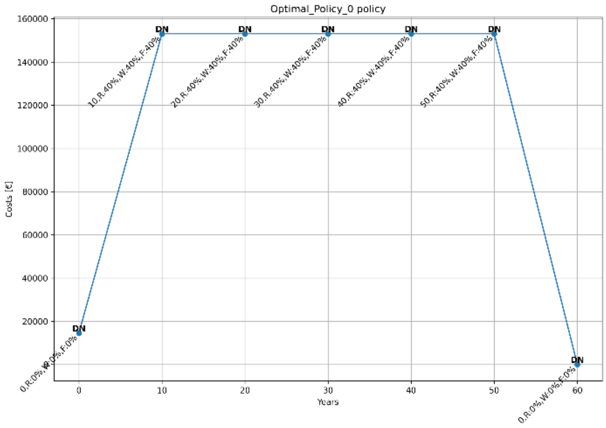
Do nothing policy: Sample Episode 3 States of energy demand



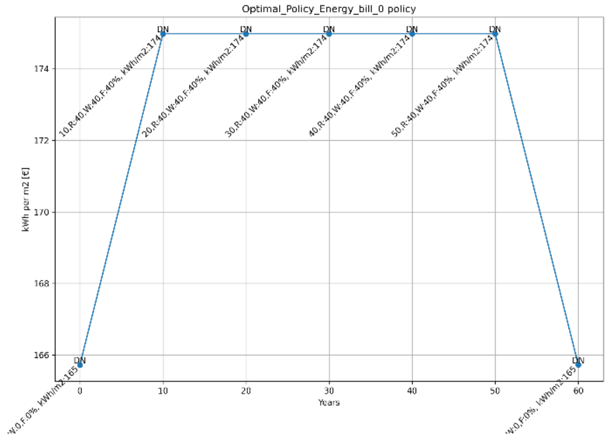
Optimal policy: Sample Episode 1 Costs



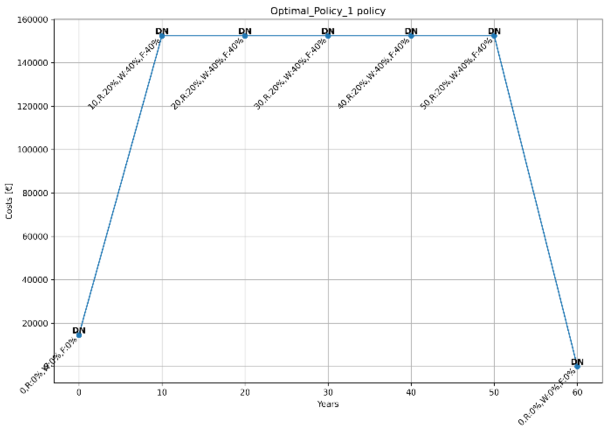
Optimal policy: Sample Episode 1 States of energy demand



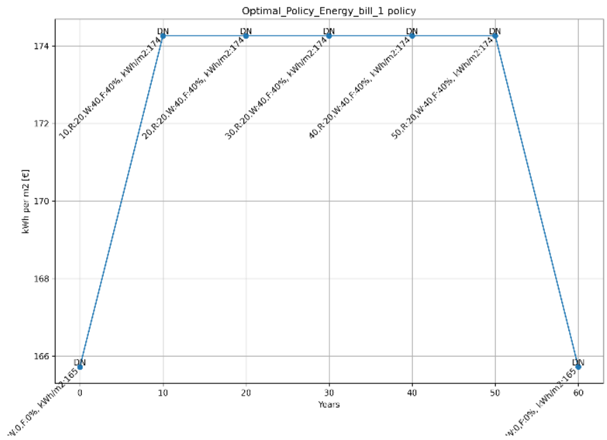
Optimal policy: Sample Episode 2 Costs



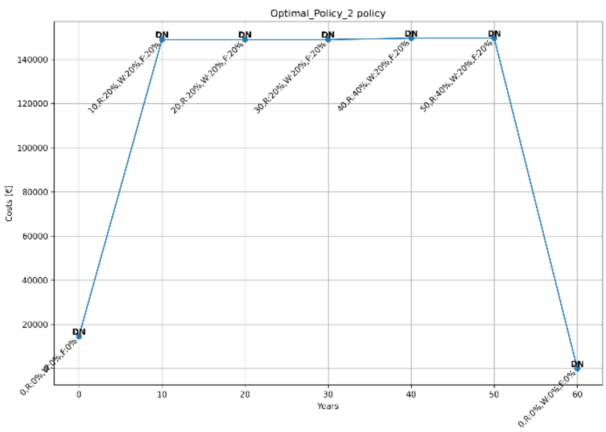
Optimal policy: Sample Episode 2 States of energy demand

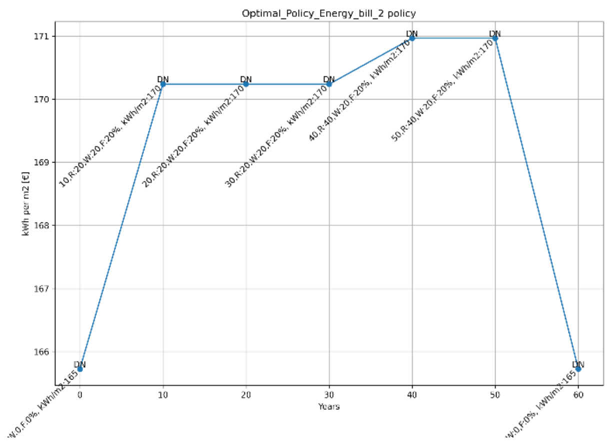


Optimal policy: Sample Episode 3 Costs

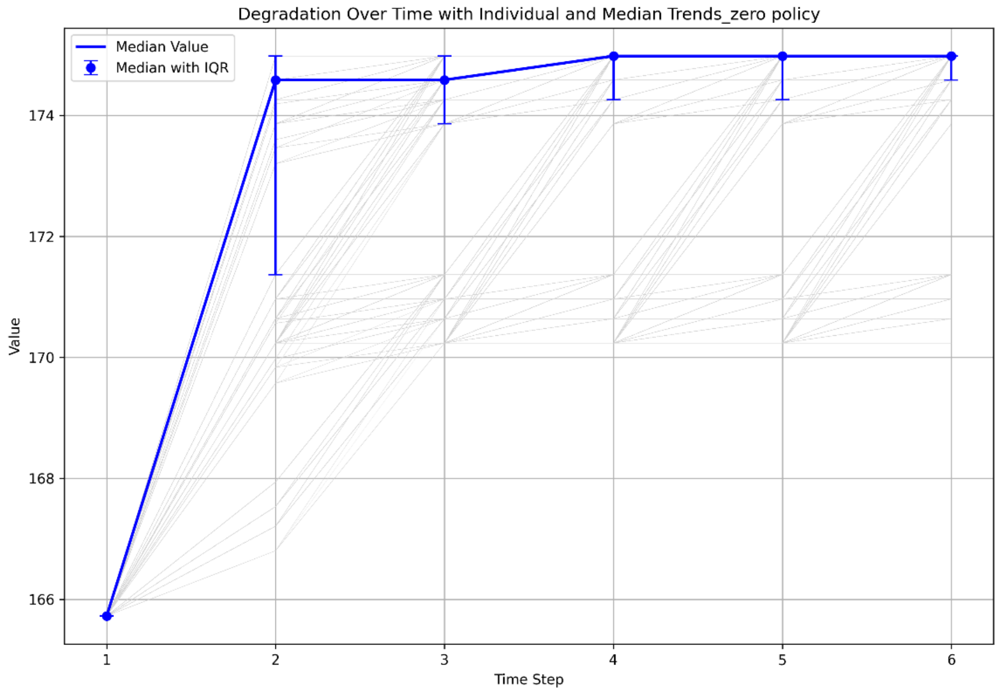


Optimal policy: Sample Episode 3 States of energy demand



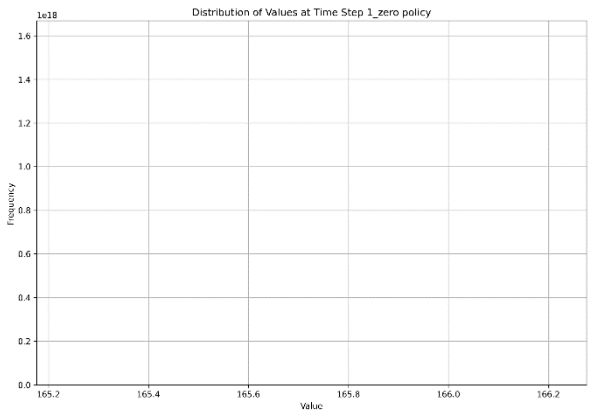


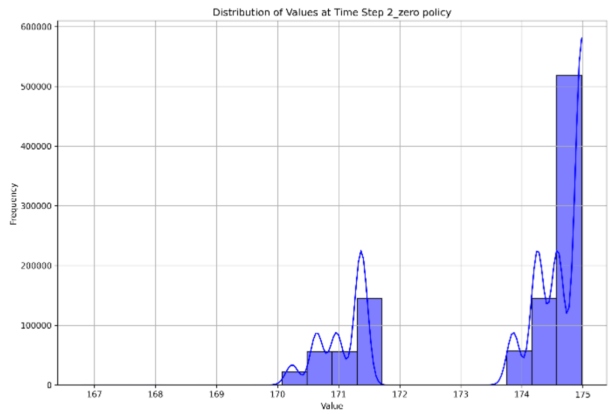
Median energy demand change with do nothing policy from one million episodes



Distribution of possible energy demand at time step 1 over million episodes with do nothing policy

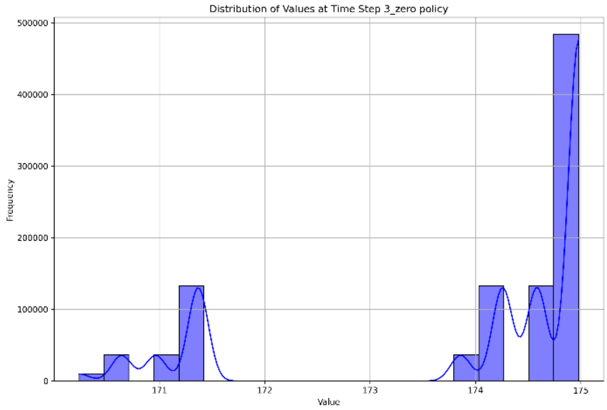
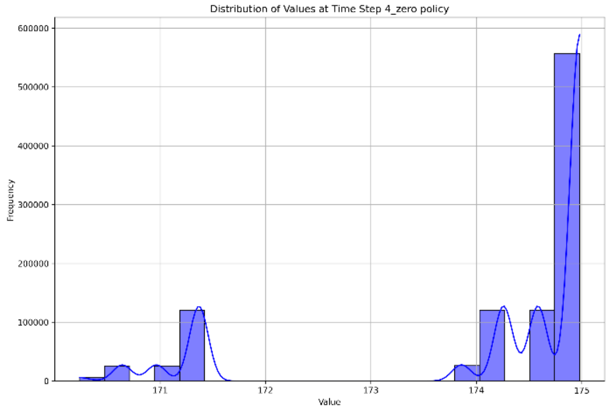
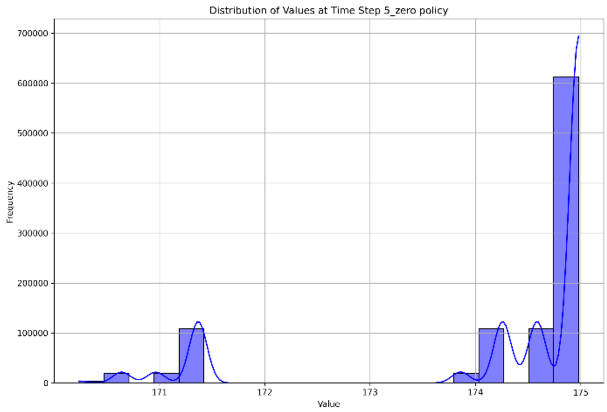
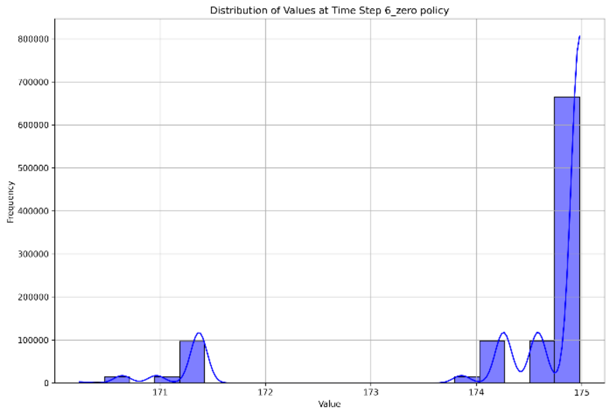
Distribution of possible energy demand at time step 2 over million episodes with do nothing policy

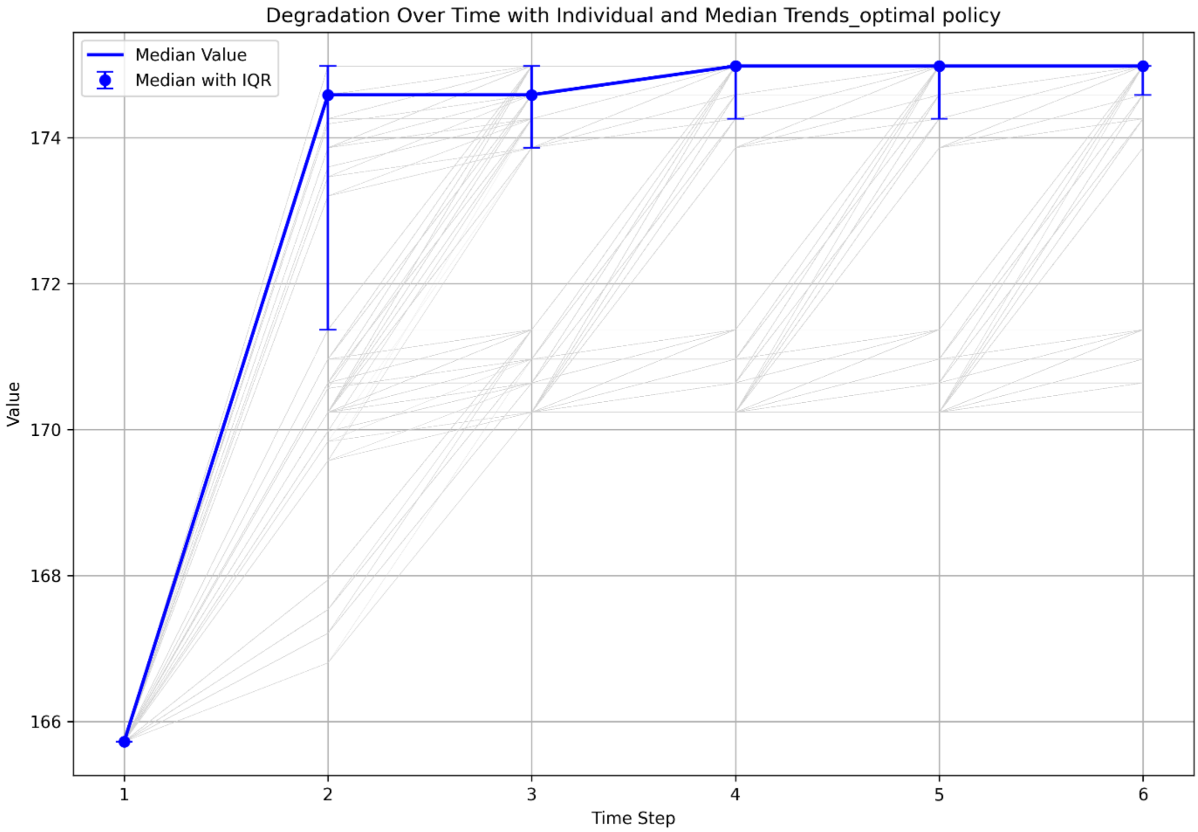


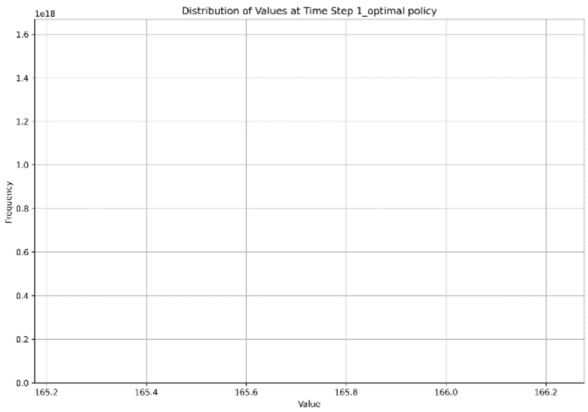
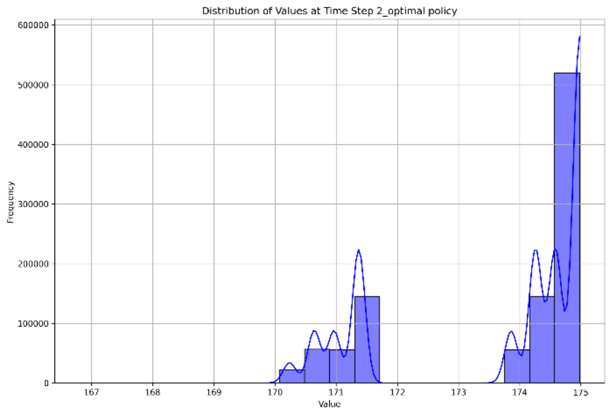


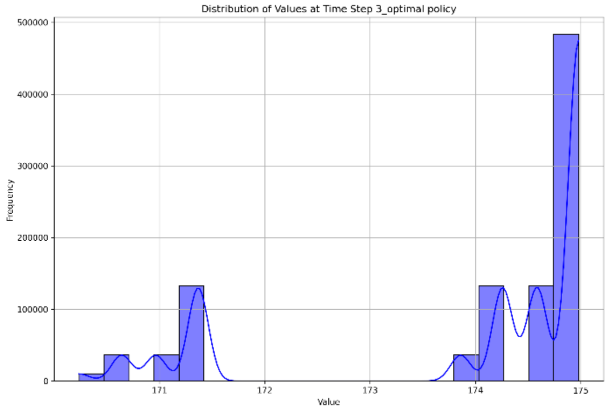
Distribution of possible energy demand at time step 3 over million episodes with do nothing policy

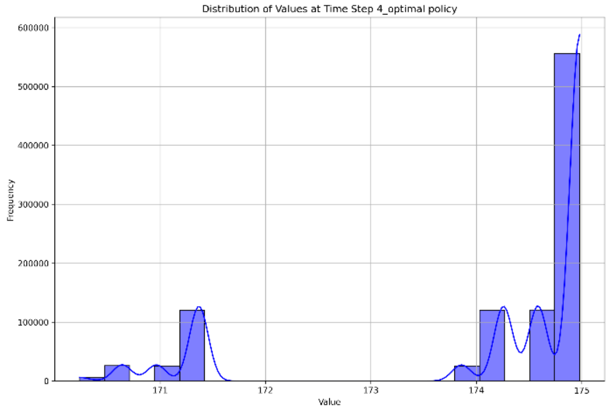
Distribution of possible energy demand at time step 4 over million episodes with do nothing policy

<table>
<tr>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>Distribution of possible energy demand at time step 5 over million episodes with do nothing policy</td>
<td>Distribution of possible energy demand at time step 6 over million episodes with do nothing policy</td>
</tr>
<tr>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td colspan="2">Median energy demand change with optimal policy from one million episodes</td>
</tr>
</table>



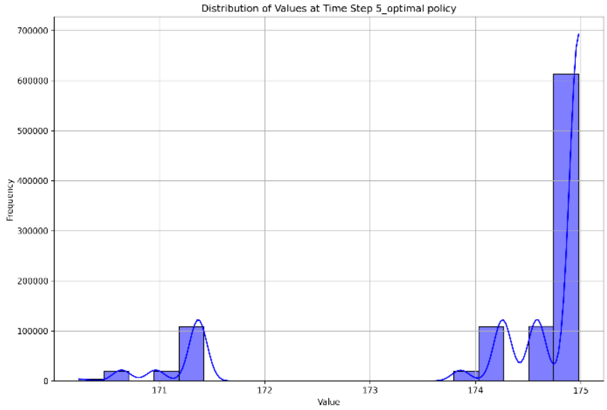
Distribution of possible energy demand at time step 1 over million episodes with optimal policy	Distribution of possible energy demand at time step 2 over million episodes with optimal policy
	
Distribution of possible energy demand at time step 3 over million episodes with optimal policy	Distribution of possible energy demand at time step 4 over million episodes with optimal policy

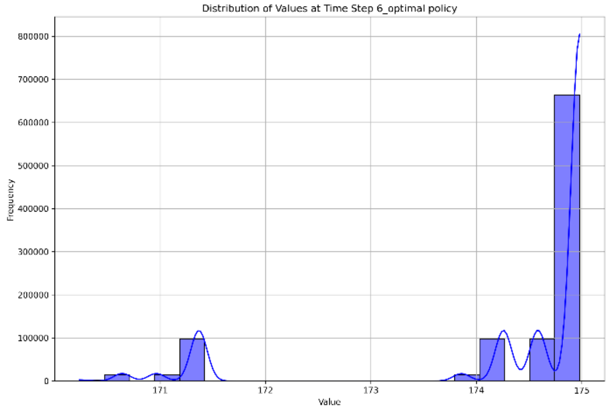




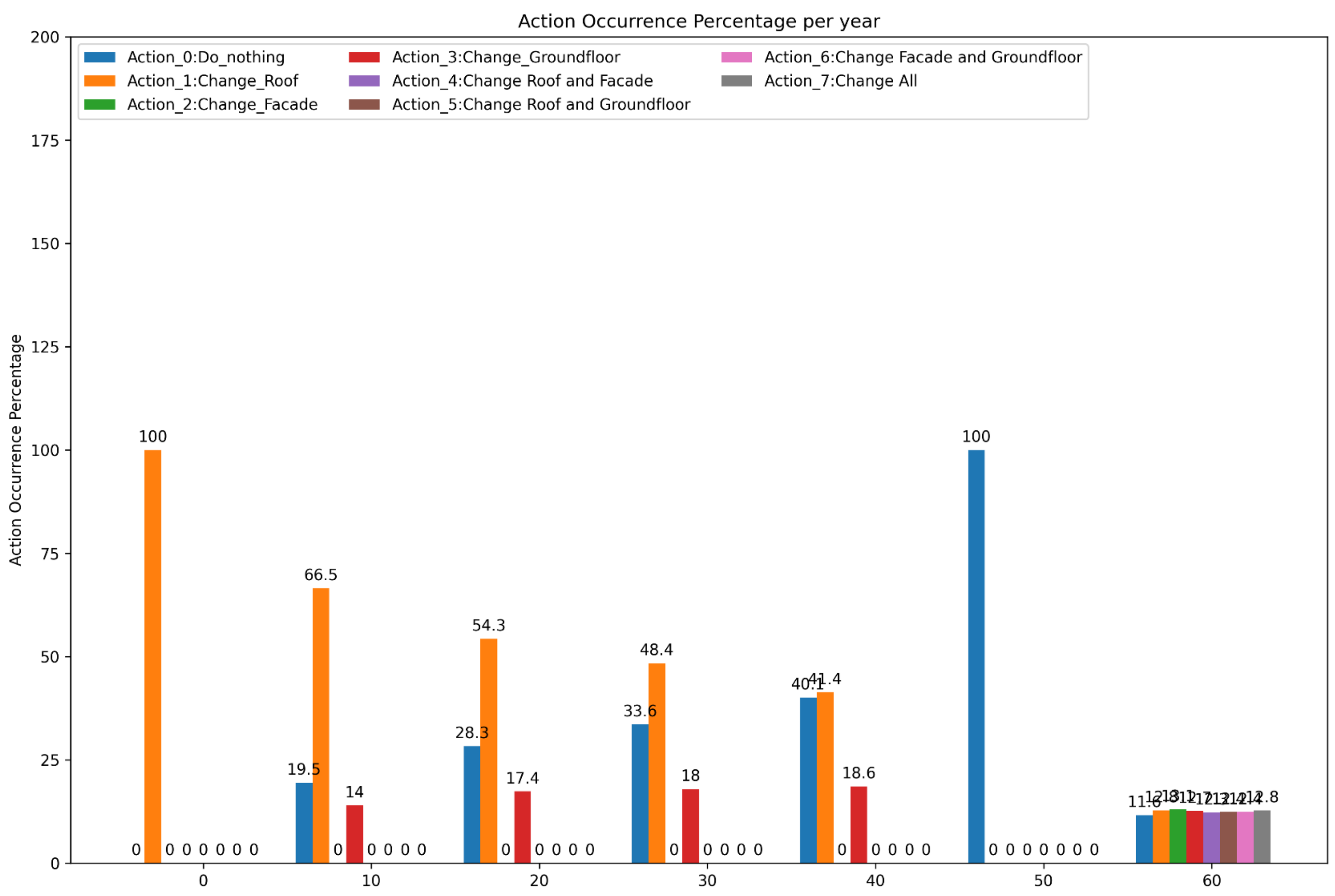
Distribution of possible energy demand at time step 5 over million episodes with optimal policy

Distribution of possible energy demand at time step 6 over million episodes with optimal policy

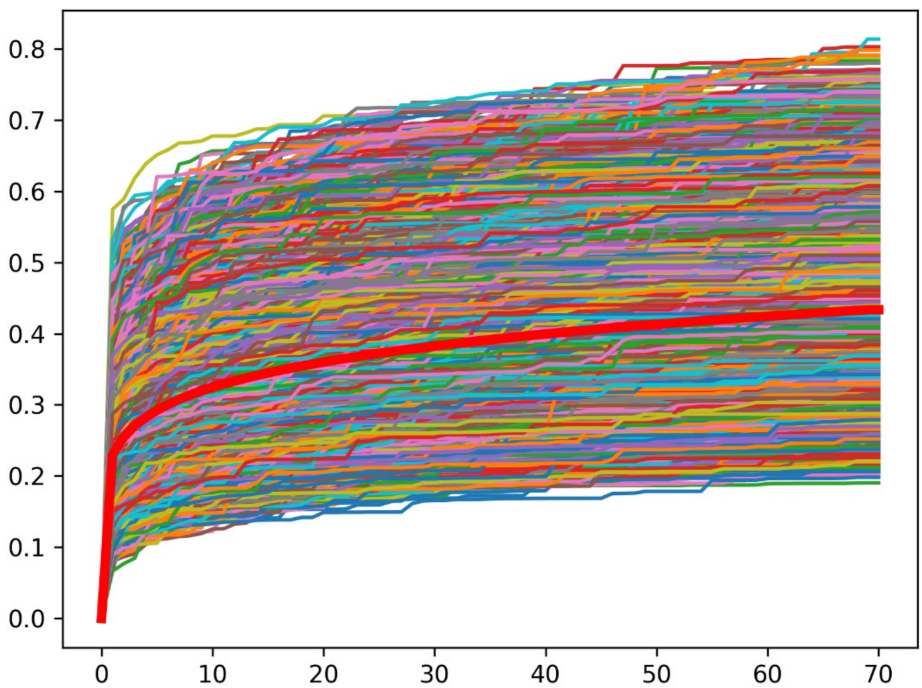




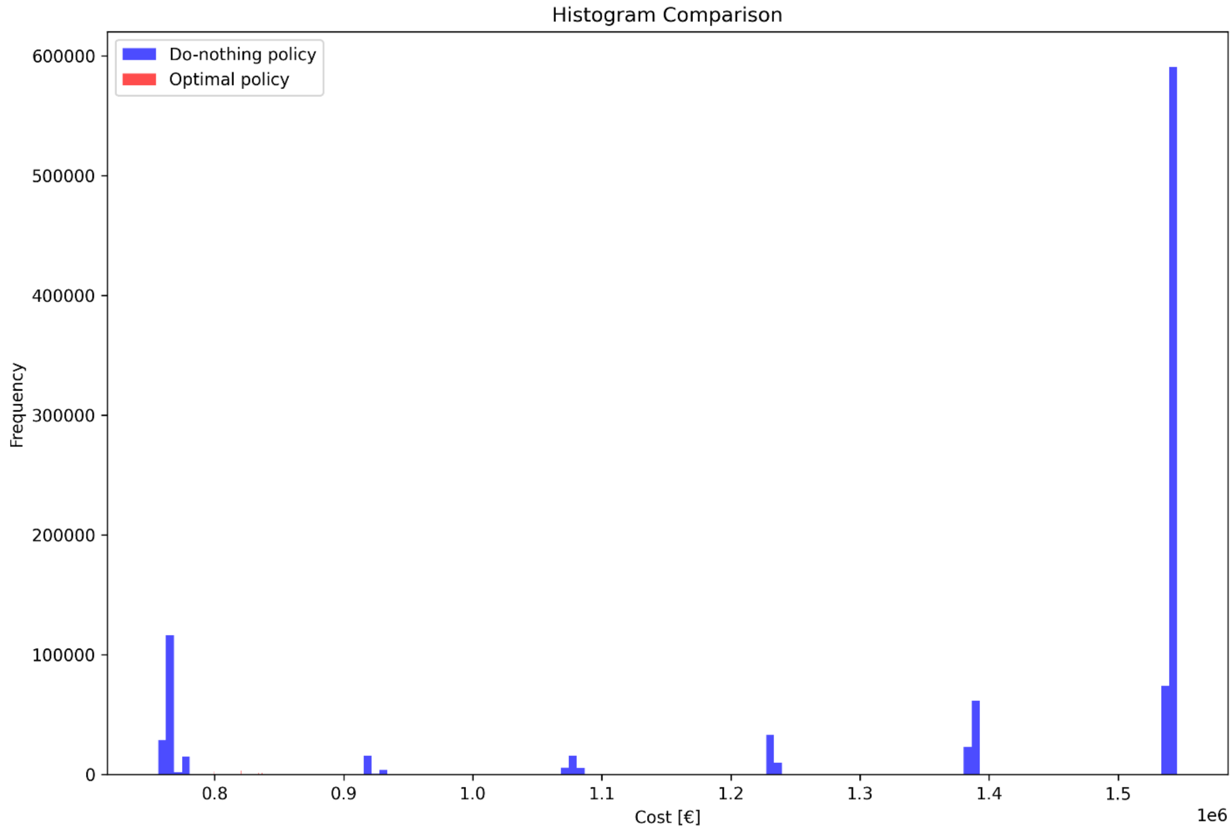
Simulation 2: Penalty, no infiltration simulation, beta 0.15
Optimal policy plot



Material degradation scenarios , 1000000 realizations

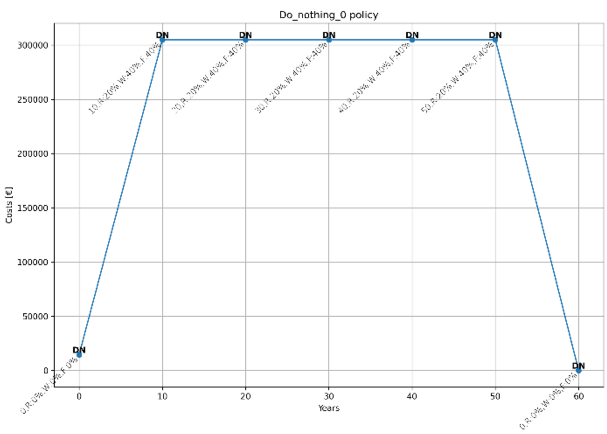


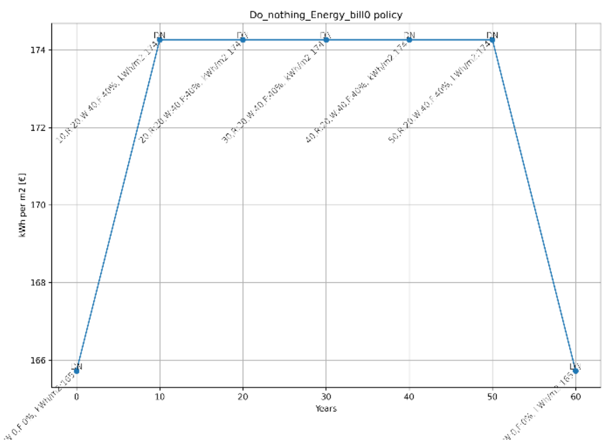
Histogram comparison of return between do nothing policy and optimal policy



Do nothing policy: Sample Episode 1 Costs

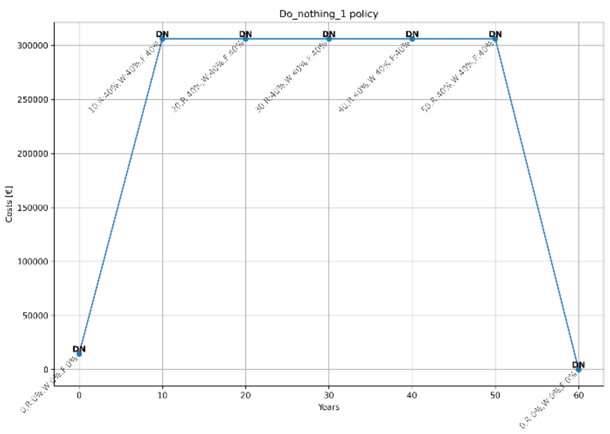
Do nothing policy: Sample Episode 1 States of energy demand



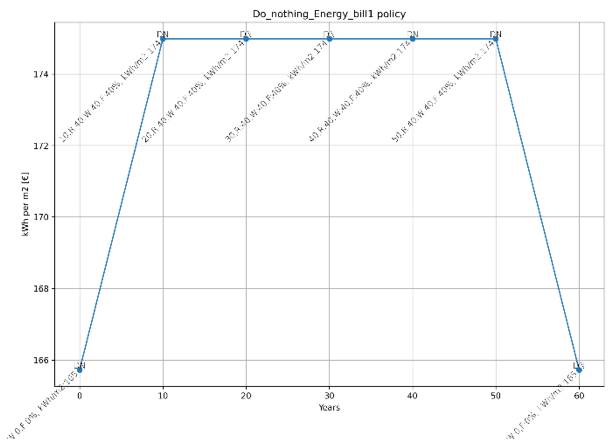


Do nothing policy: Sample Episode 2 Costs

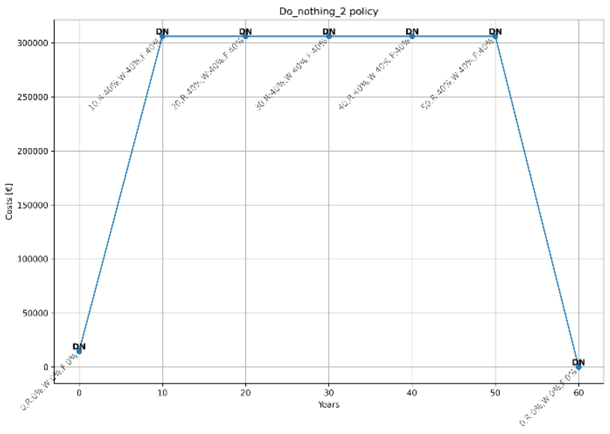
Do nothing policy: Sample Episode 2 States of energy demand



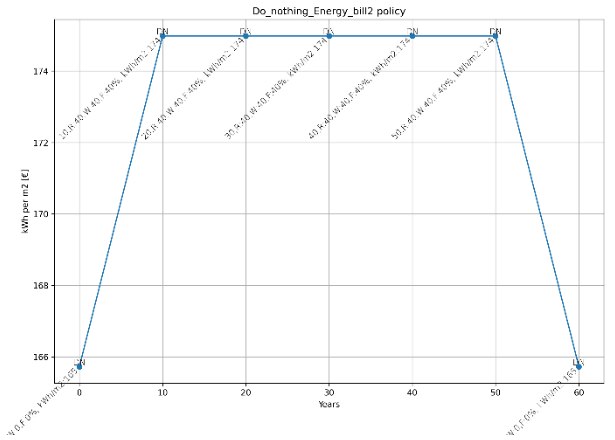
Do nothing policy: Sample Episode 3 Costs



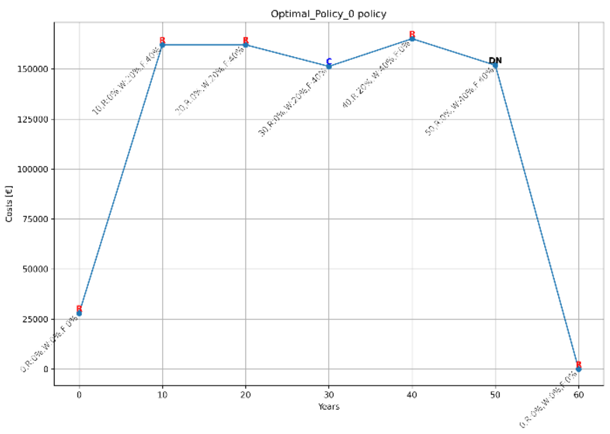
Do nothing policy: Sample Episode 3 States of energy demand



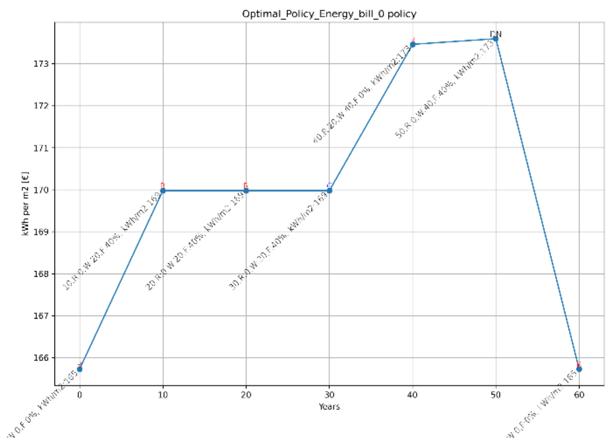
Optimal policy: Sample Episode 1 Costs



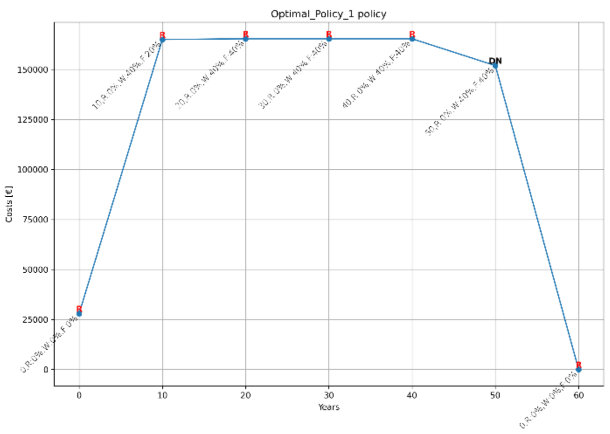
Optimal policy: Sample Episode 1 States of energy demand

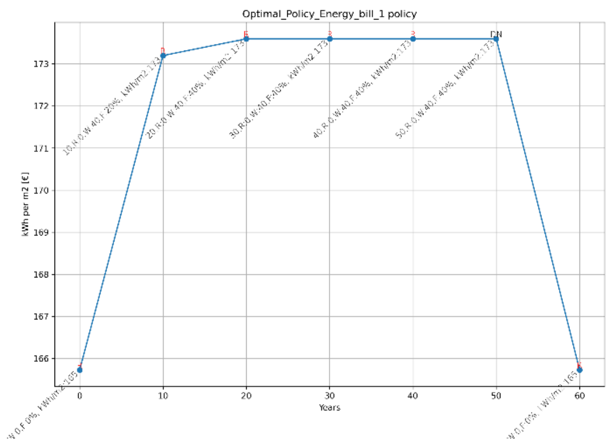


Optimal policy: Sample Episode 2 Costs



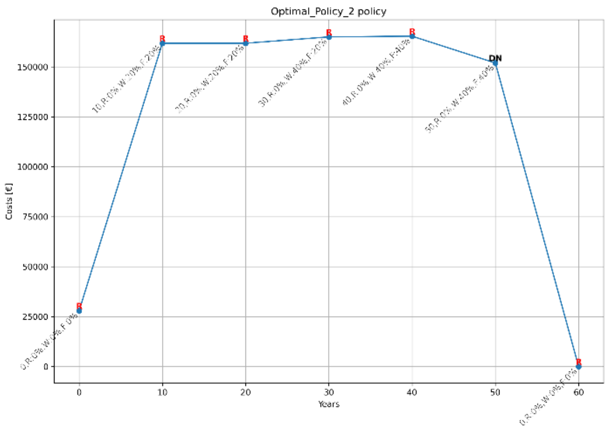
Optimal policy: Sample Episode 2 States of energy demand

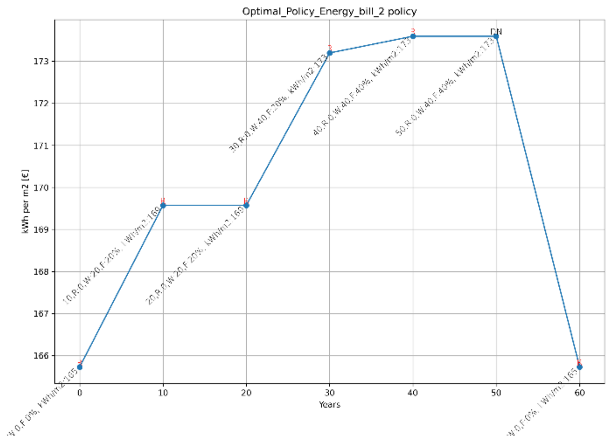




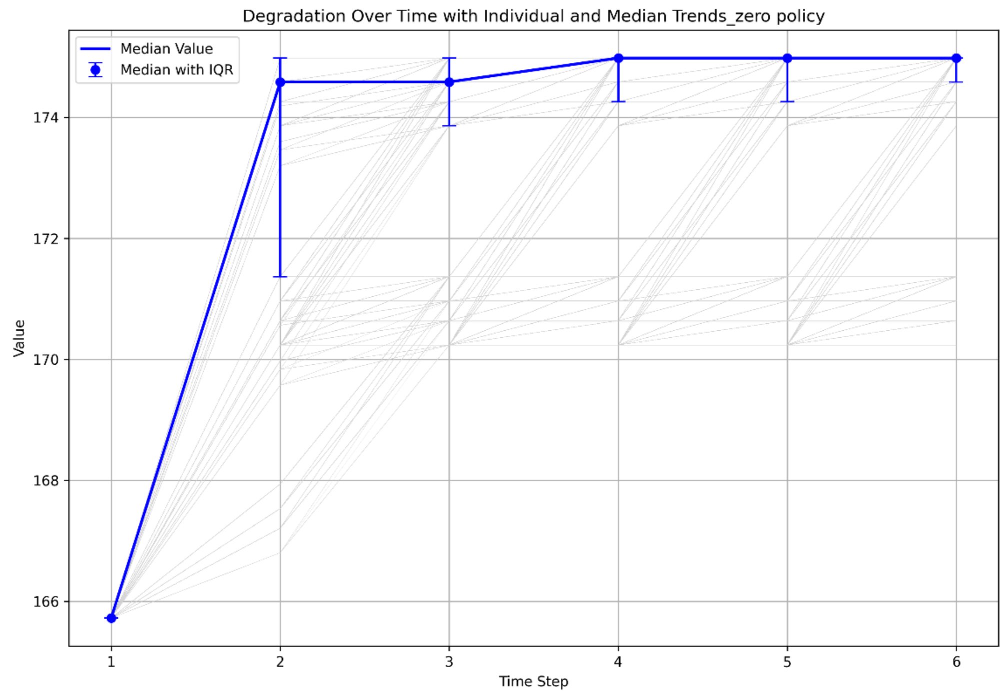
Optimal policy: Sample Episode 3 Costs

Optimal policy: Sample Episode 3 States of energy demand



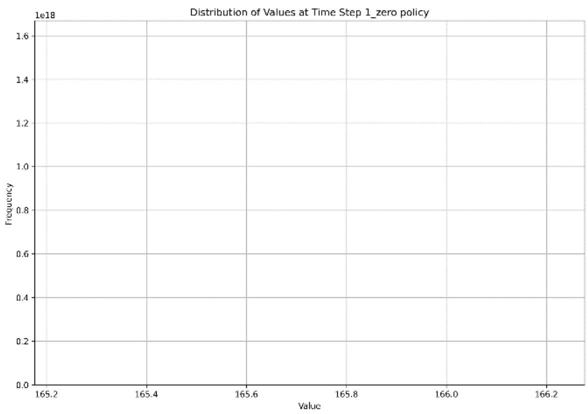


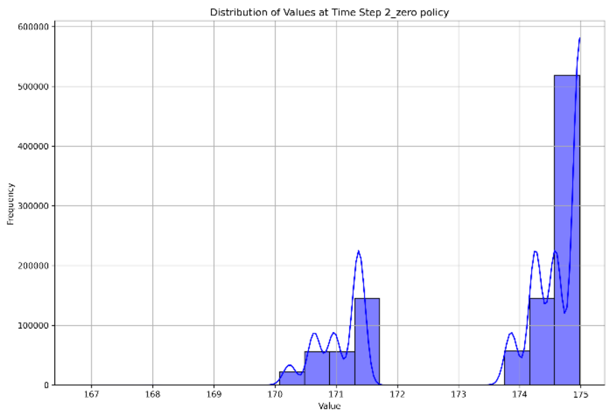
Median energy demand change with do nothing policy from one million episodes



Distribution of possible energy demand at time step 1 over million episodes with do nothing policy

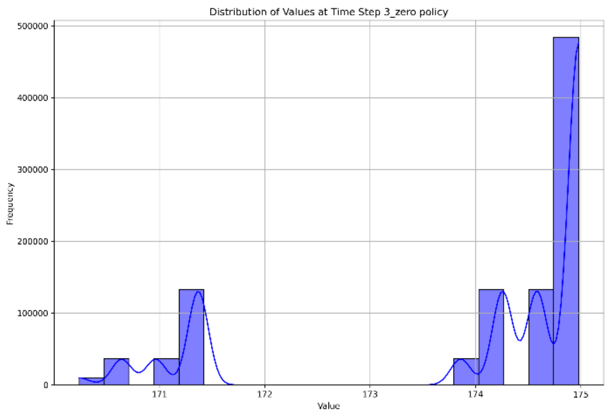
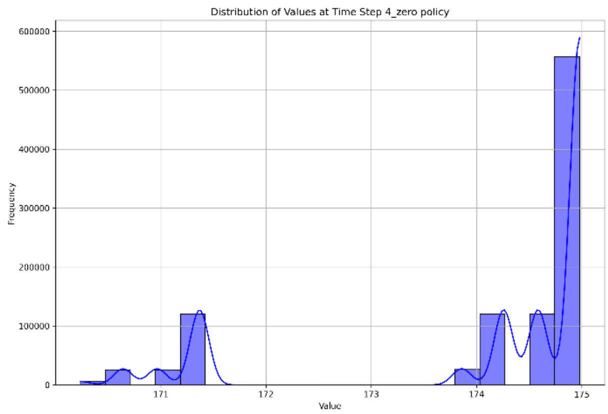
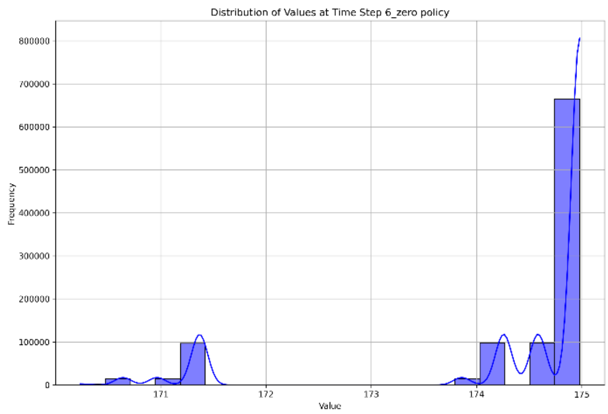
Distribution of possible energy demand at time step 2 over million episodes with do nothing policy

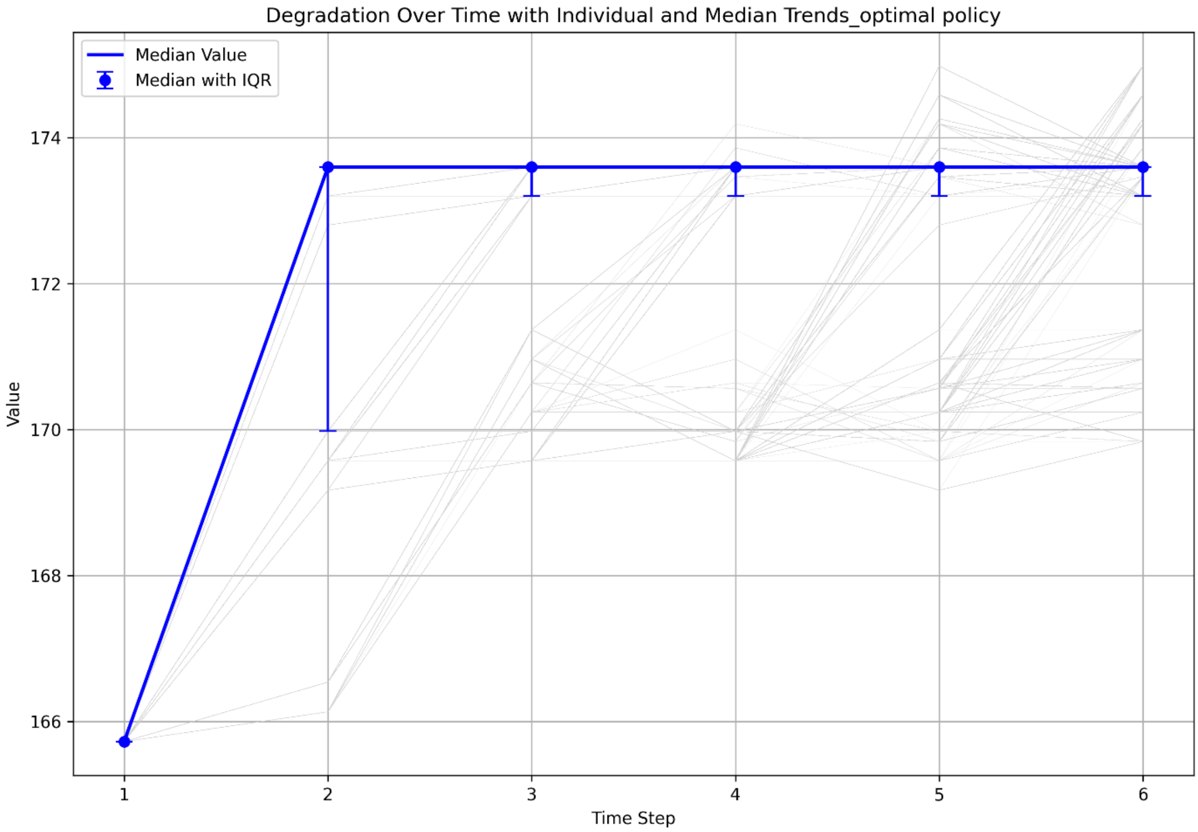


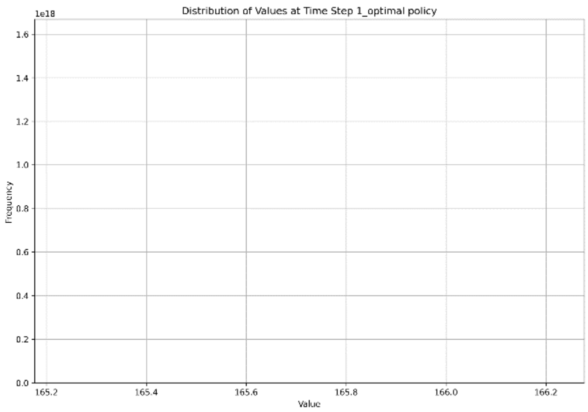
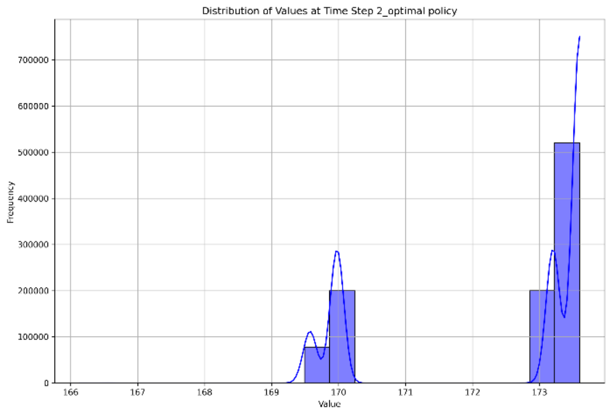


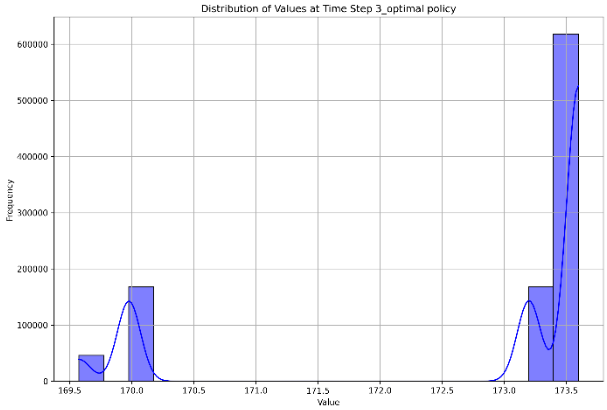
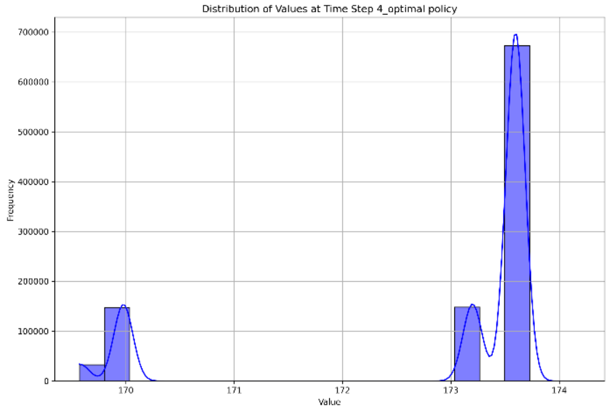
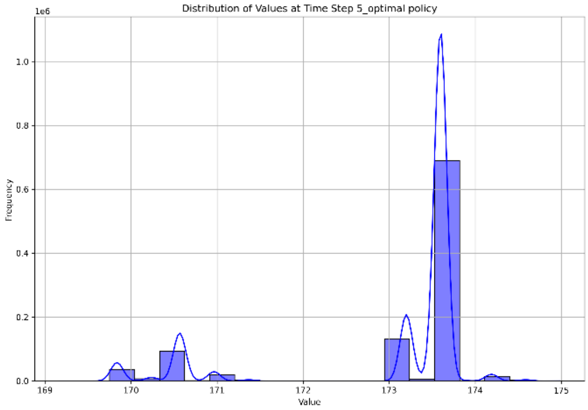
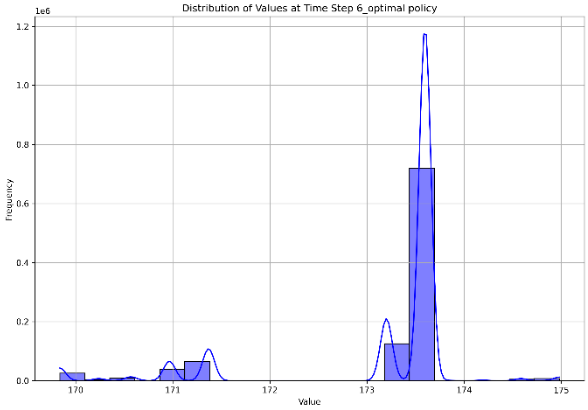
Distribution of possible energy demand at time step 3 over million episodes with do nothing policy

Distribution of possible energy demand at time step 4 over million episodes with do nothing policy

<table>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>Distribution of possible energy demand at time step 5 over million episodes with do nothing policy</td>
<td>Distribution of possible energy demand at time step 6 over million episodes with do nothing policy</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td colspan="2">Median energy demand change with optimal policy from one million episodes</td>
</tr>
</table>

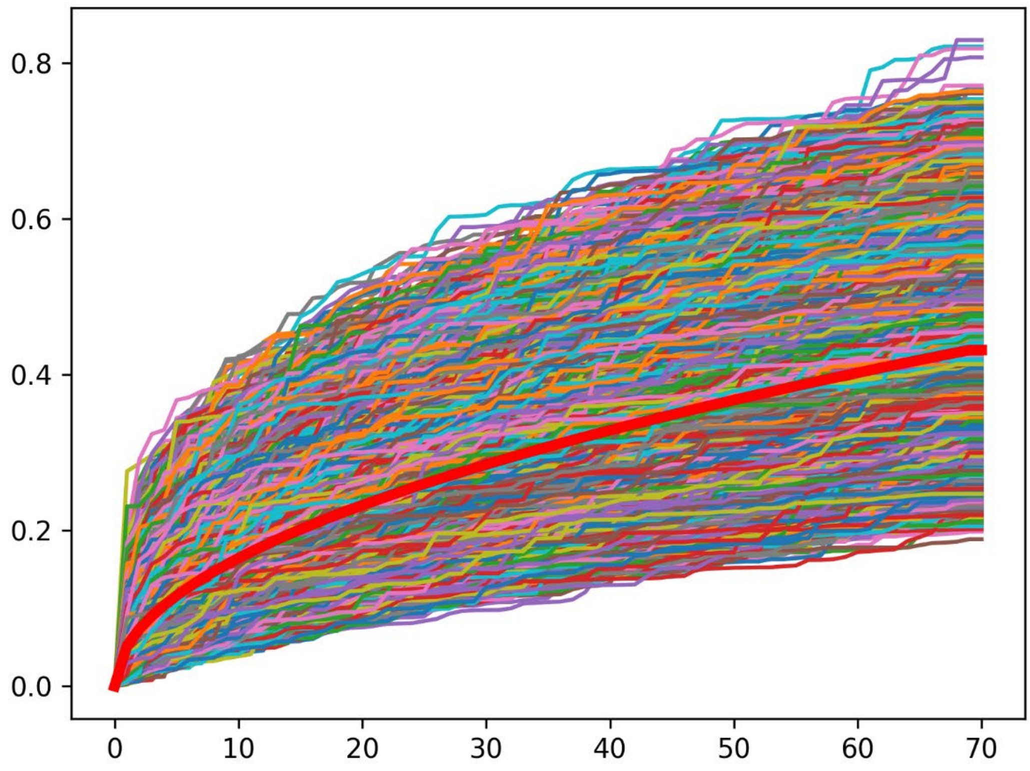


Distribution of possible energy demand at time step 1 over million episodes with optimal policy	Distribution of possible energy demand at time step 2 over million episodes with optimal policy
	
Distribution of possible energy demand at time step 3 over million episodes with optimal policy	Distribution of possible energy demand at time step 4 over million episodes with optimal policy

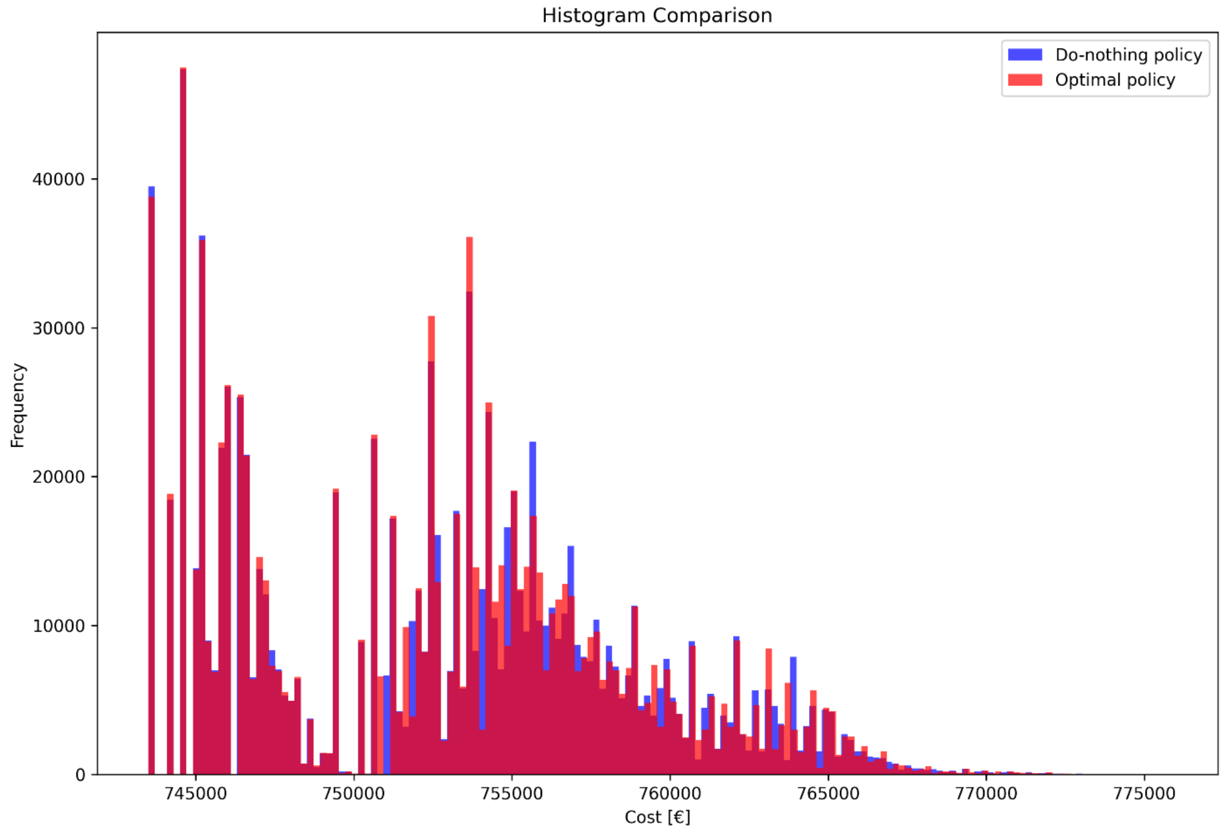
	
Distribution of possible energy demand at time step 5 over million episodes with optimal policy	Distribution of possible energy demand at time step 6 over million episodes with optimal policy
	

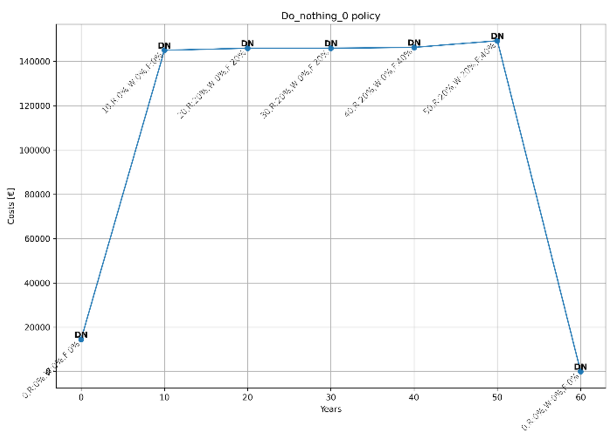
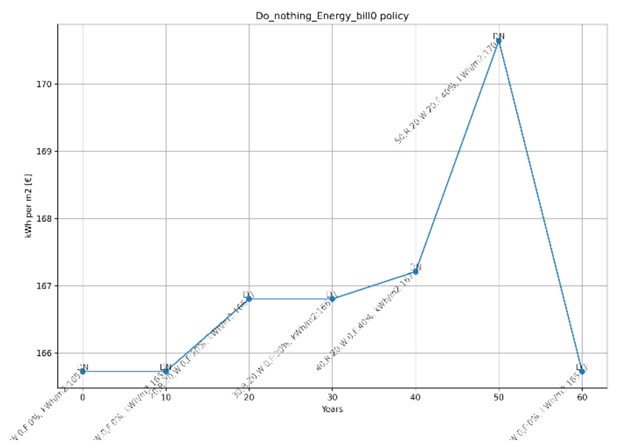
\

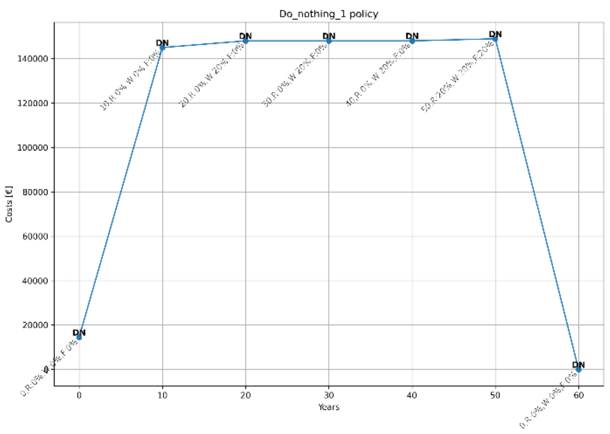
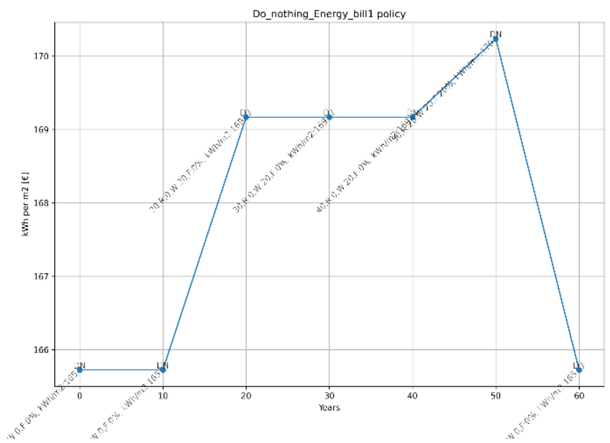
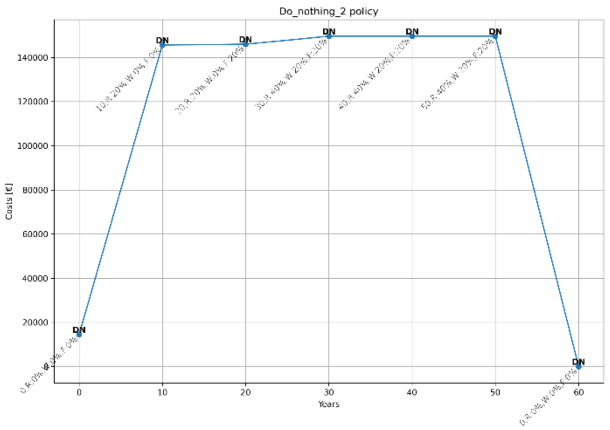
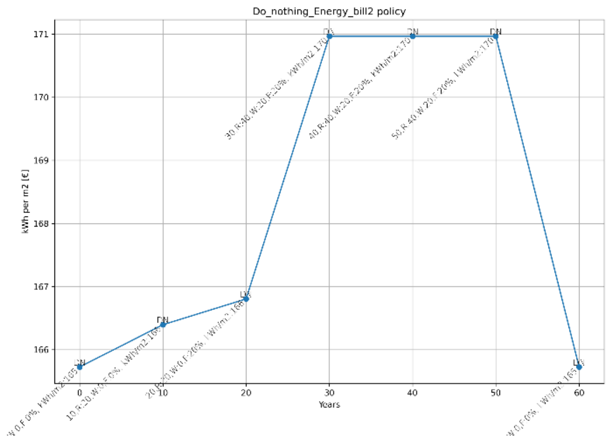
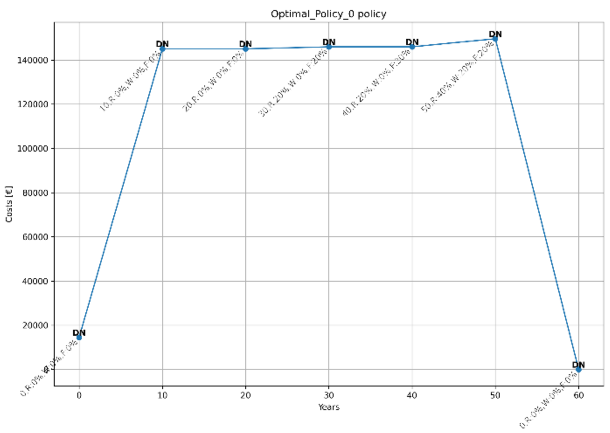
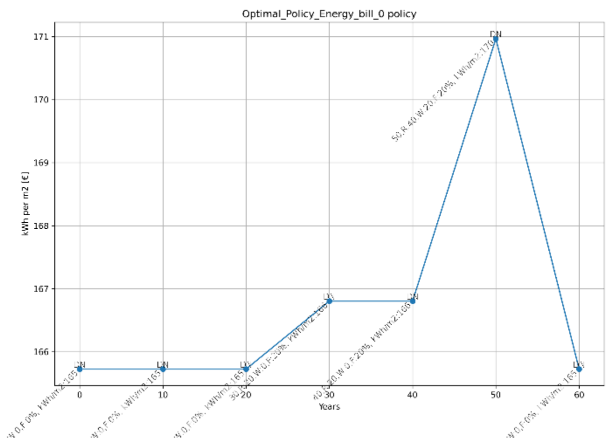
Simulation 3: No penalty, no infiltration simulation, beta 0.5
Optimal policy plot
Material degradation scenarios , 1000000 realizations

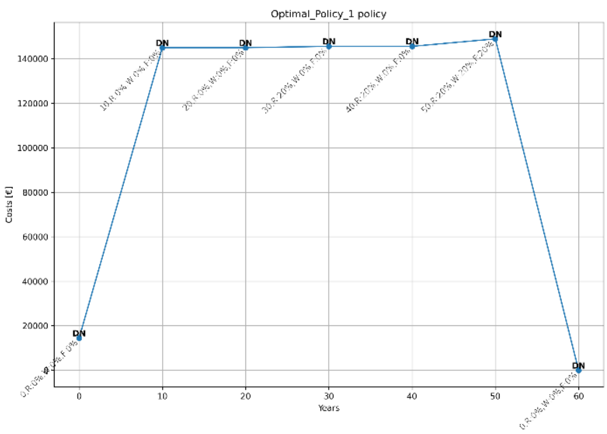


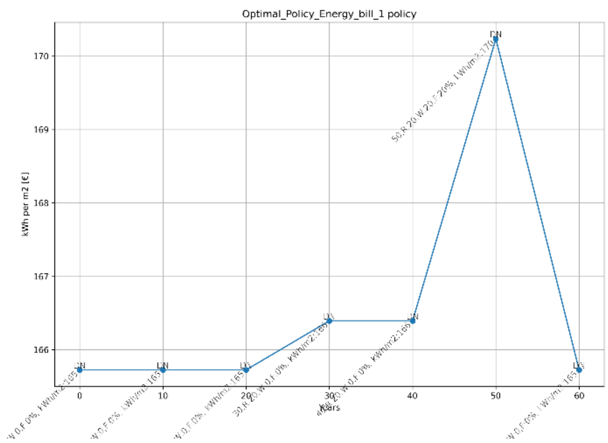
Histogram comparison of return between do nothing policy and optimal policy



Do nothing policy: Sample Episode 1 Costs	Do nothing policy: Sample Episode 1 States of energy demand
	
Do nothing policy: Sample Episode 2 Costs	Do nothing policy: Sample Episode 2 States of energy demand

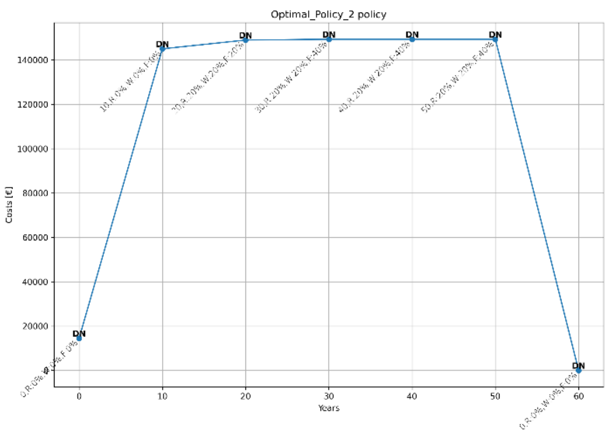
	
Do nothing policy: Sample Episode 3 Costs	Do nothing policy: Sample Episode 3 States of energy demand
	
Optimal policy: Sample Episode 1 Costs	Optimal policy: Sample Episode 1 States of energy demand
	
Optimal policy: Sample Episode 2 Costs	Optimal policy: Sample Episode 2 States of energy demand

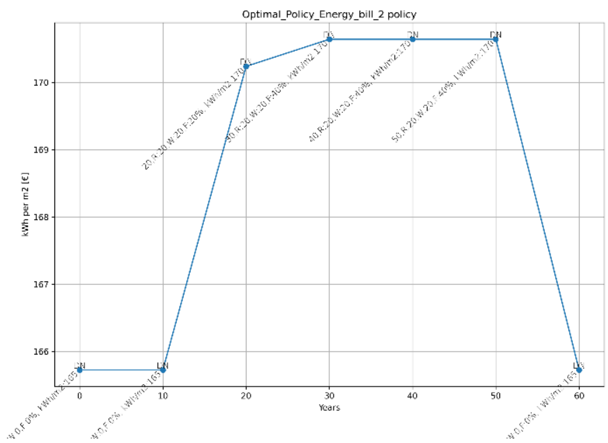




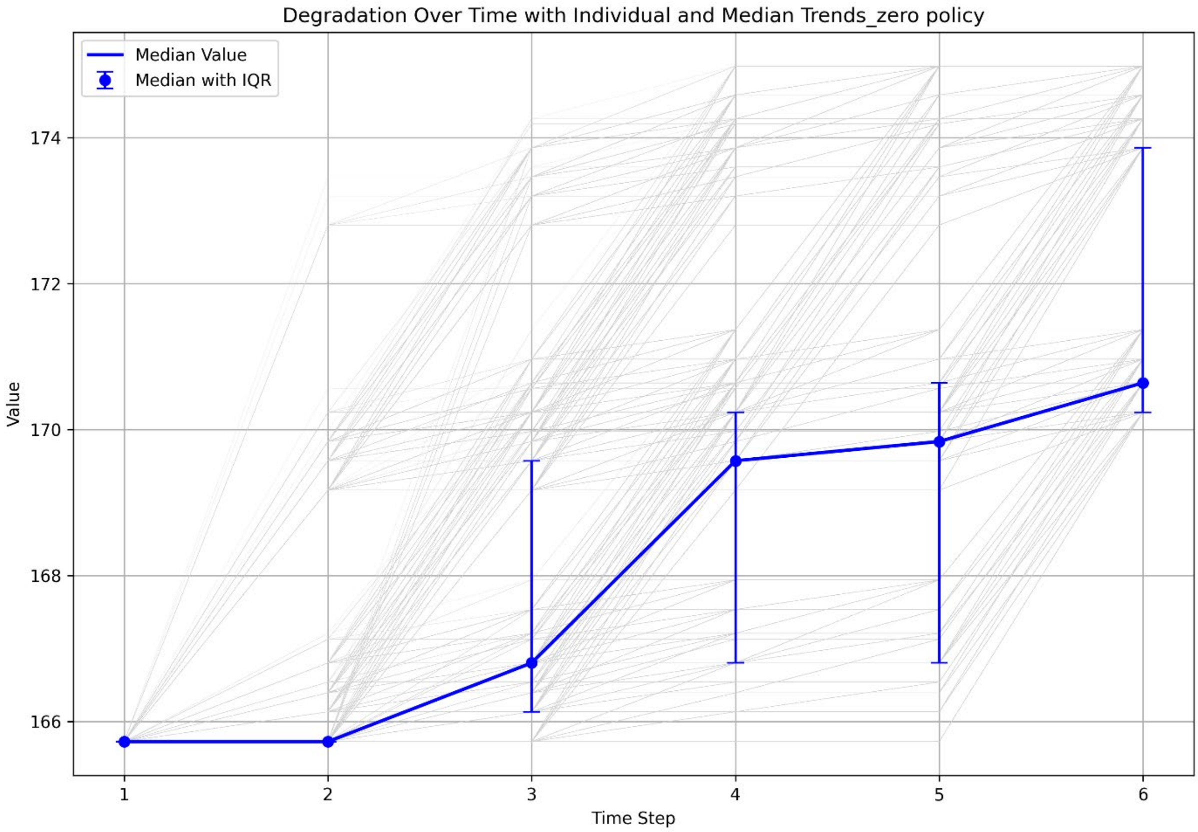
Optimal policy: Sample Episode 3 Costs

Optimal policy: Sample Episode 3 States of energy demand



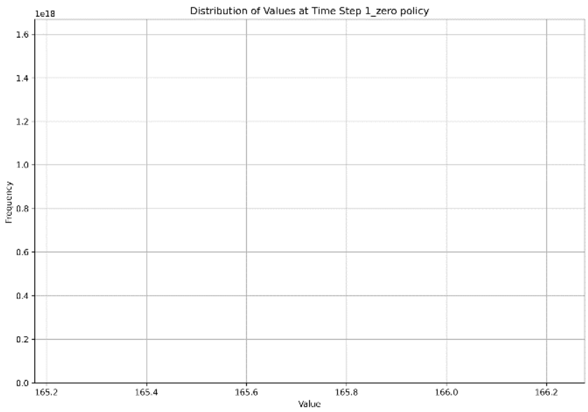


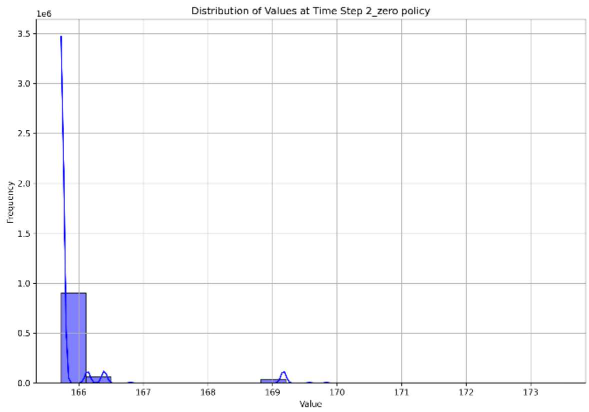
Median energy demand change with do nothing policy from one million episodes



Distribution of possible energy demand at time step 1 over million episodes with do nothing policy

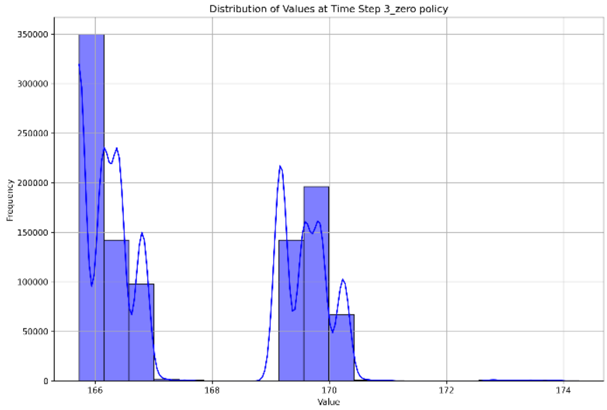
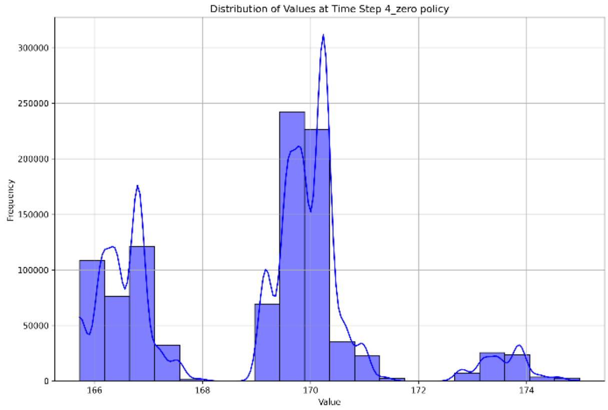
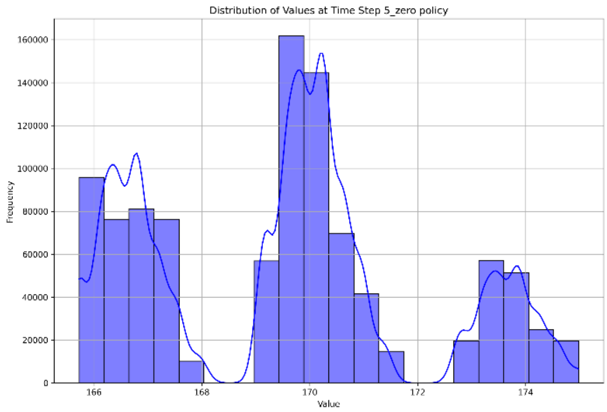
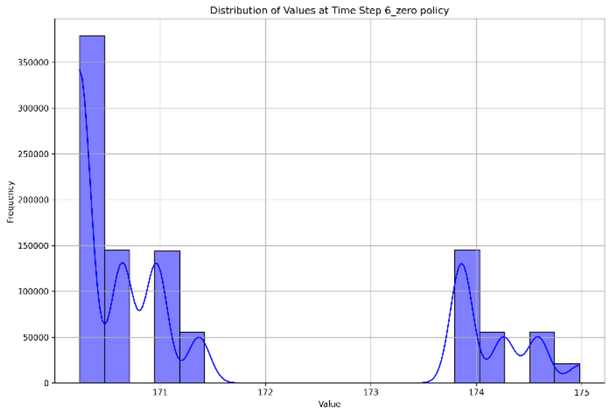
Distribution of possible energy demand at time step 2 over million episodes with do nothing policy



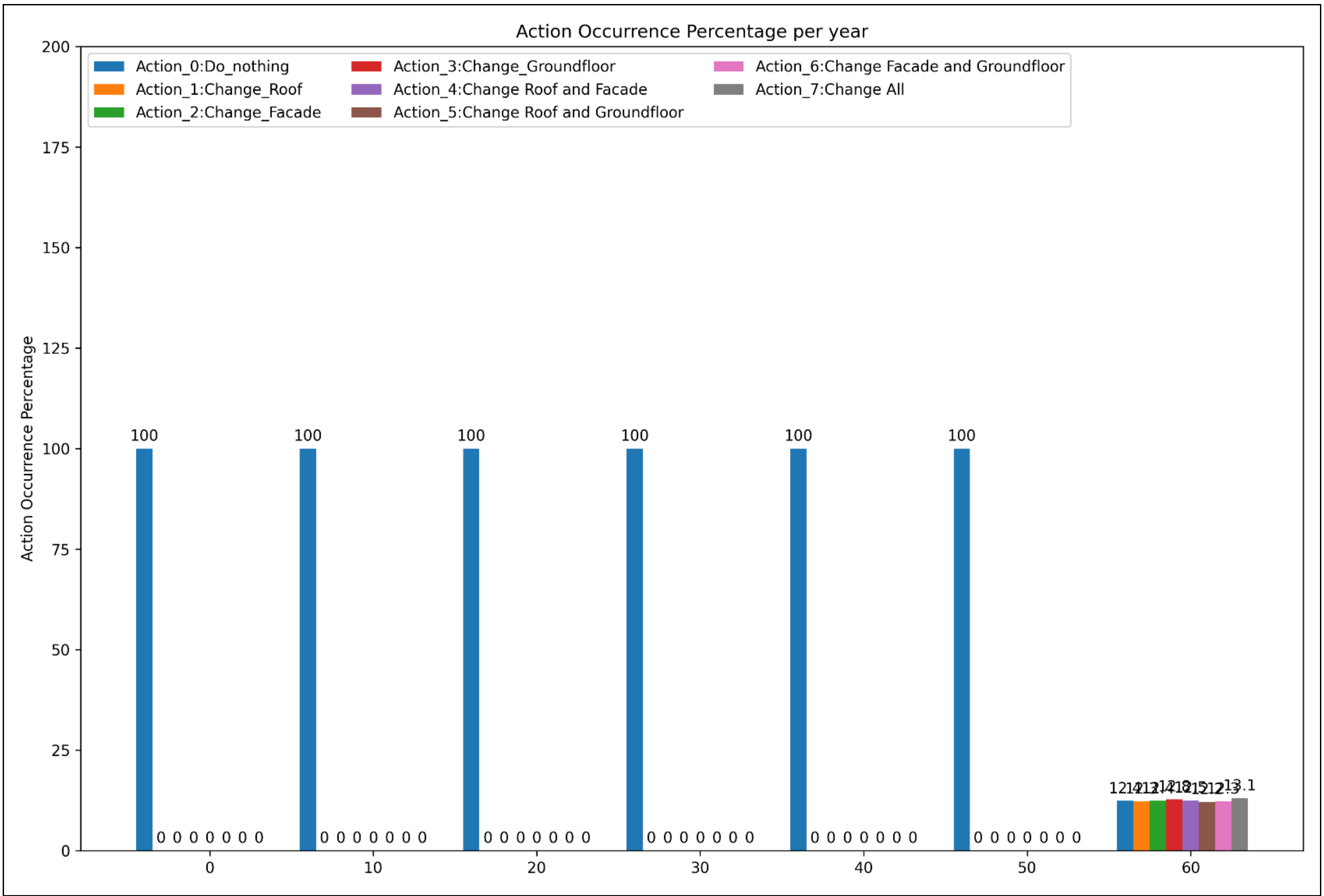


Distribution of possible energy demand at time step 3 over million episodes with do nothing policy

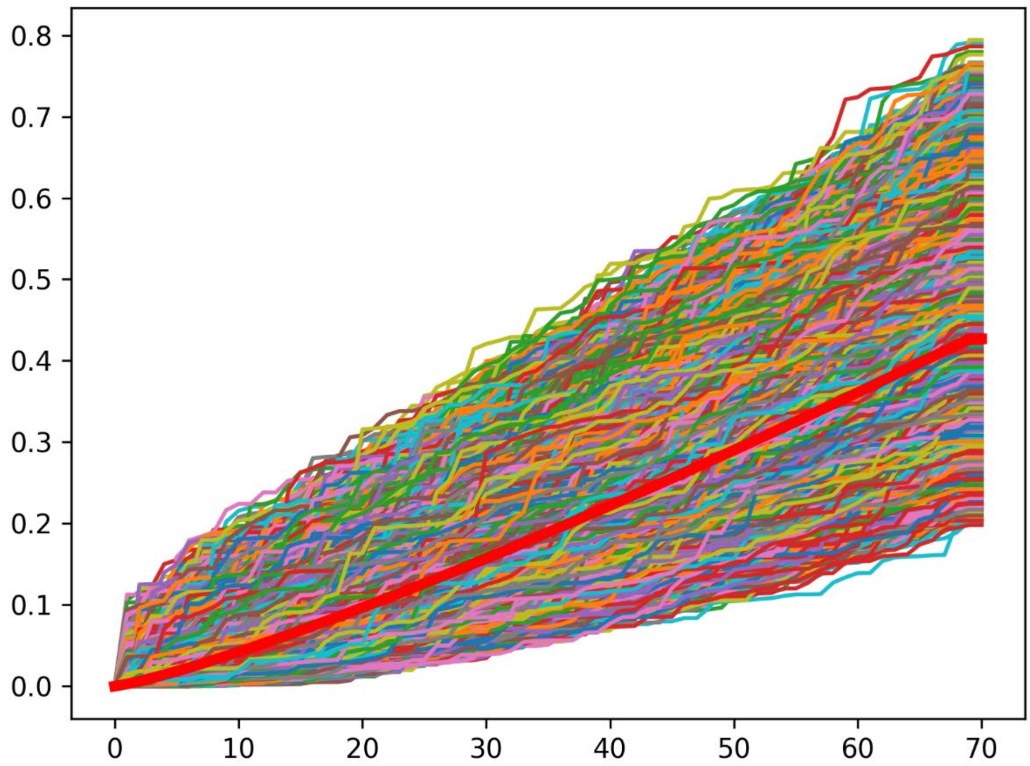
Distribution of possible energy demand at time step 4 over million episodes with do nothing policy

<table>
<tr>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>Distribution of possible energy demand at time step 5 over million episodes with do nothing policy</td>
<td>Distribution of possible energy demand at time step 6 over million episodes with do nothing policy</td>
</tr>
<tr>
<td>
</td>
<td>
</td>
</tr>
</table>

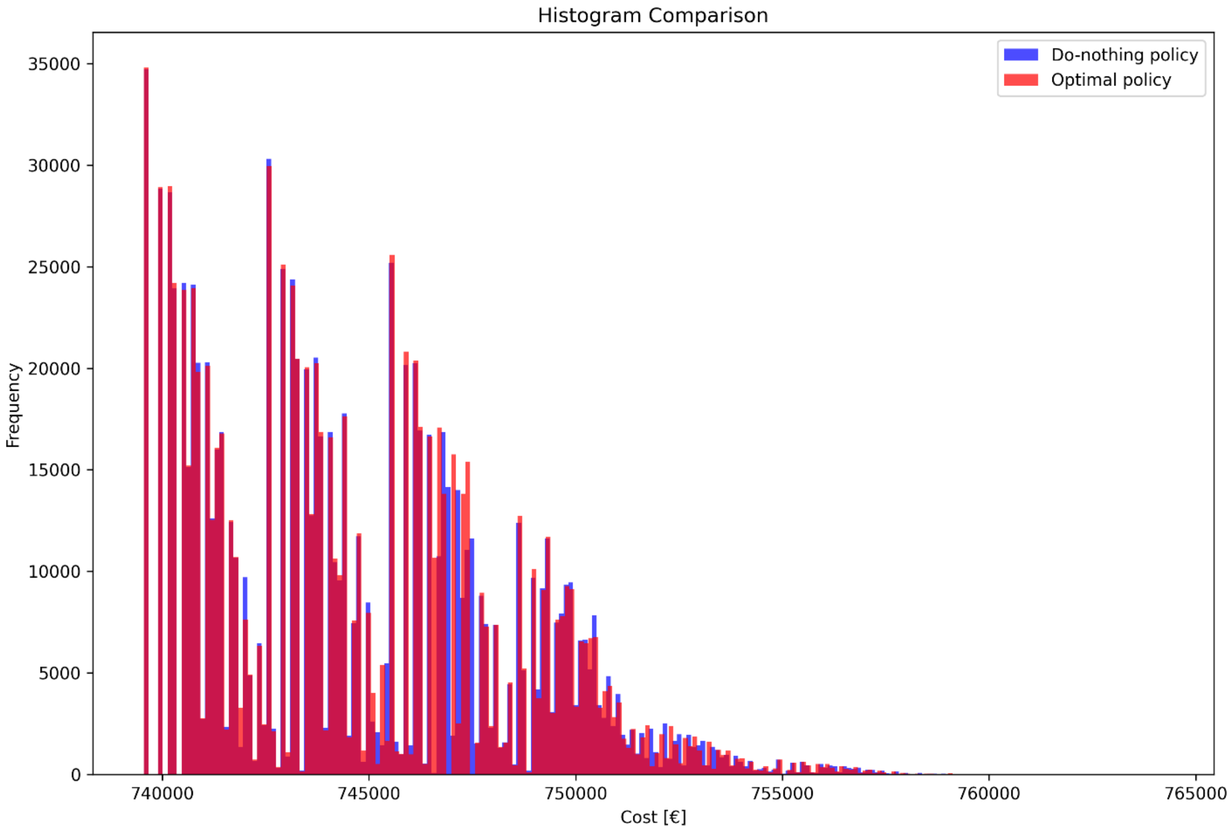
Simulation 4: No penalty, no infiltration simulation, beta 1.2
Optimal policy plot



Material degradation scenarios , 1000000 realizations

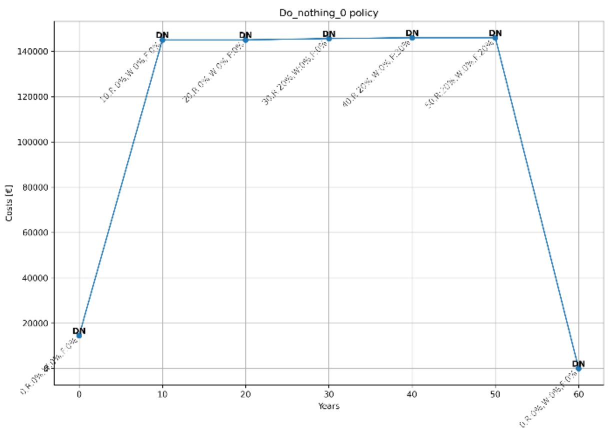


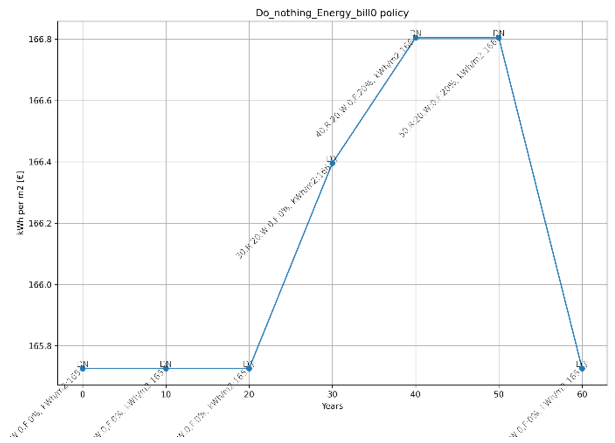
Histogram comparison of return between do nothing policy and optimal policy



Do nothing policy: Sample Episode 1 Costs

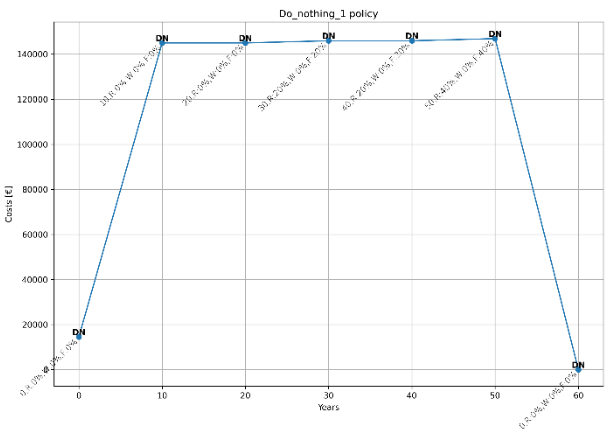
Do nothing policy: Sample Episode 1 States of energy demand



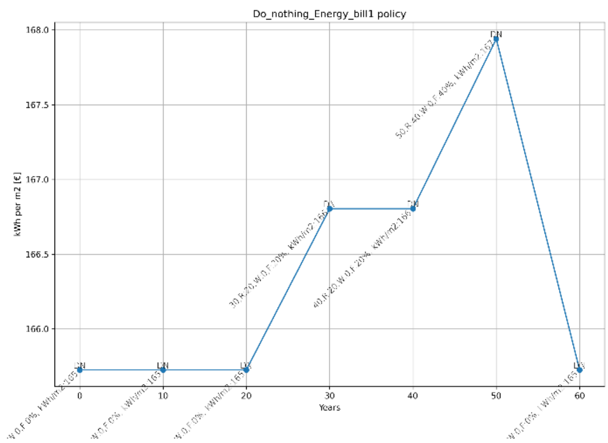


Do nothing policy: Sample Episode 2 Costs

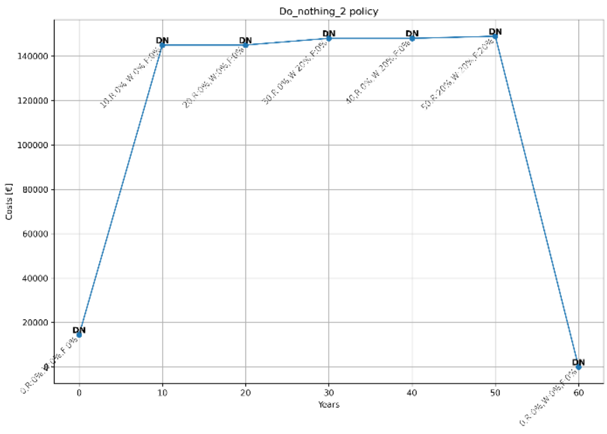
Do nothing policy: Sample Episode 2 States of energy demand



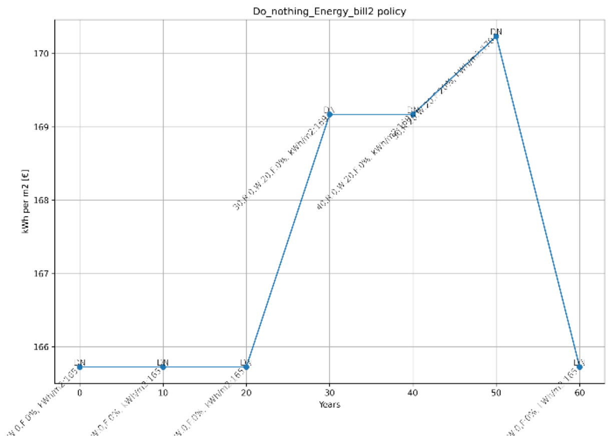
Do nothing policy: Sample Episode 3 Costs



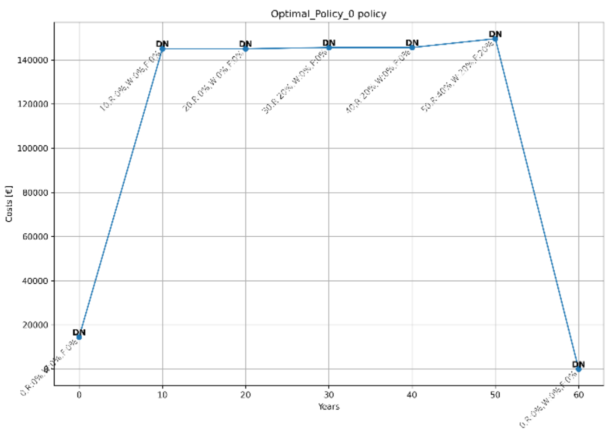
Do nothing policy: Sample Episode 3 States of energy demand



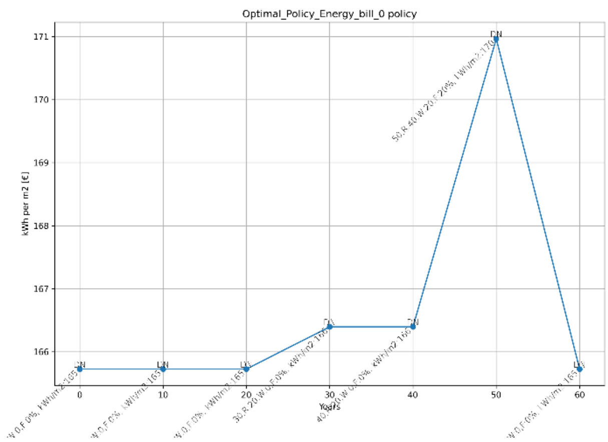
Optimal policy: Sample Episode 1 Costs



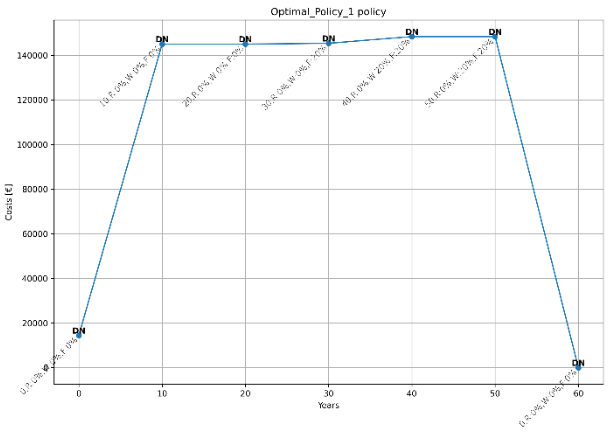
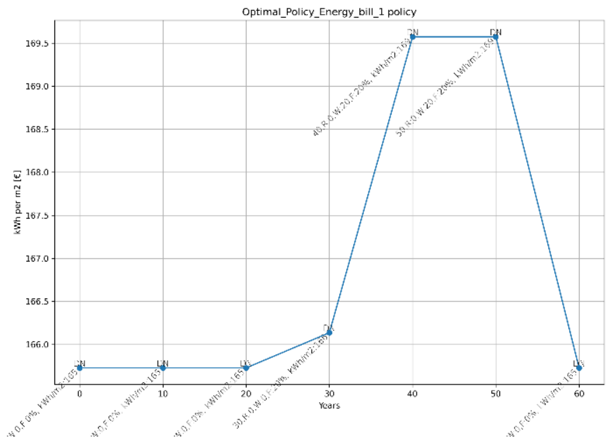
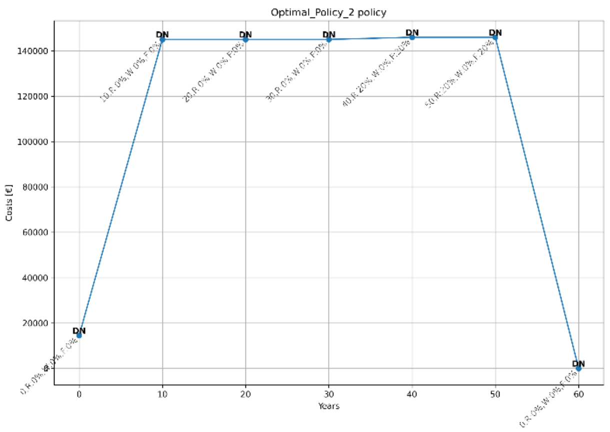
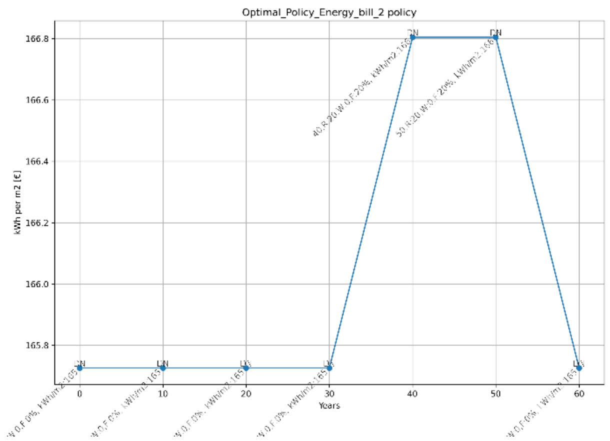
Optimal policy: Sample Episode 1 States of energy demand

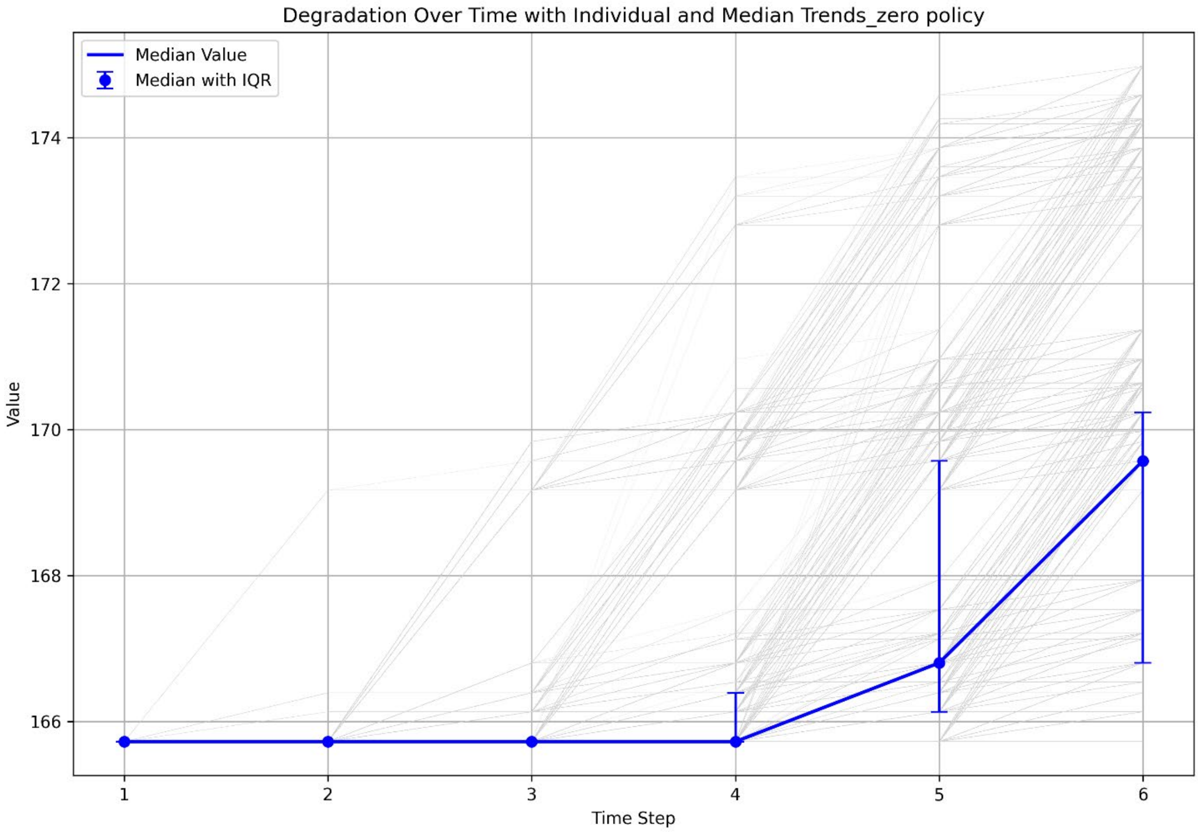


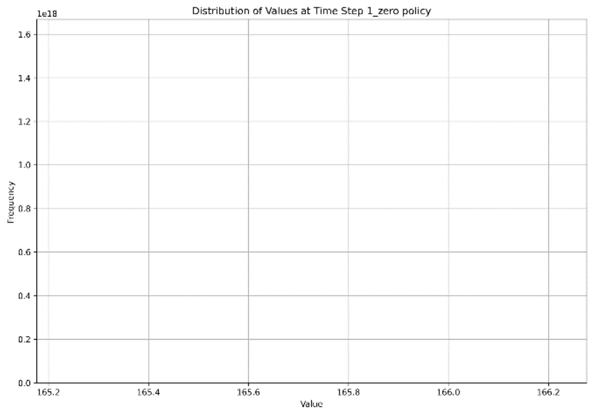
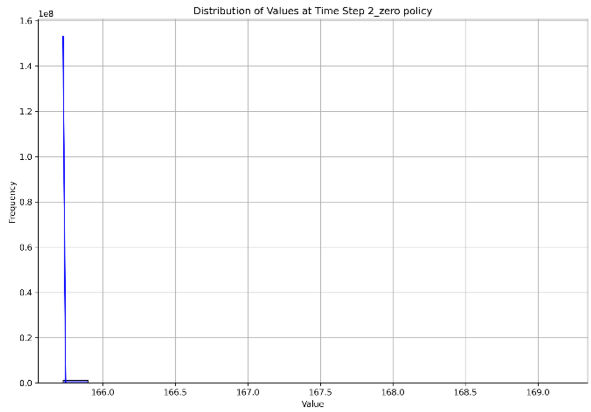
Optimal policy: Sample Episode 2 Costs

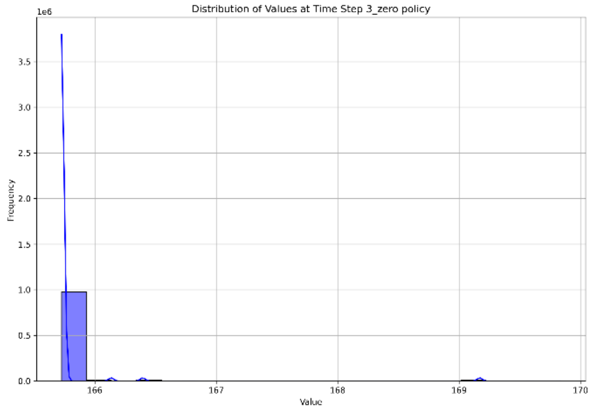
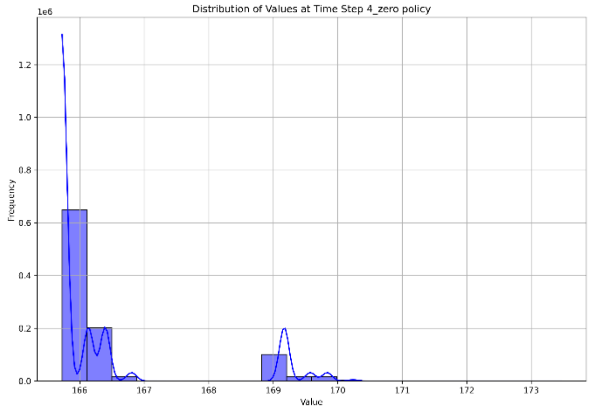
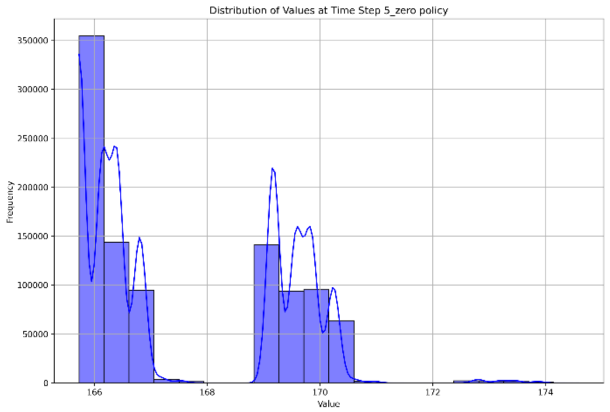
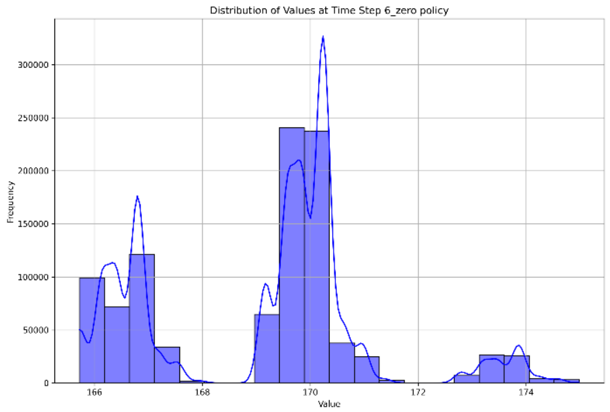


Optimal policy: Sample Episode 2 States of energy demand

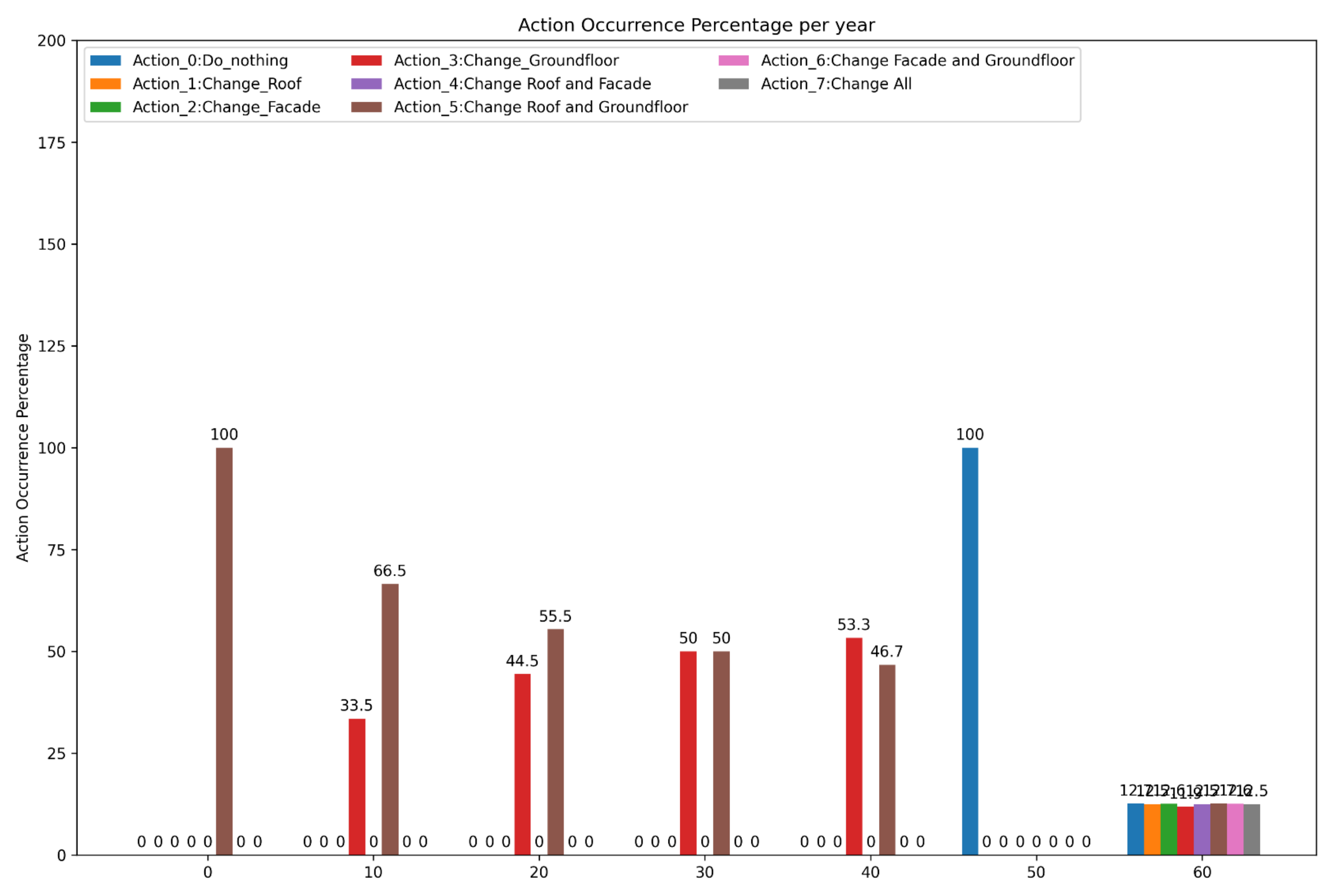
	
Optimal policy: Sample Episode 3 Costs	Optimal policy: Sample Episode 3 States of energy demand
	
Median energy demand change with do nothing policy from one million episodes	



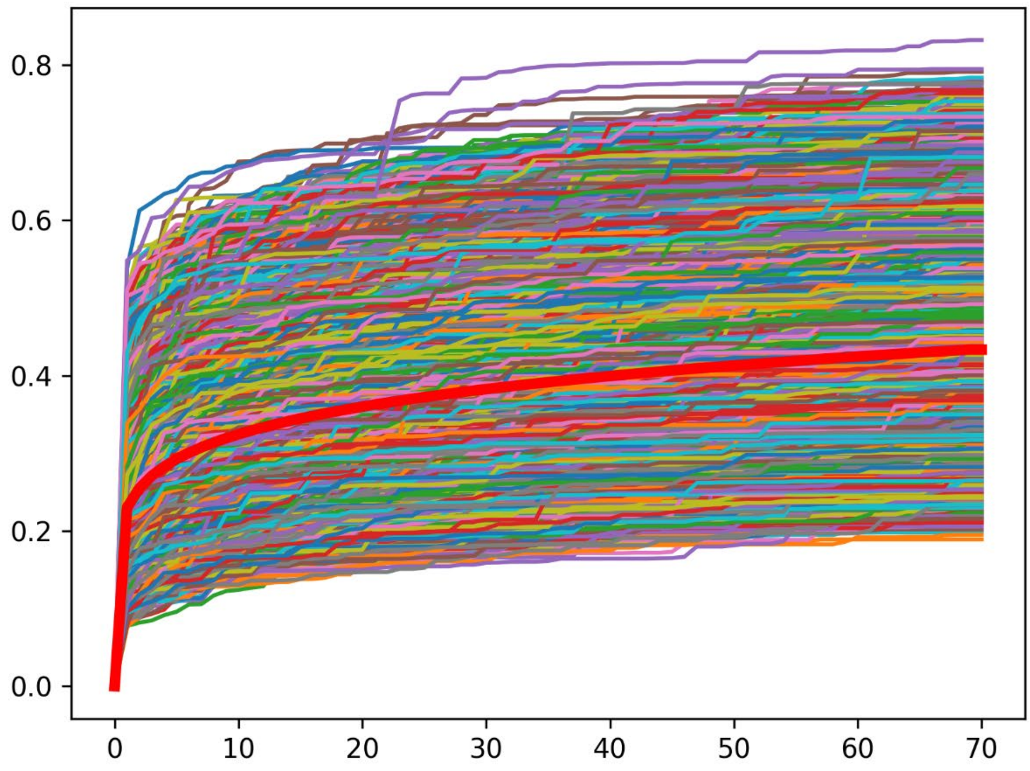
Distribution of possible energy demand at time step 1 over million episodes with do nothing policy	Distribution of possible energy demand at time step 2 over million episodes with do nothing policy
	
Distribution of possible energy demand at time step 3 over million episodes with do nothing policy	Distribution of possible energy demand at time step 4 over million episodes with do nothing policy

 	
Distribution of possible energy demand at time step 5 over million episodes with do nothing policy	Distribution of possible energy demand at time step 6 over million episodes with do nothing policy
 	

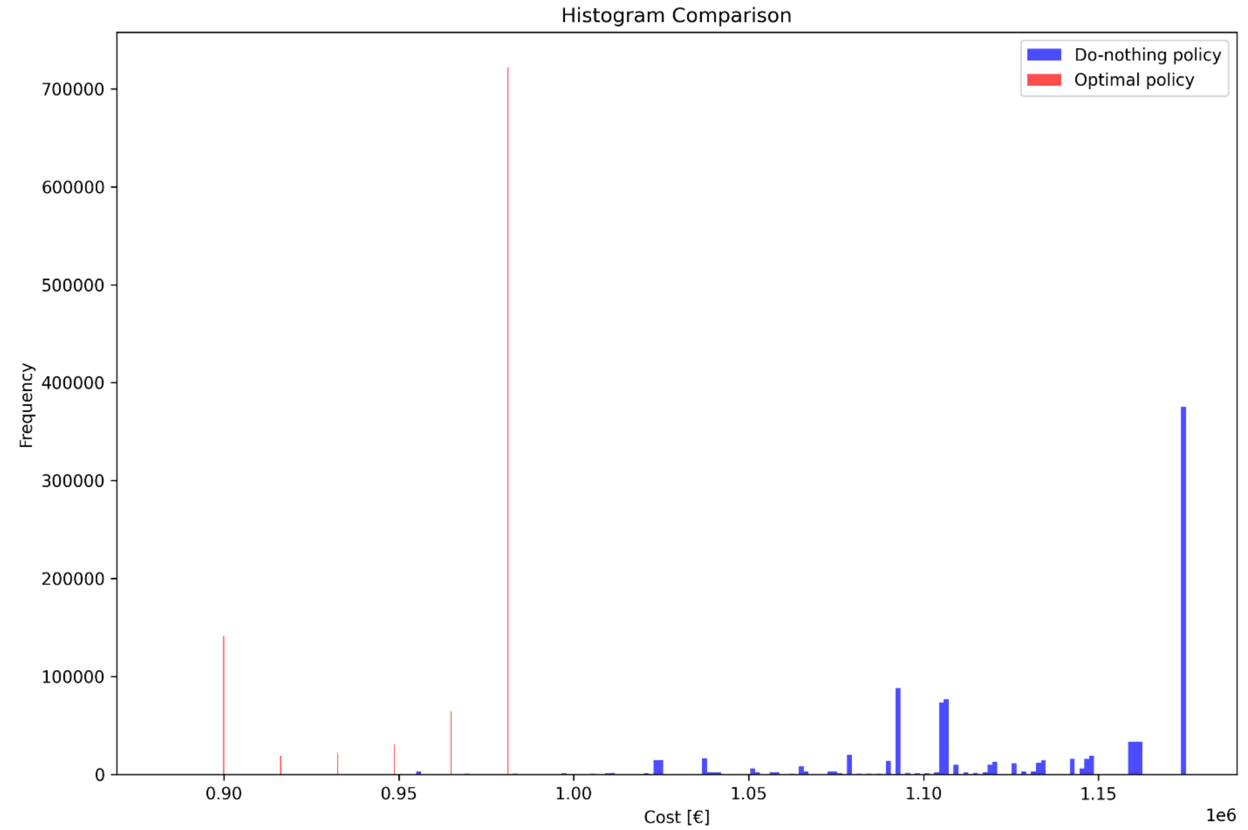
Simulation 5: No penalty, infiltration simulation, beta 0.15
Optimal policy plot

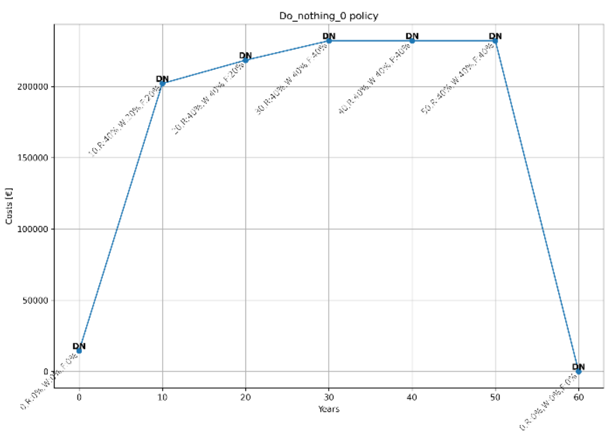
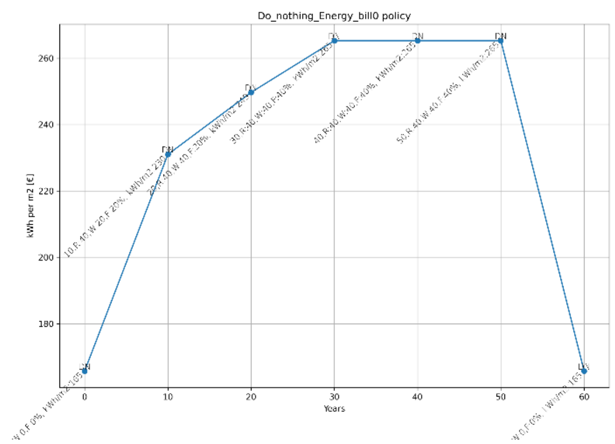


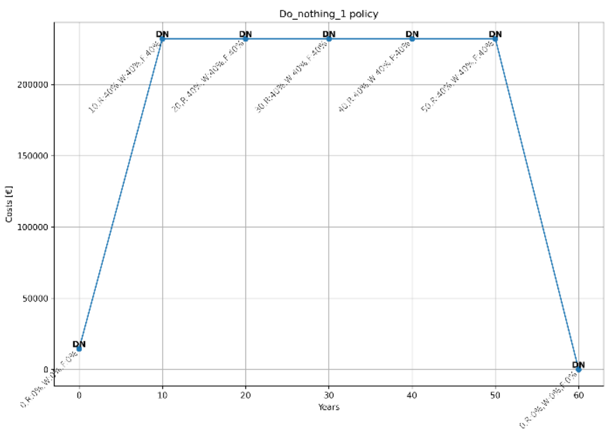
Material degradation scenarios , 1000000 realizations



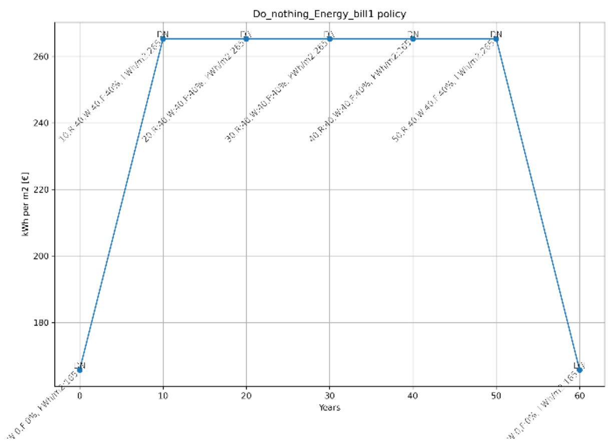
Histogram comparison of return between do nothing policy and optimal policy



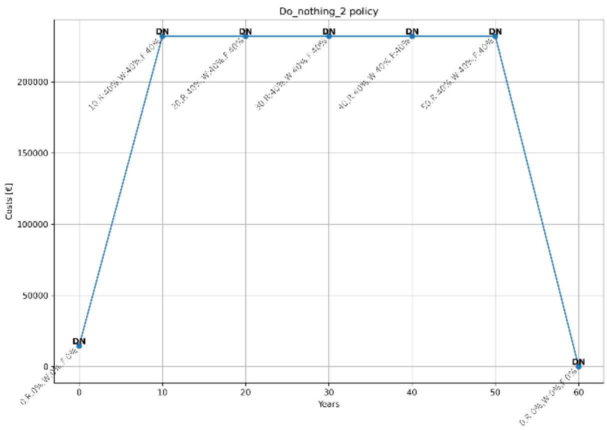
Do nothing policy: Sample Episode 1 Costs	Do nothing policy: Sample Episode 1 States of energy demand
	
Do nothing policy: Sample Episode 2 Costs	Do nothing policy: Sample Episode 2 States of energy demand



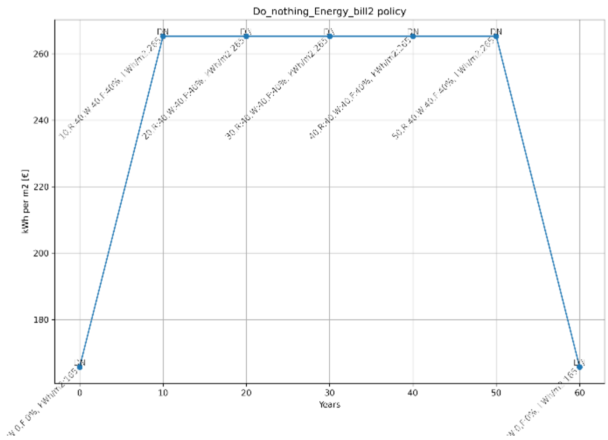
Do nothing policy: Sample Episode 3 Costs



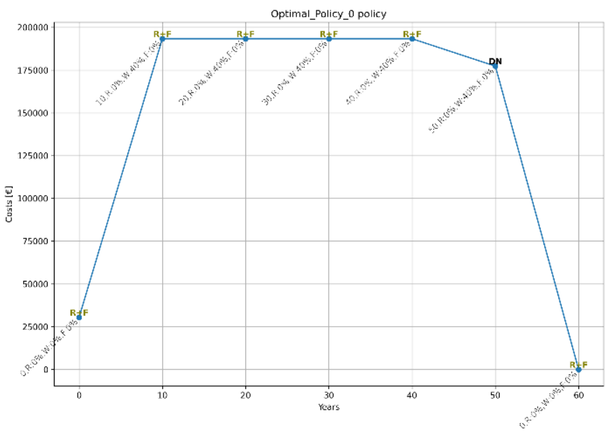
Do nothing policy: Sample Episode 3 States of energy demand



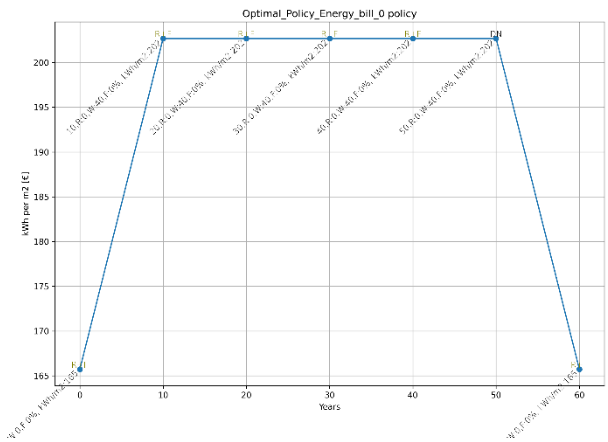
Optimal policy: Sample Episode 1 Costs



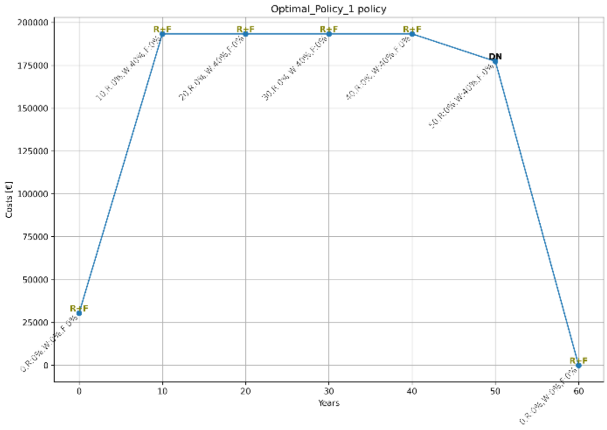
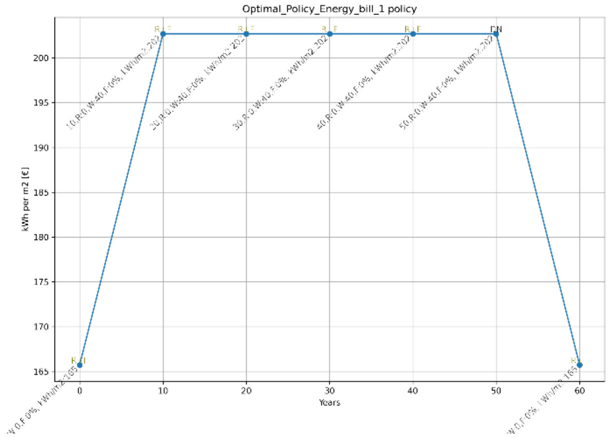
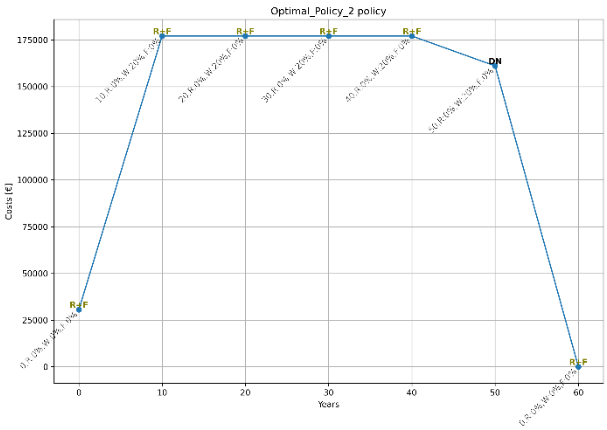
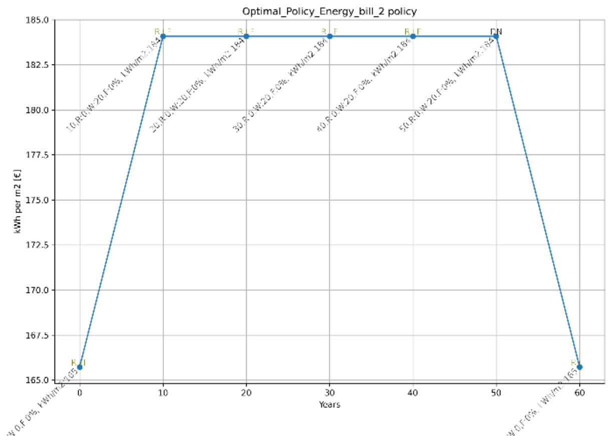
Optimal policy: Sample Episode 1 States of energy demand

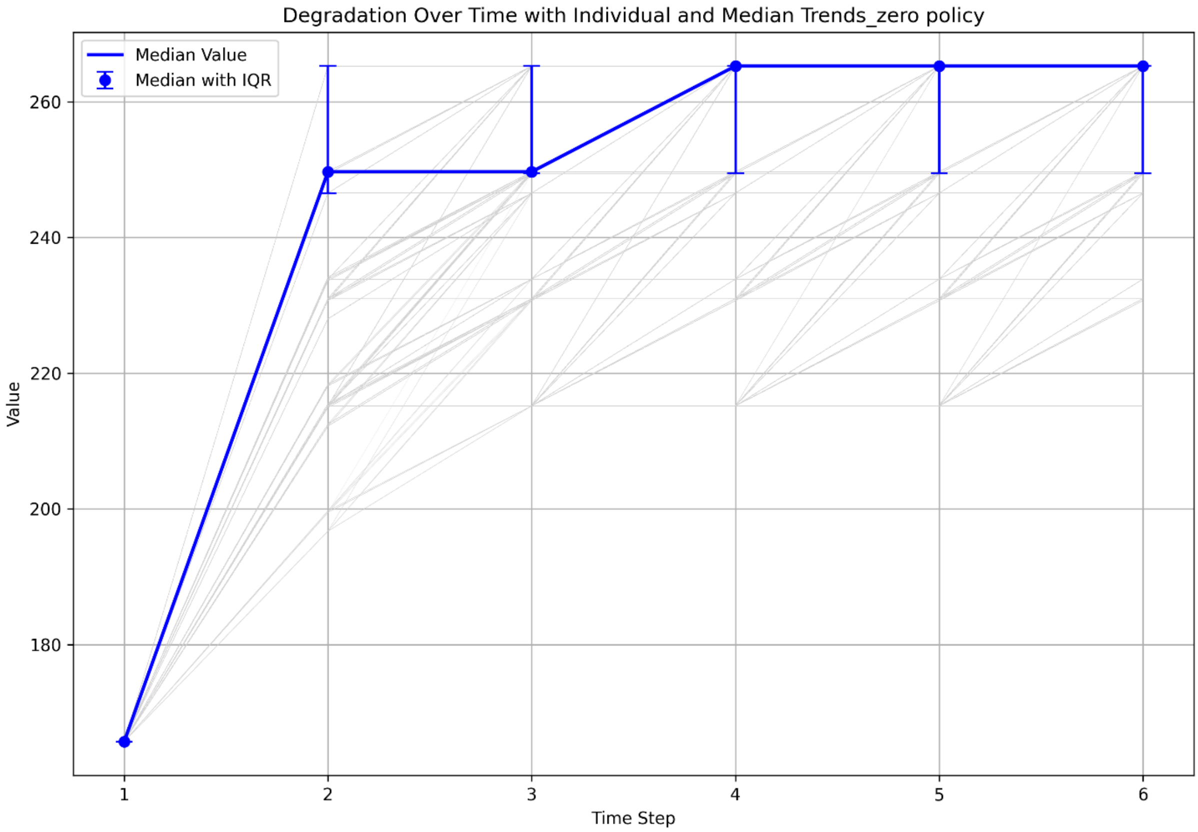


Optimal policy: Sample Episode 2 Costs



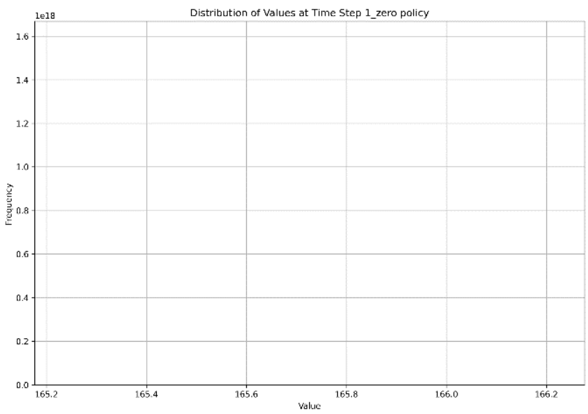
Optimal policy: Sample Episode 2 States of energy demand

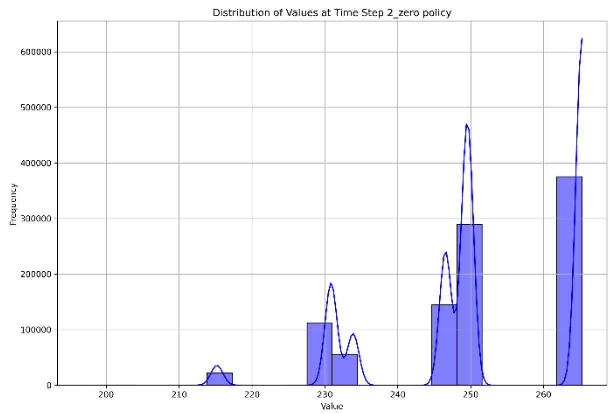
	
Optimal policy: Sample Episode 3 Costs	Optimal policy: Sample Episode 3 States of energy demand
	
Median energy demand change with do nothing policy from one million episodes	



Distribution of possible energy demand at time step 1 over million episodes with do nothing policy

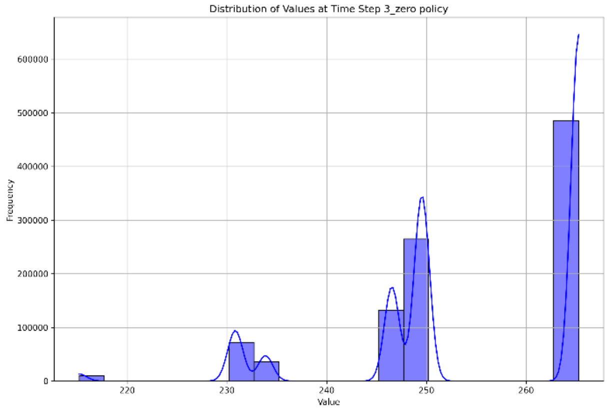
Distribution of possible energy demand at time step 2 over million episodes with do nothing policy

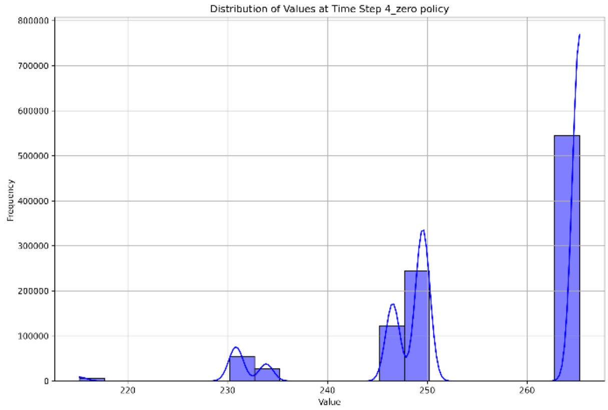




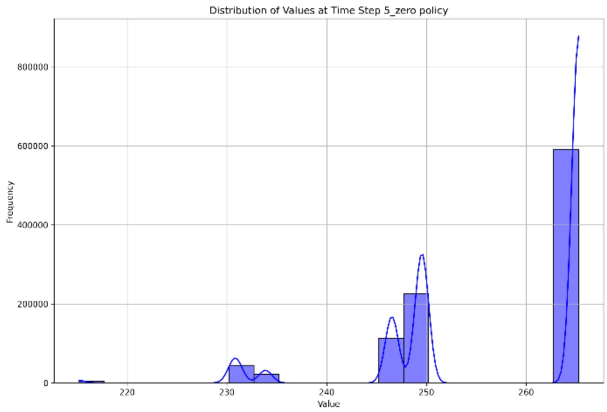
Distribution of possible energy demand at time step 3 over million episodes with do nothing policy

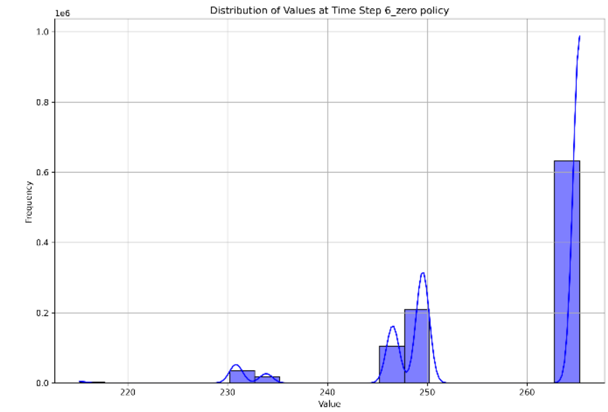
Distribution of possible energy demand at time step 4 over million episodes with do nothing policy



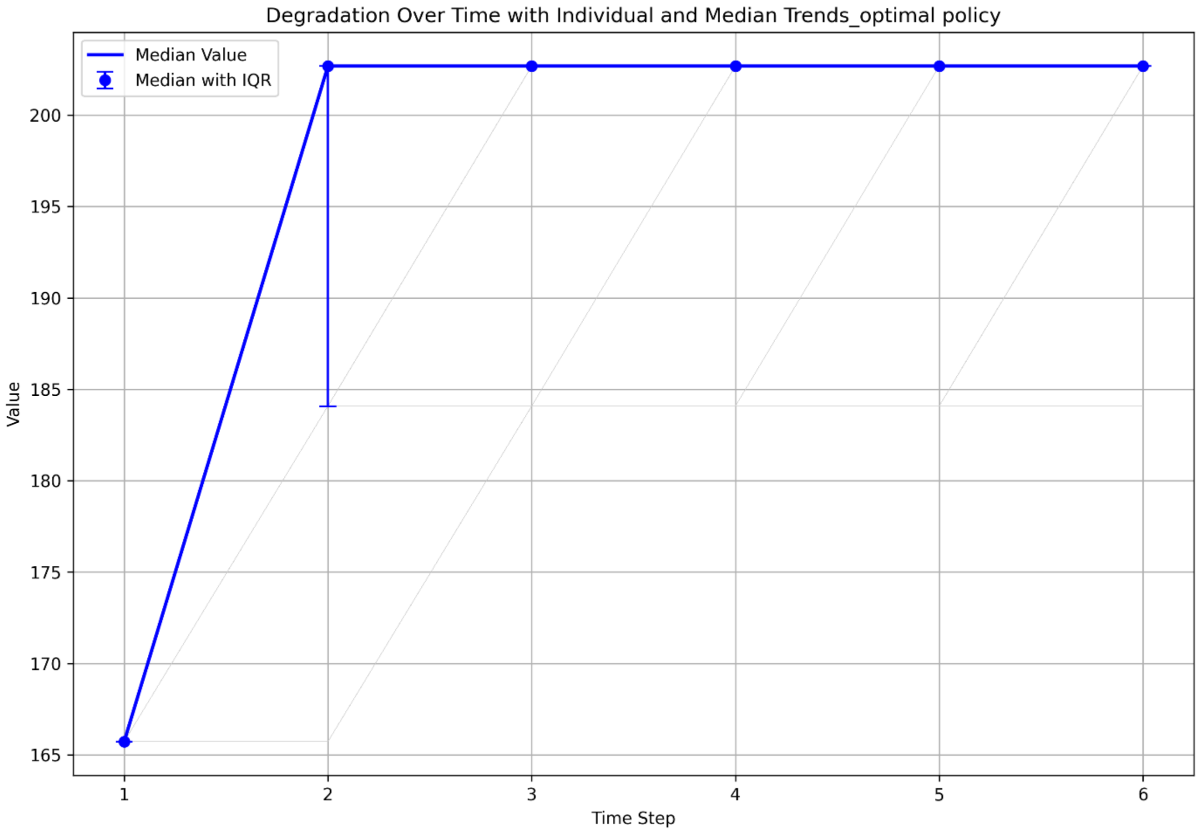


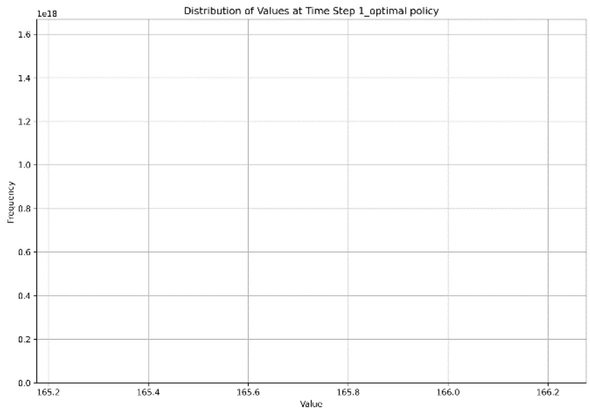
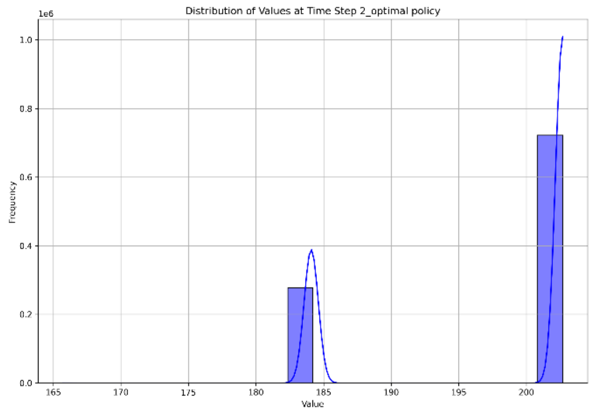
Distribution of possible energy demand at time step 5 over million episodes with do nothing policy	Distribution of possible energy demand at time step 6 over million episodes with do nothing policy

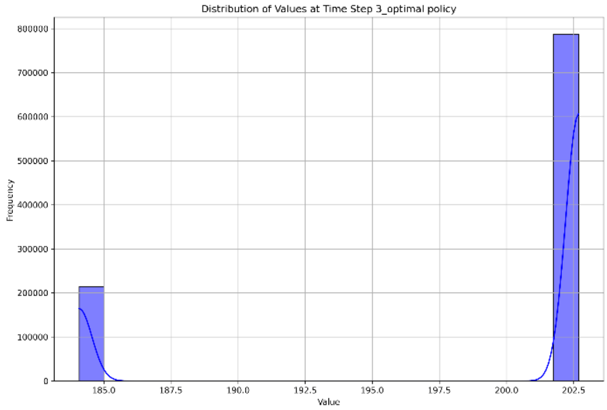


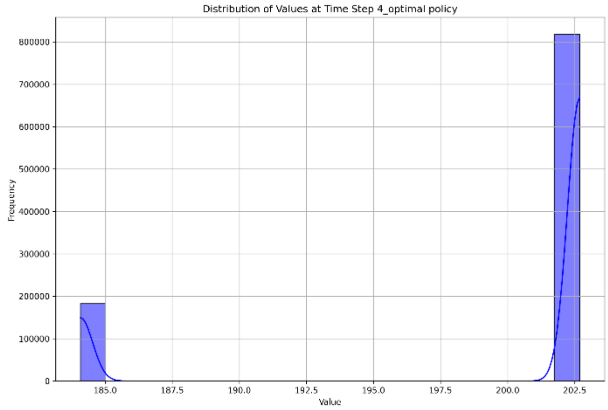


Median energy demand change with optimal policy from one million episodes

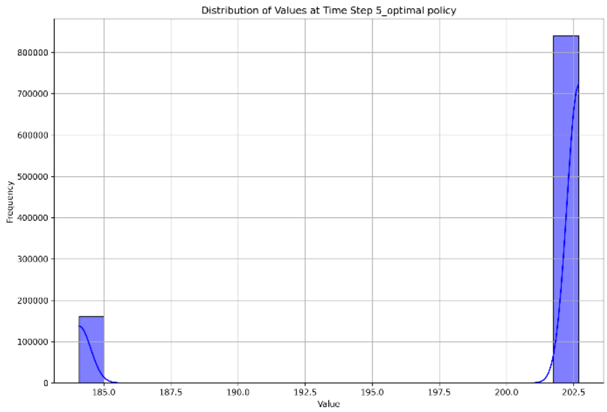


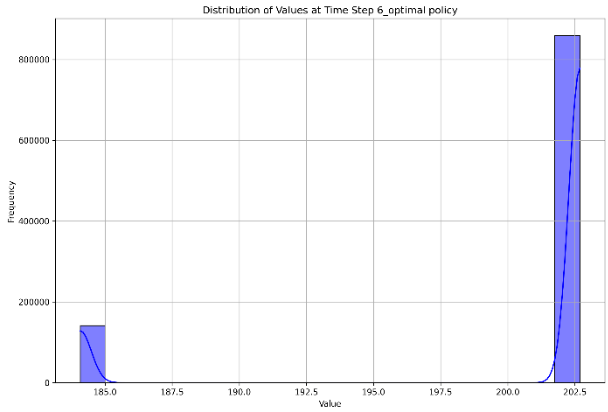
Distribution of possible energy demand at time step 1 over million episodes with optimal policy	Distribution of possible energy demand at time step 2 over million episodes with optimal policy
	
Distribution of possible energy demand at time step 3 over million episodes with optimal policy	Distribution of possible energy demand at time step 4 over million episodes with optimal policy



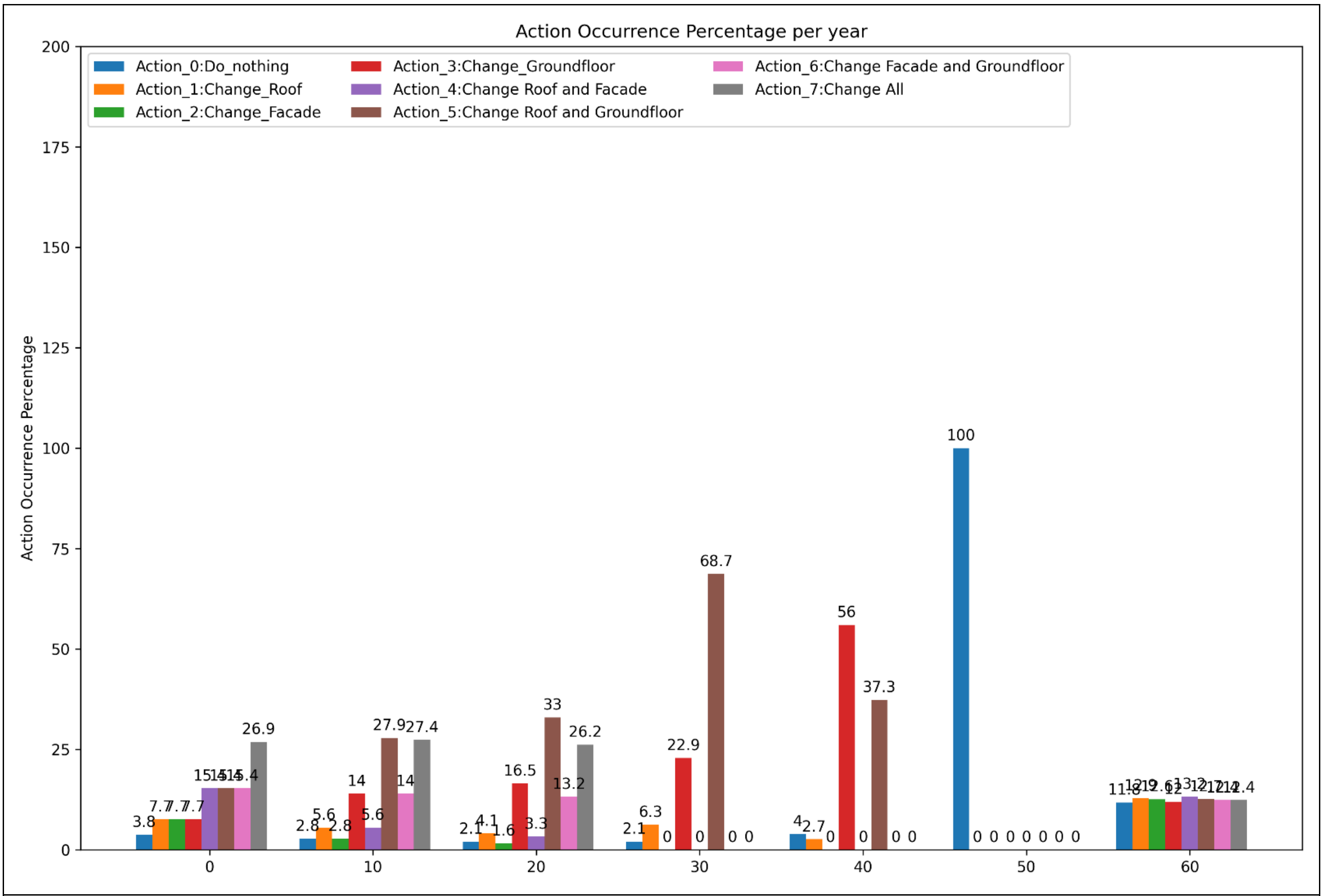


Distribution of possible energy demand at time step 5 over million episodes with optimal policy	Distribution of possible energy demand at time step 6 over million episodes with optimal policy

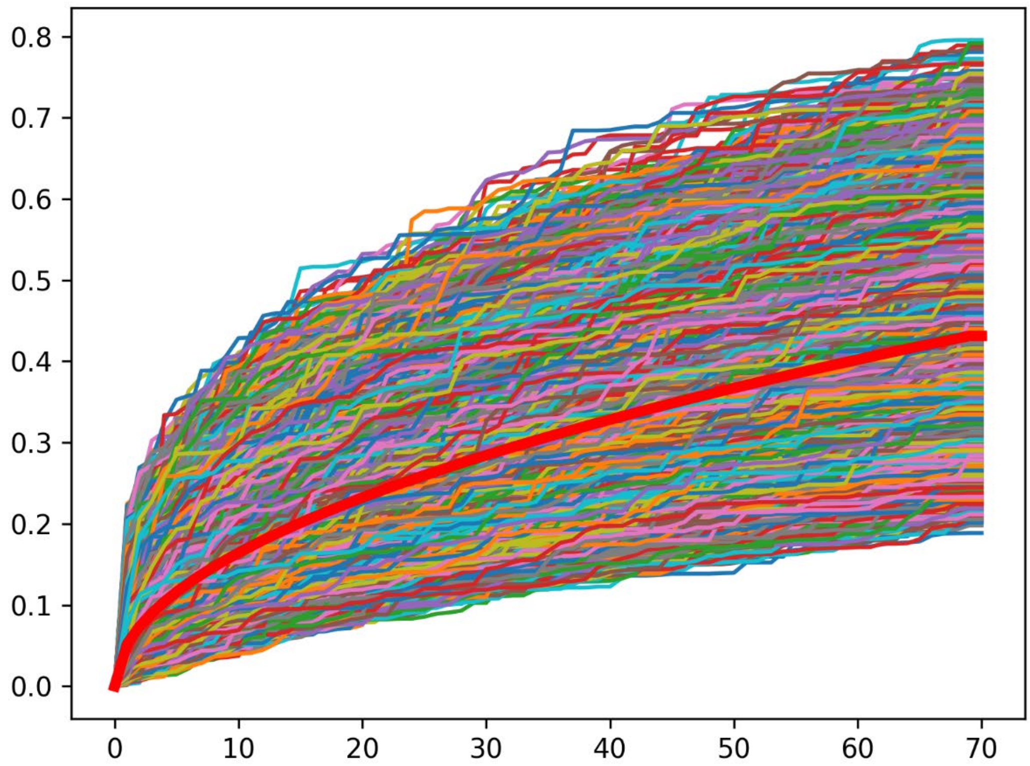




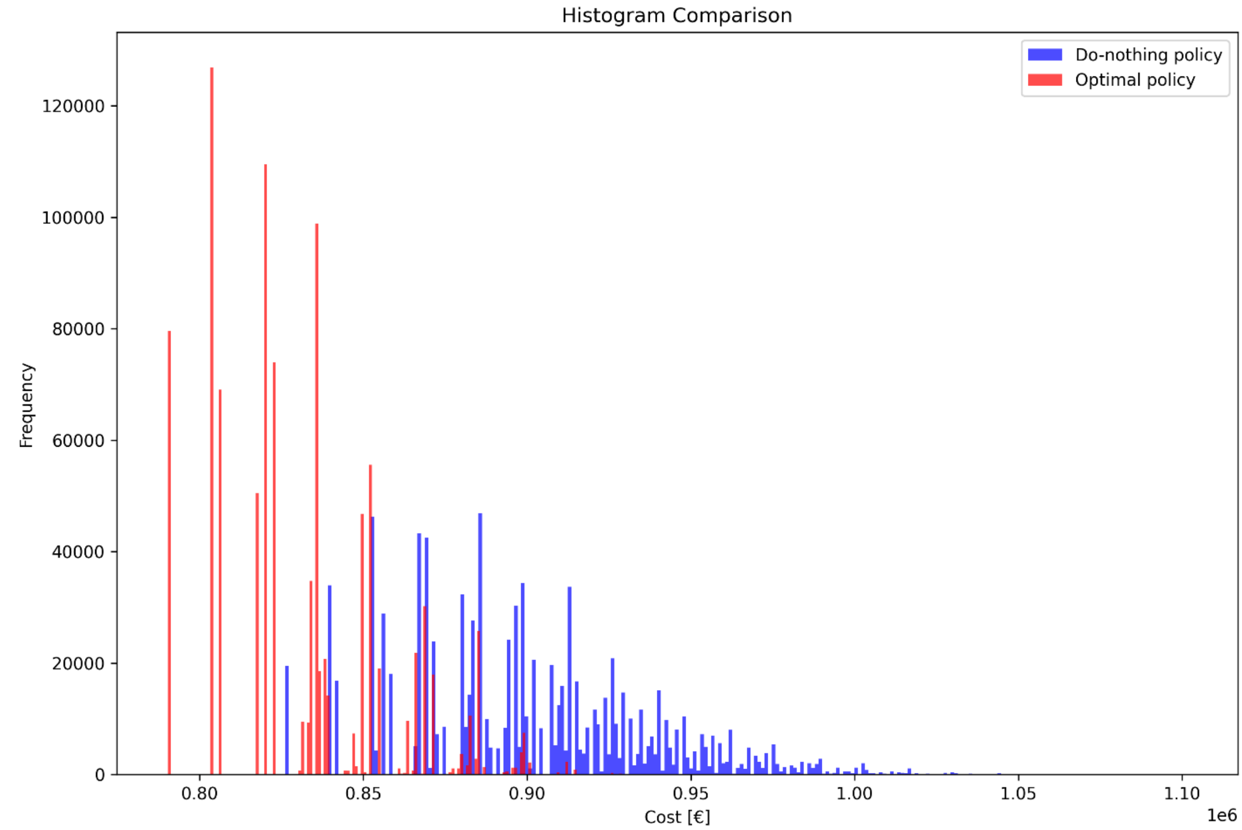
Simulation 6: No penalty, infiltration simulation, beta 0.5
Optimal policy plot

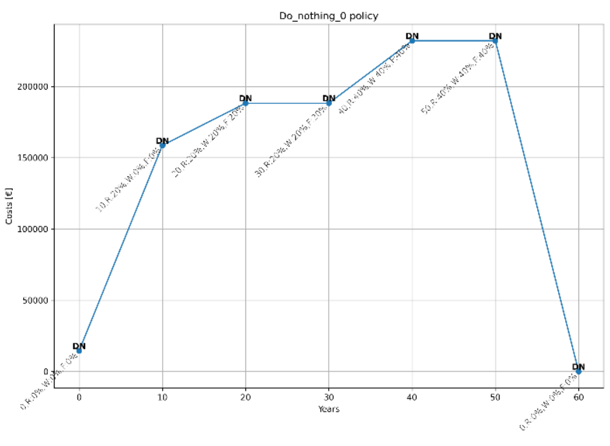
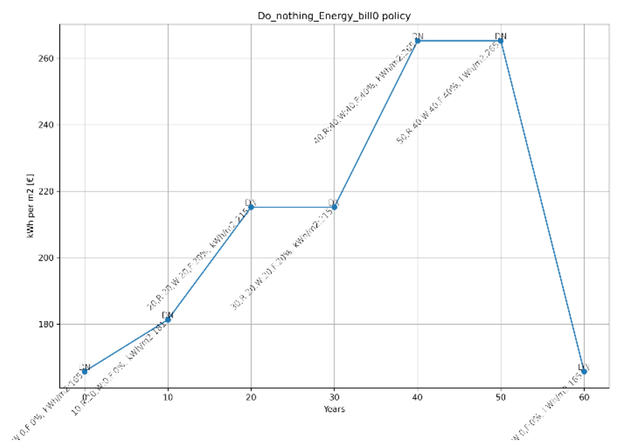


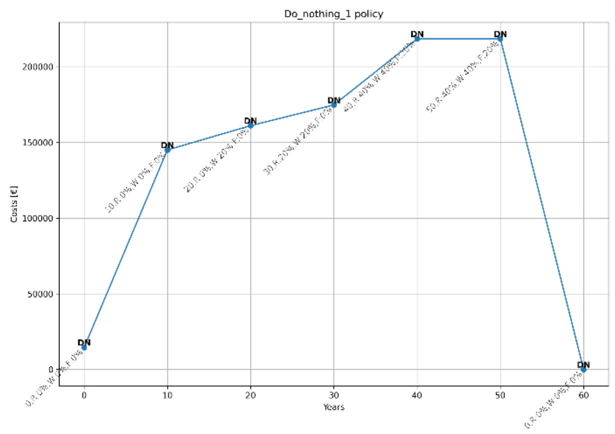
Material degradation scenarios , 1000000 realizations



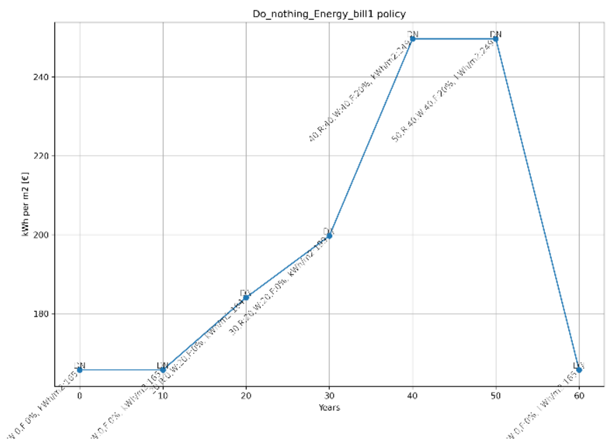
Histogram comparison of return between do nothing policy and optimal policy



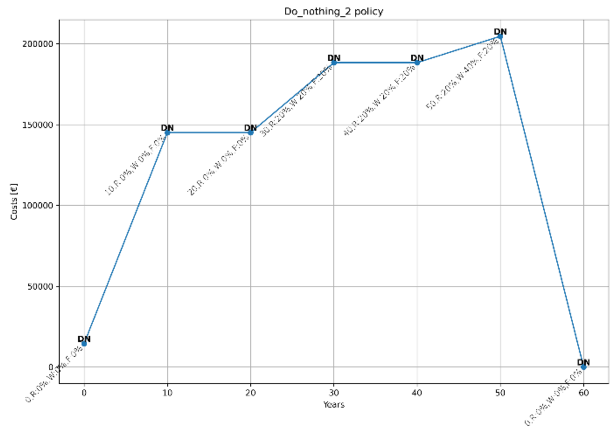
Do nothing policy: Sample Episode 1 Costs	Do nothing policy: Sample Episode 1 States of energy demand
	
Do nothing policy: Sample Episode 2 Costs	Do nothing policy: Sample Episode 2 States of energy demand



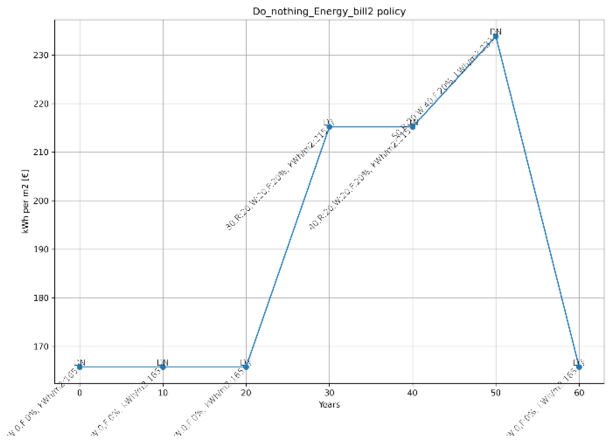
Do nothing policy: Sample Episode 3 Costs



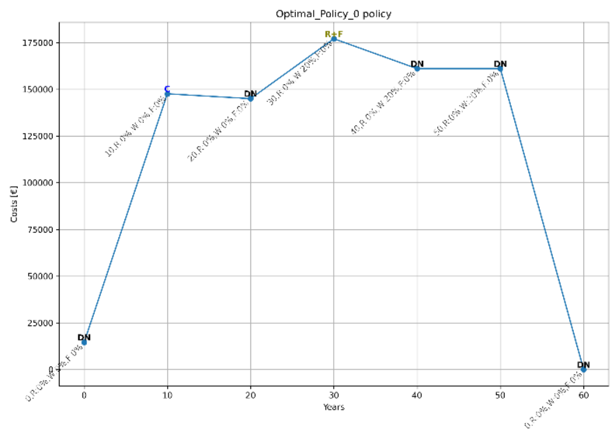
Do nothing policy: Sample Episode 3 States of energy demand



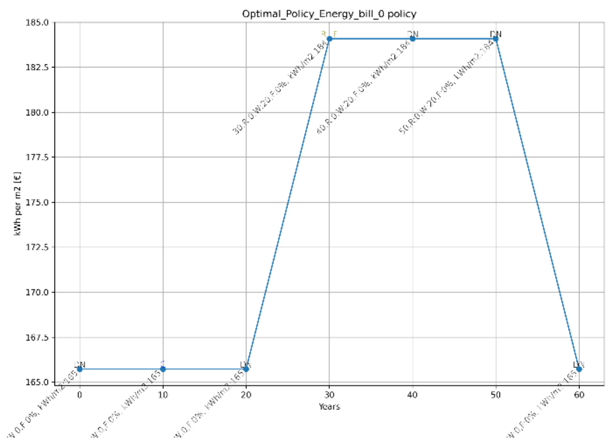
Optimal policy: Sample Episode 1 Costs



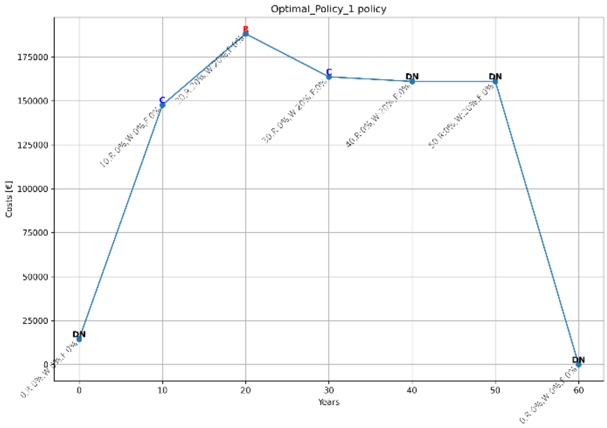
Optimal policy: Sample Episode 1 States of energy demand



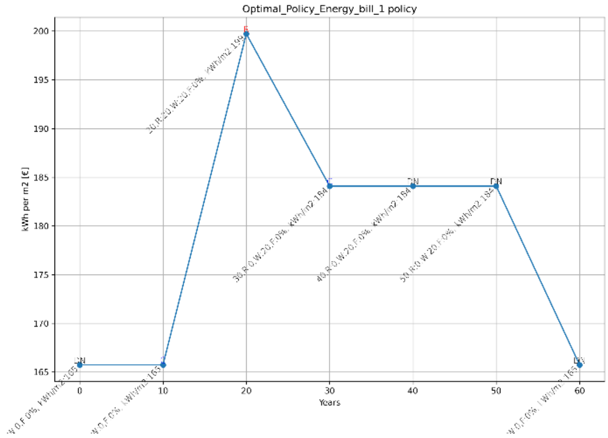
Optimal policy: Sample Episode 2 Costs



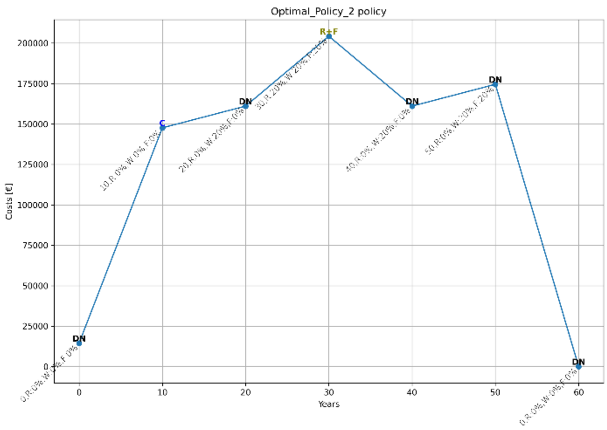
Optimal policy: Sample Episode 2 States of energy demand

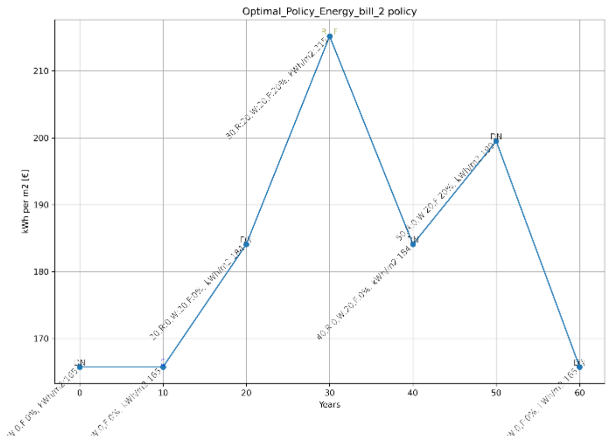


Optimal policy: Sample Episode 3 Costs

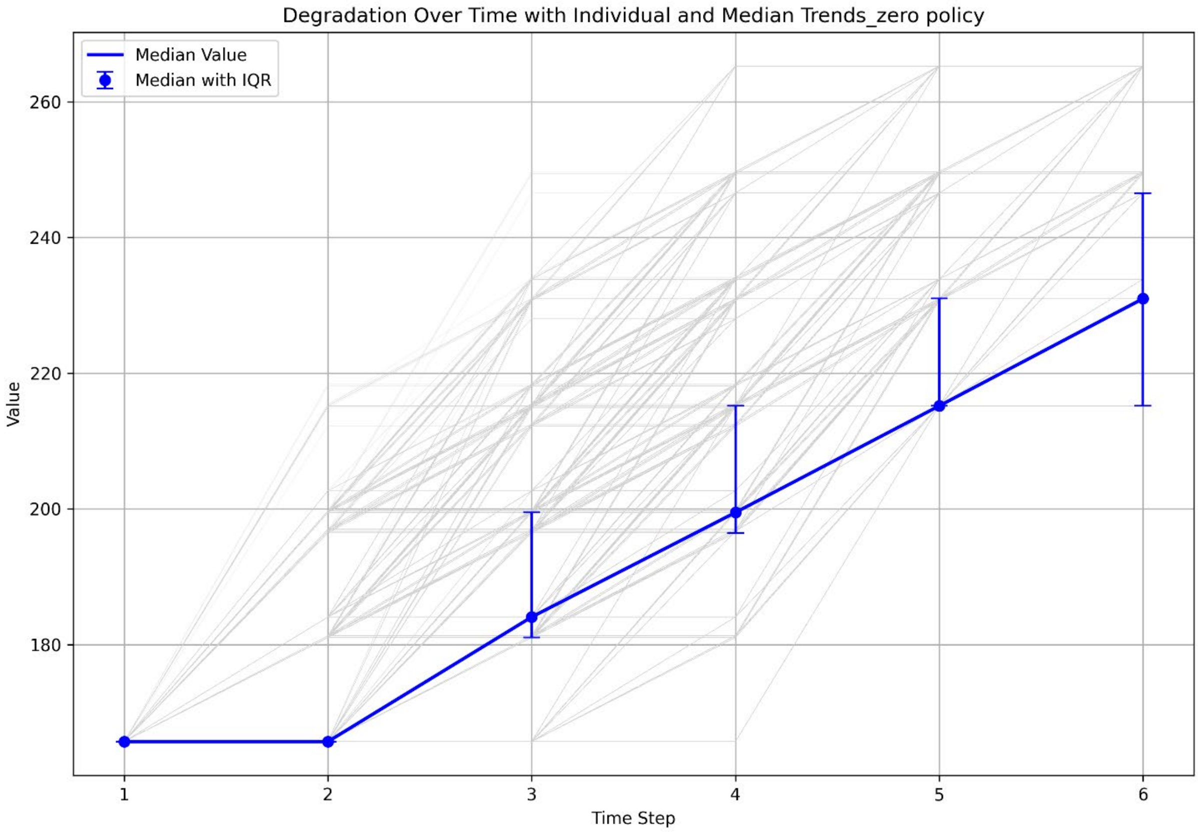


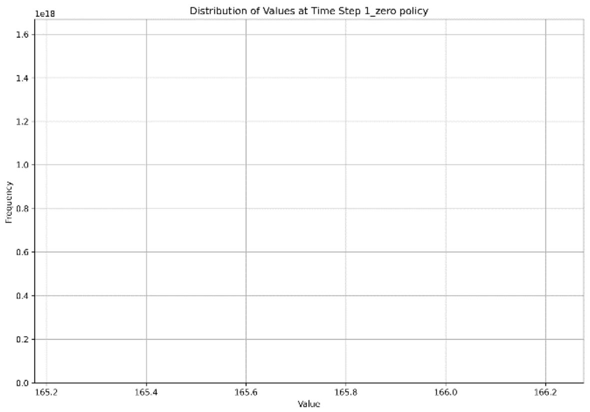
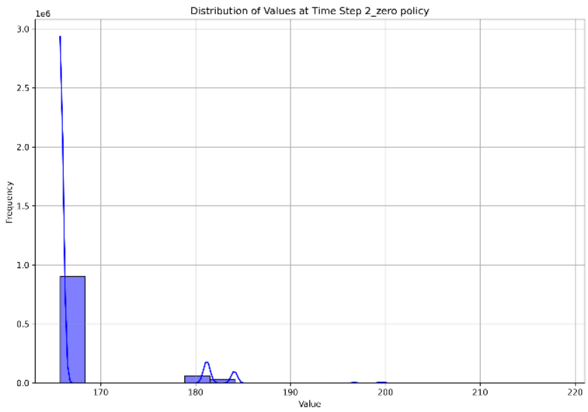
Optimal policy: Sample Episode 3 States of energy demand

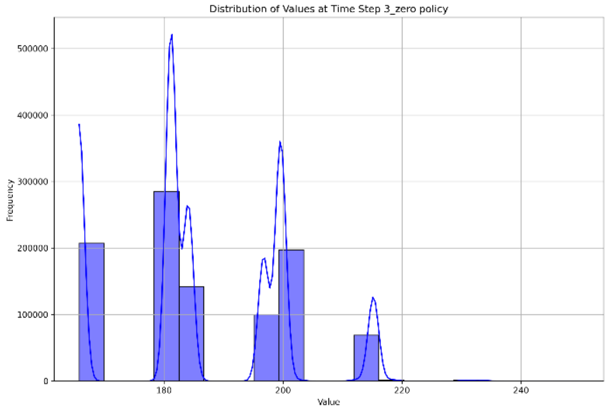
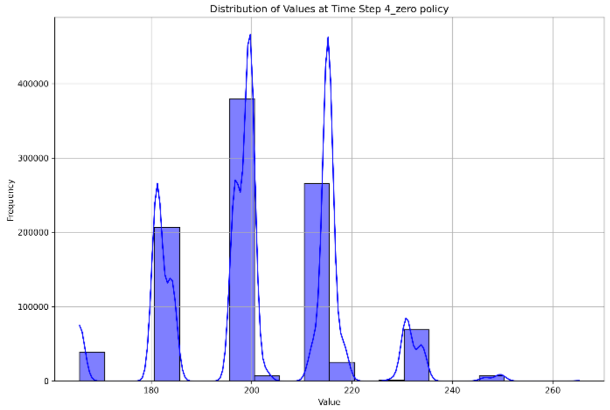
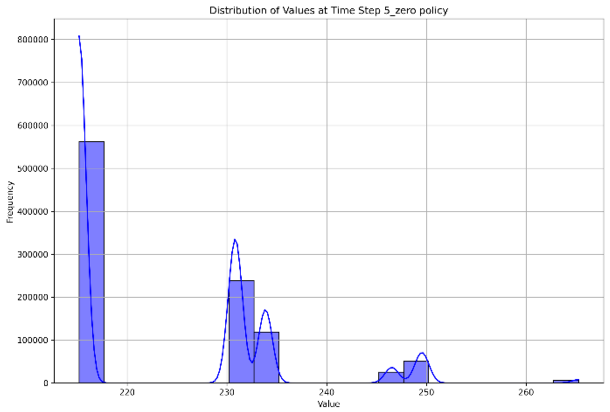
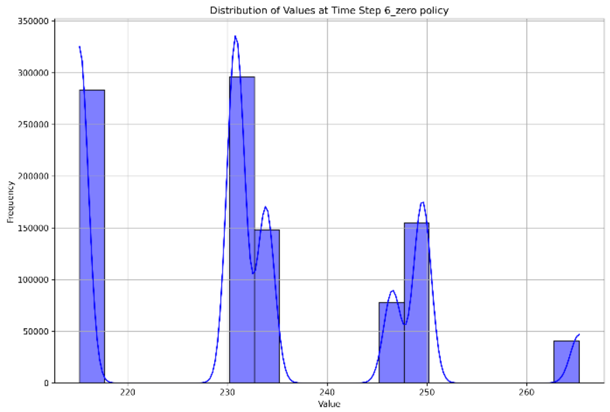


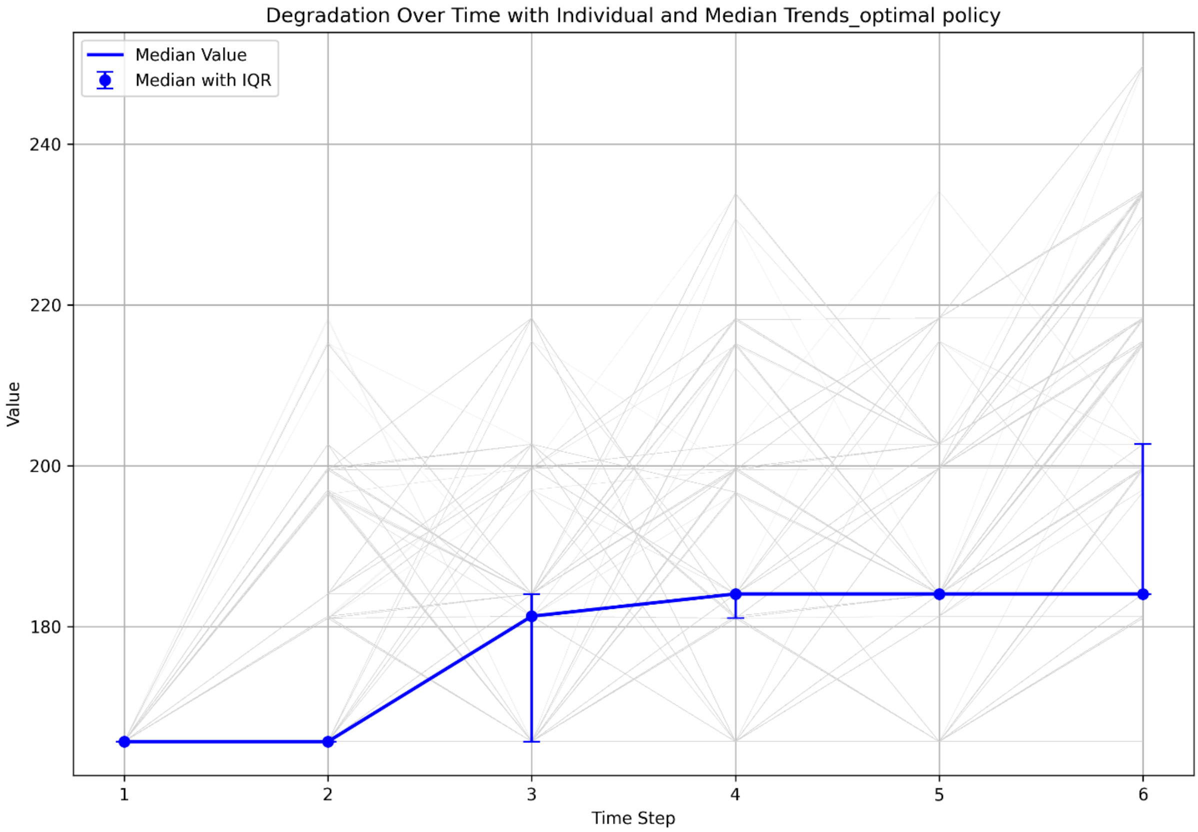


Median energy demand change with do nothing policy from one million episodes



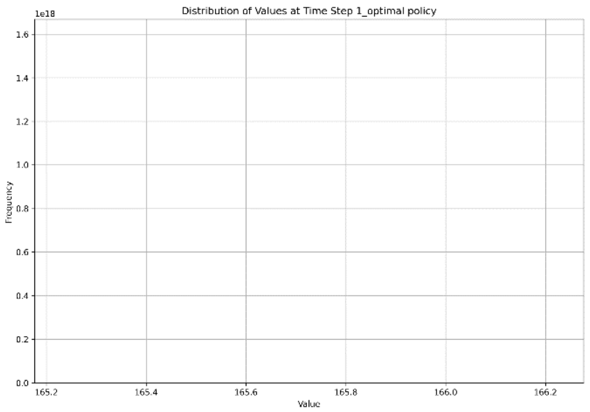
Distribution of possible energy demand at time step 1 over million episodes with do nothing policy	Distribution of possible energy demand at time step 2 over million episodes with do nothing policy
	
Distribution of possible energy demand at time step 3 over million episodes with do nothing policy	Distribution of possible energy demand at time step 4 over million episodes with do nothing policy

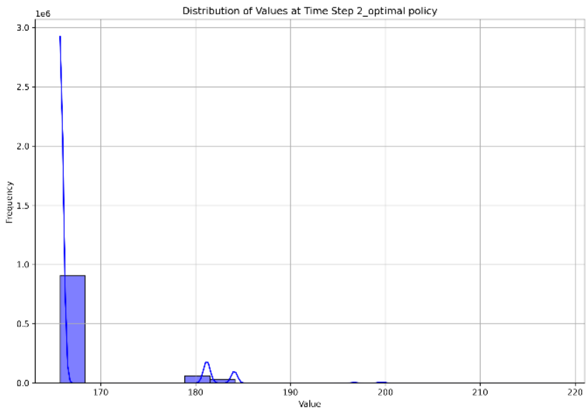
	
Distribution of possible energy demand at time step 5 over million episodes with do nothing policy	Distribution of possible energy demand at time step 6 over million episodes with do nothing policy
	
Median energy demand change with optimal policy from one million episodes	



Distribution of possible energy demand at time step 1 over million episodes with optimal policy

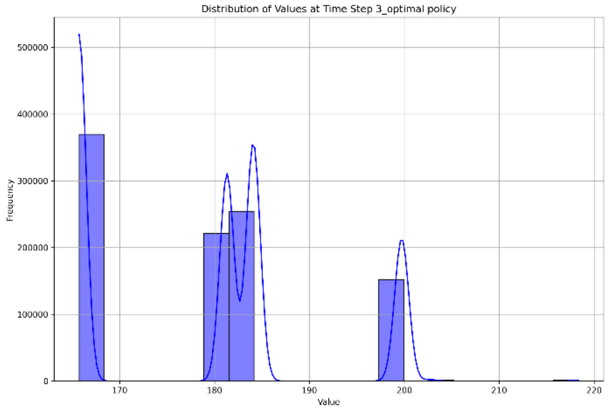
Distribution of possible energy demand at time step 2 over million episodes with optimal policy

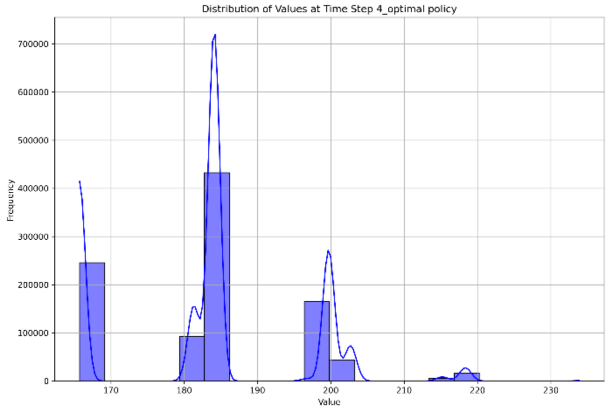




Distribution of possible energy demand at time step 3 over million episodes with optimal policy

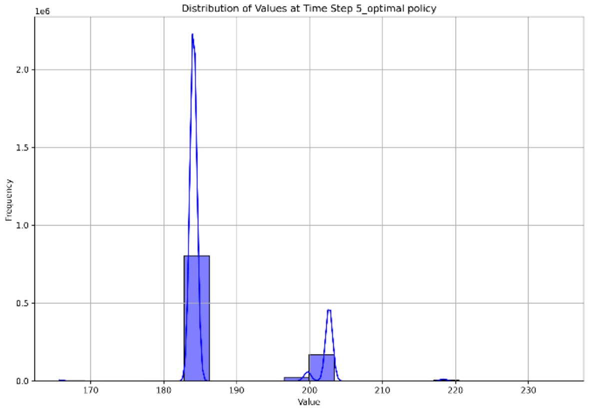
Distribution of possible energy demand at time step 4 over million episodes with optimal policy

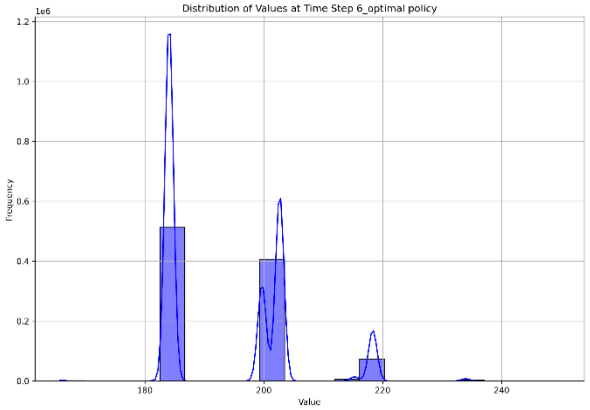




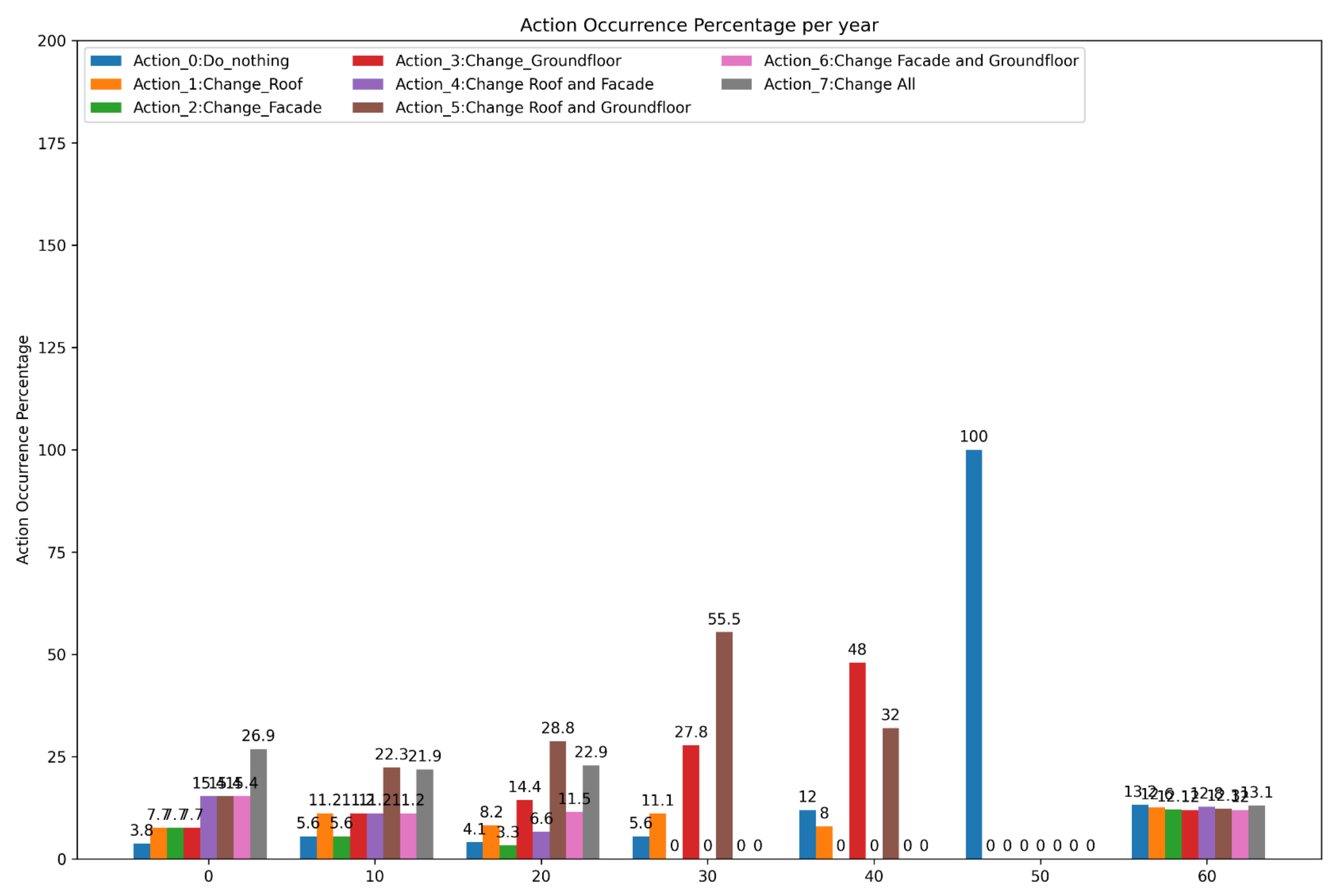
Distribution of possible energy demand at time step 5 over million episodes with optimal policy

Distribution of possible energy demand at time step 6 over million episodes with optimal policy

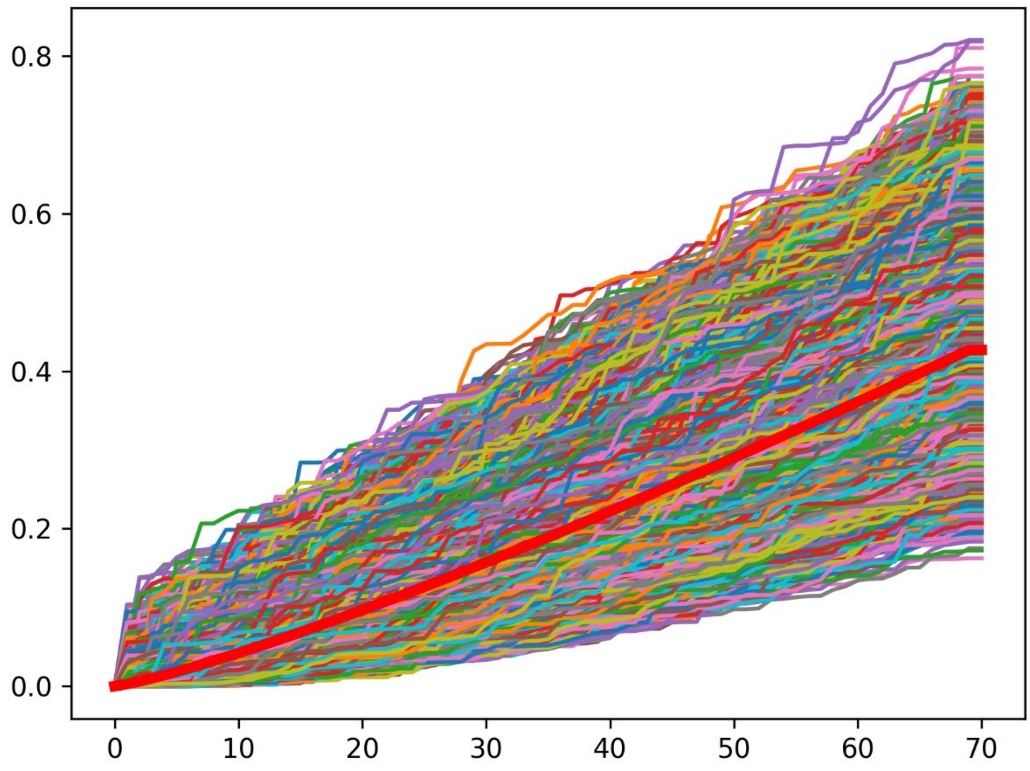




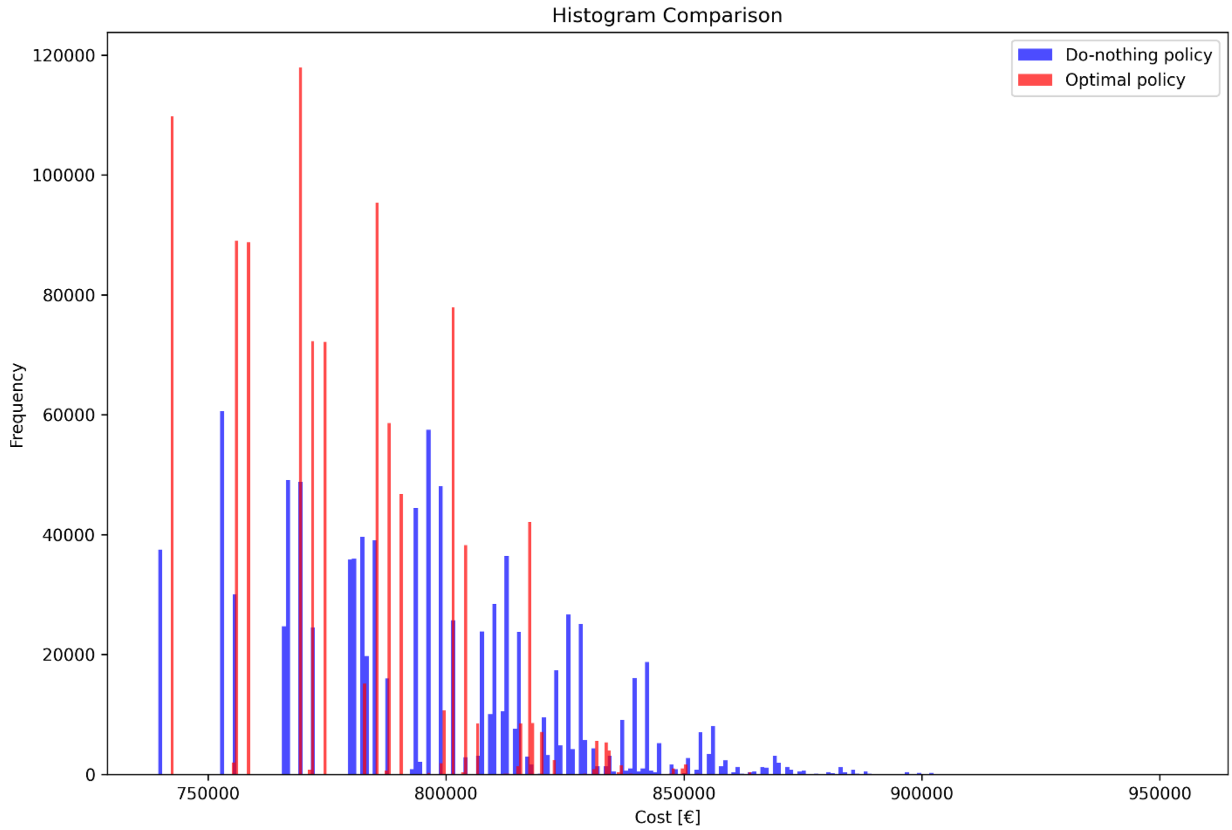
Simulation 7: No penalty, infiltration simulation, beta 1.2
Optimal policy plot

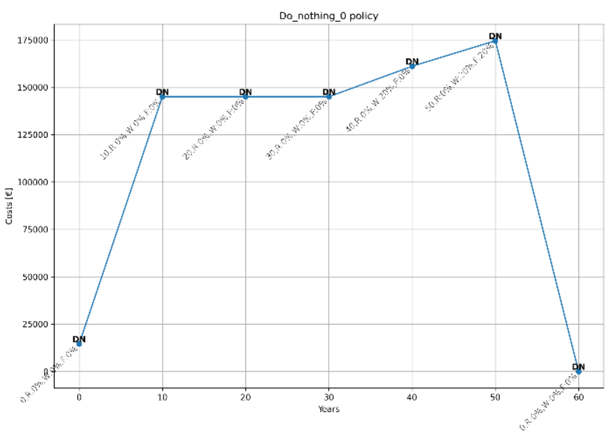
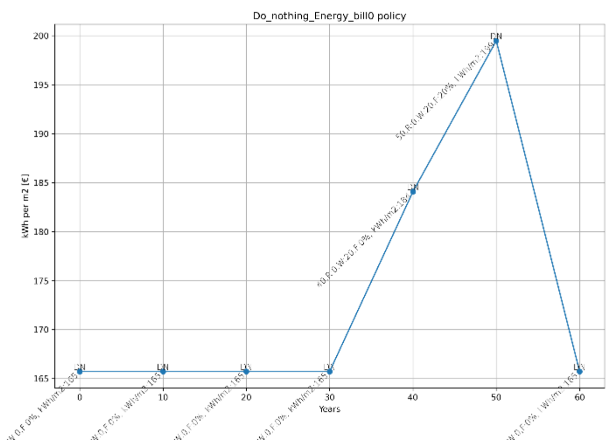


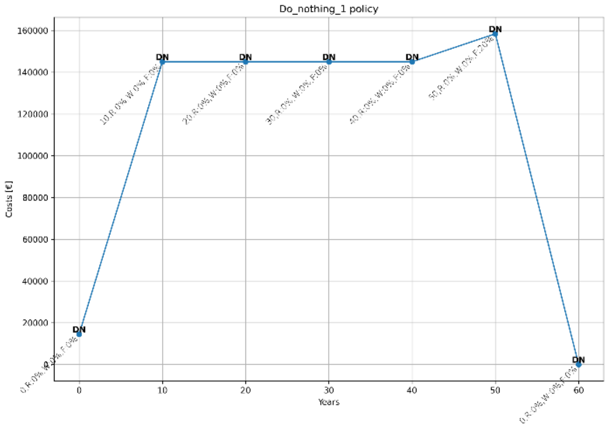
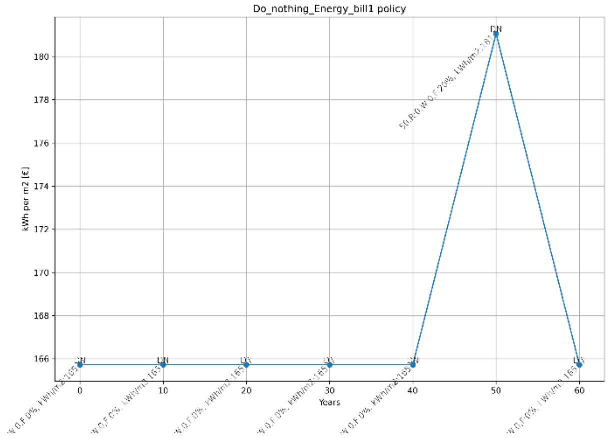
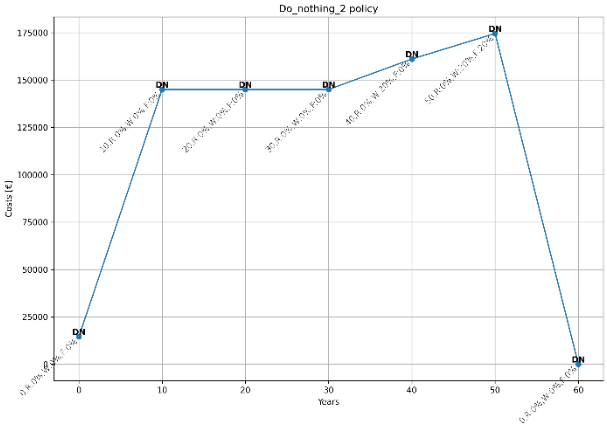
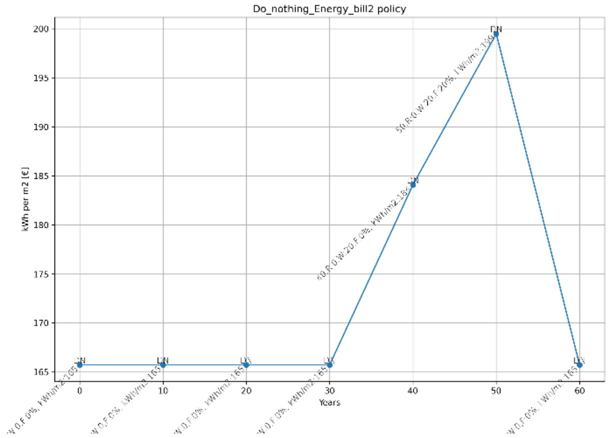
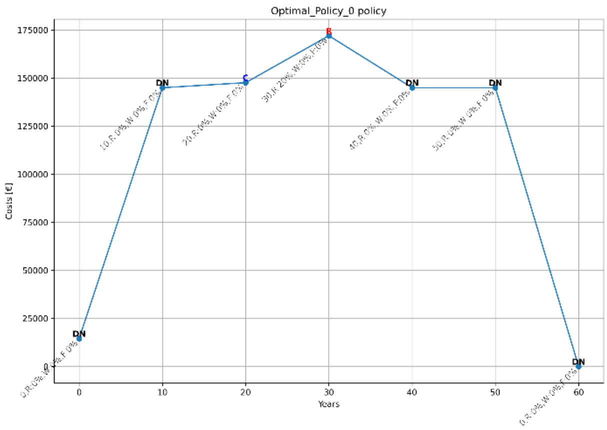
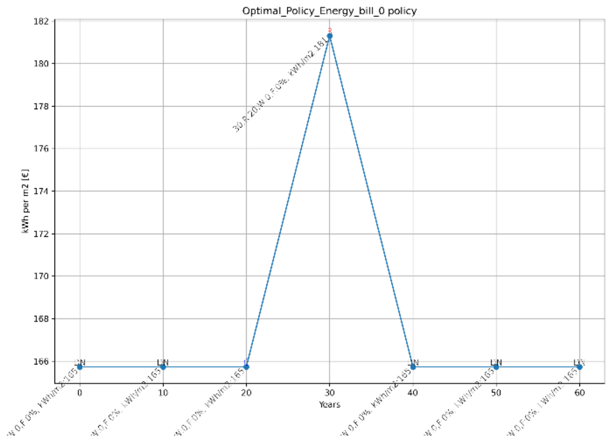
Material degradation scenarios , 1000000 realizations

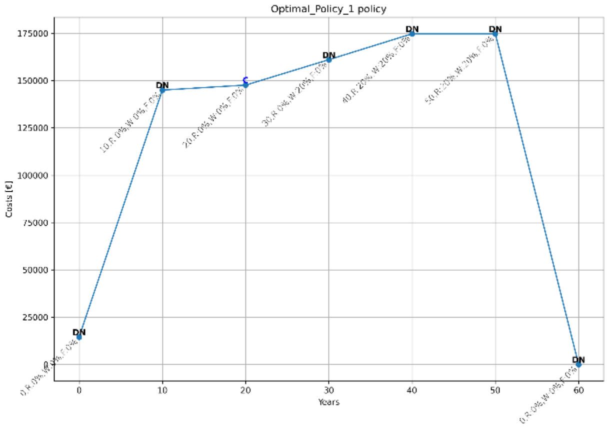


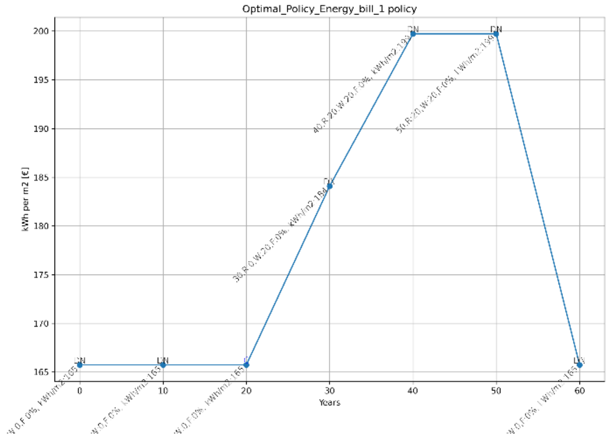
Histogram comparison of return between do nothing policy and optimal policy



Do nothing policy: Sample Episode 1 Costs	Do nothing policy: Sample Episode 1 States of energy demand
	
Do nothing policy: Sample Episode 2 Costs	Do nothing policy: Sample Episode 2 States of energy demand

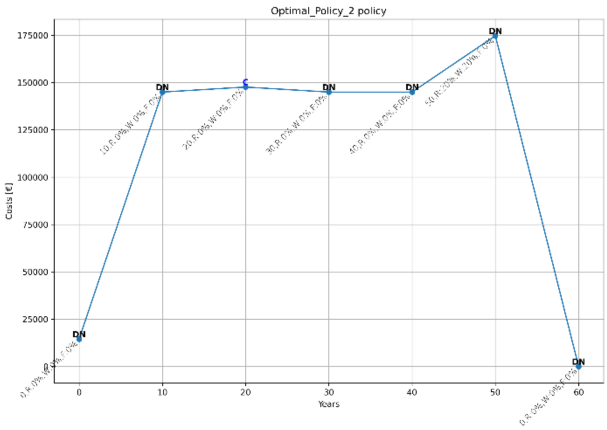
	
Do nothing policy: Sample Episode 3 Costs	Do nothing policy: Sample Episode 3 States of energy demand
	
Optimal policy: Sample Episode 1 Costs	Optimal policy: Sample Episode 1 States of energy demand
	
Optimal policy: Sample Episode 2 Costs	Optimal policy: Sample Episode 2 States of energy demand

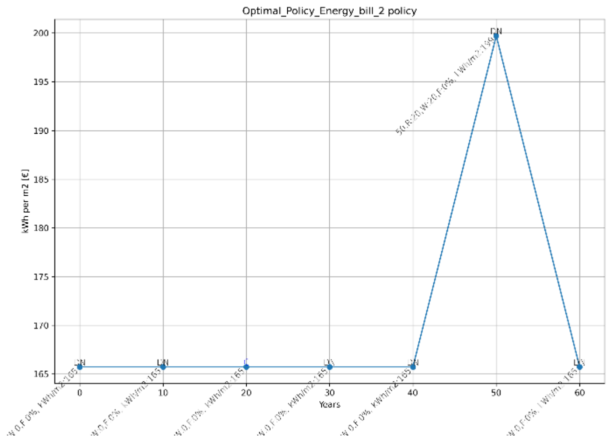




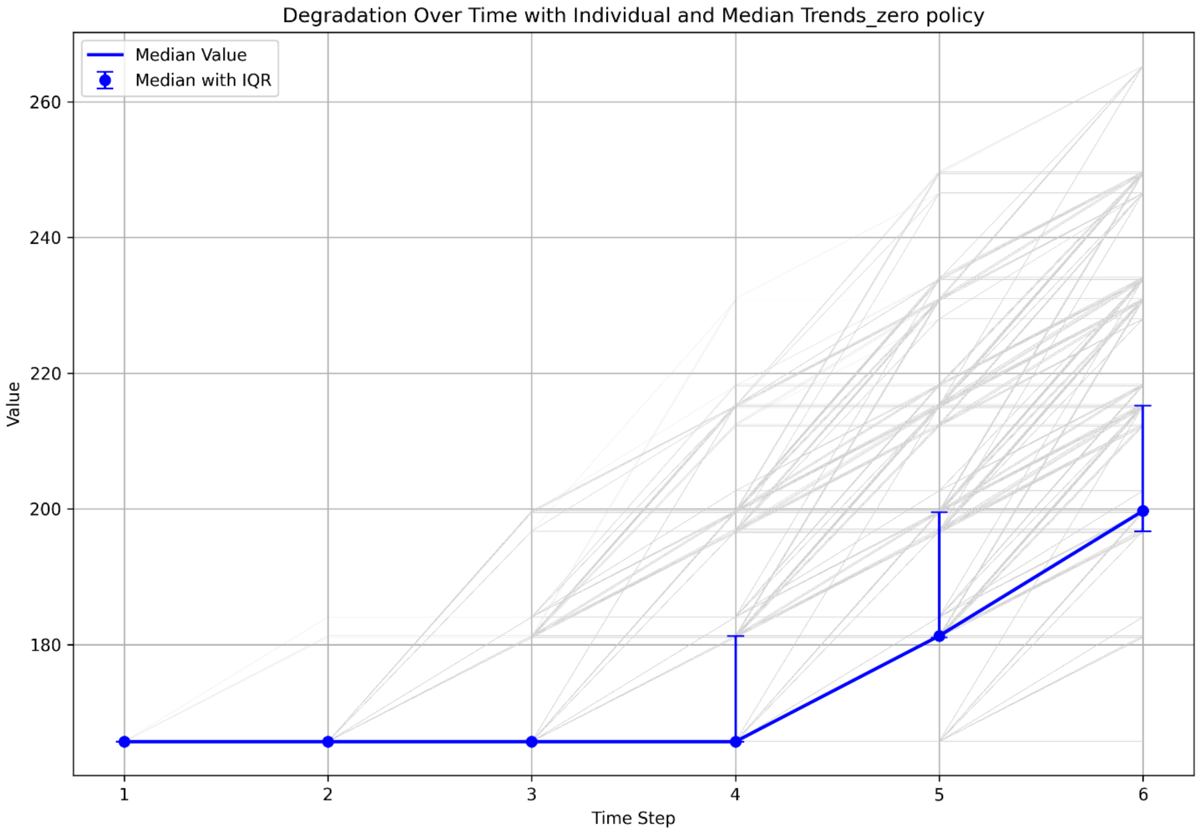
Optimal policy: Sample Episode 3 Costs

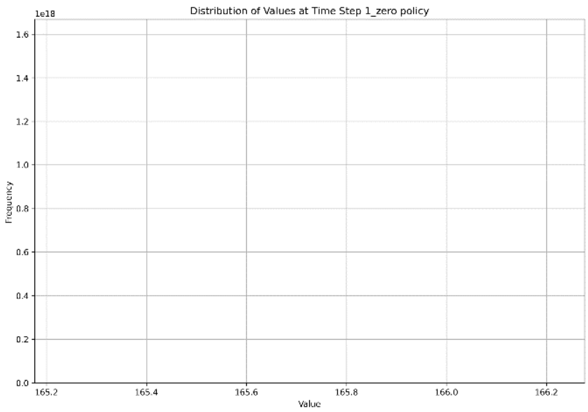
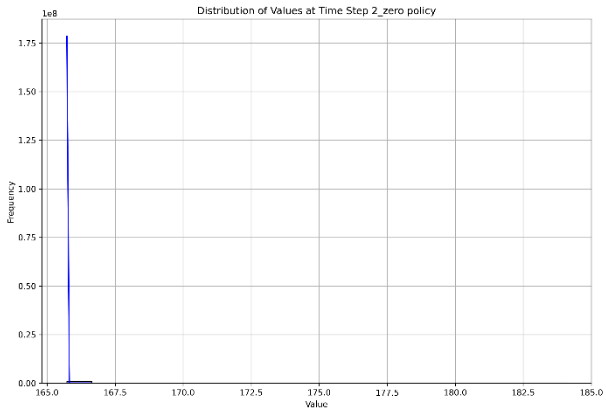
Optimal policy: Sample Episode 3 States of energy demand

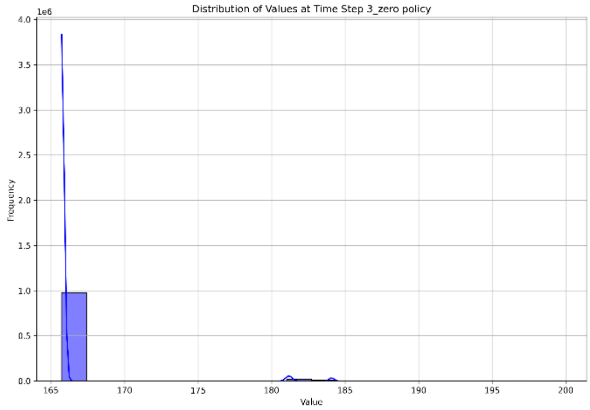
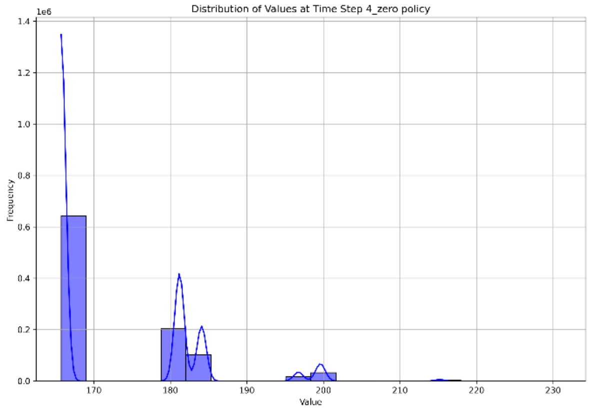
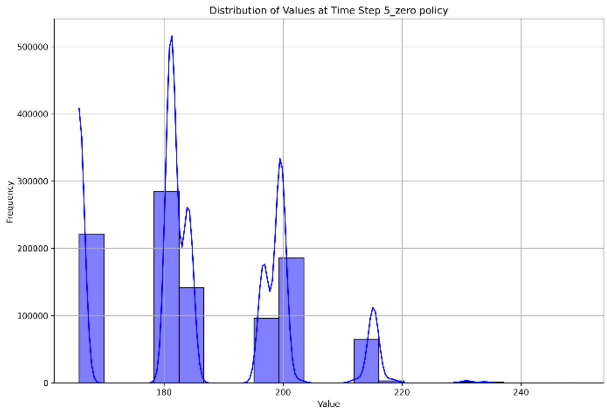
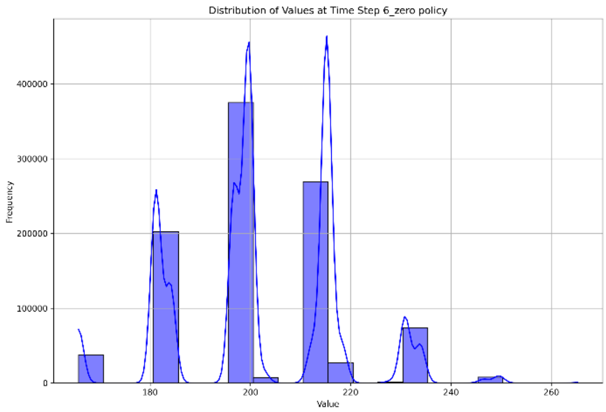


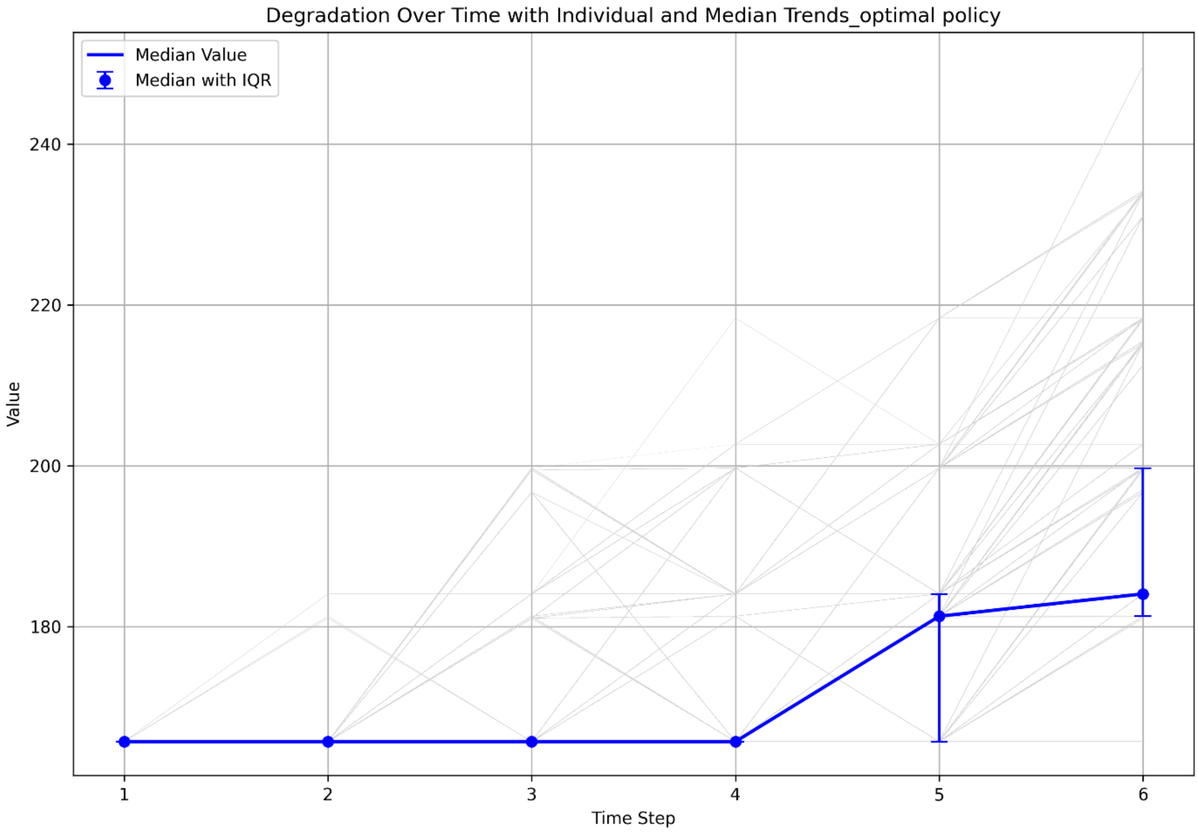


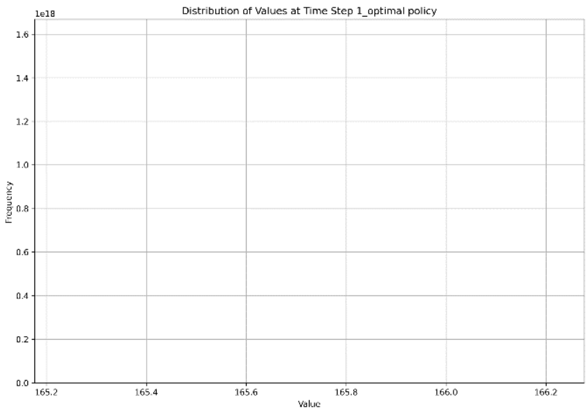
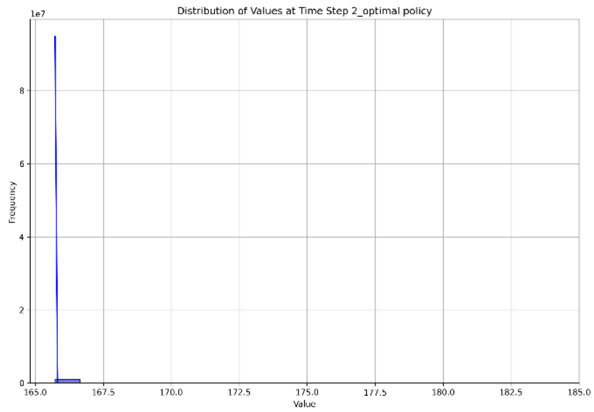
Median energy demand change with do nothing policy from one million episodes

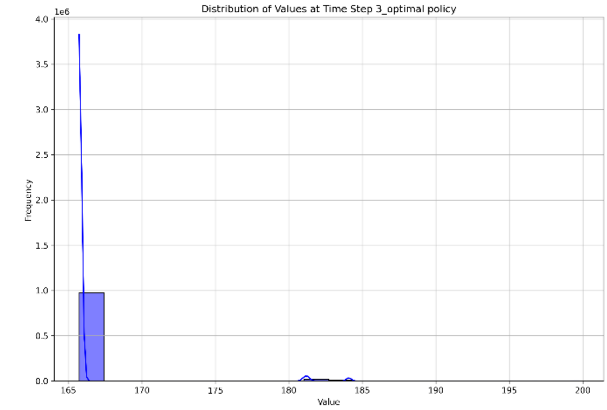


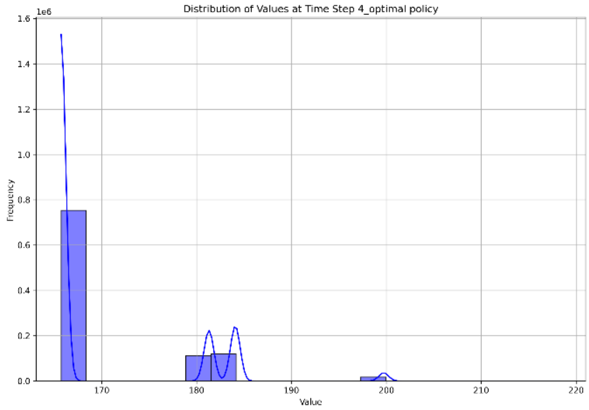
Distribution of possible energy demand at time step 1 over million episodes with do nothing policy	Distribution of possible energy demand at time step 2 over million episodes with do nothing policy
	
Distribution of possible energy demand at time step 3 over million episodes with do nothing policy	Distribution of possible energy demand at time step 4 over million episodes with do nothing policy

<table>
<tr>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>Distribution of possible energy demand at time step 5 over million episodes with do nothing policy</td>
<td>Distribution of possible energy demand at time step 6 over million episodes with do nothing policy</td>
</tr>
<tr>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td colspan="2">Median energy demand change with optimal policy from one million episodes</td>
</tr>
</table>



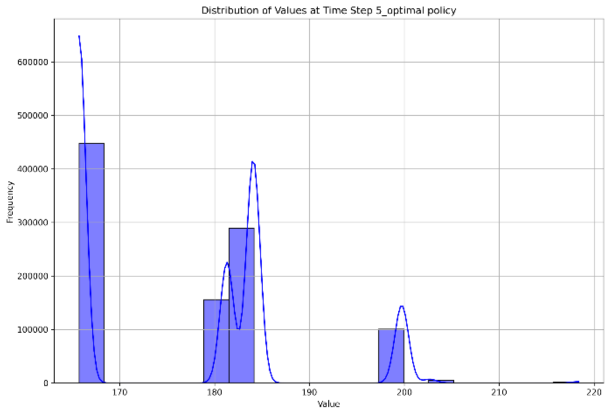
Distribution of possible energy demand at time step 1 over million episodes with optimal policy	Distribution of possible energy demand at time step 2 over million episodes with optimal policy
	
Distribution of possible energy demand at time step 3 over million episodes with optimal policy	Distribution of possible energy demand at time step 4 over million episodes with optimal policy

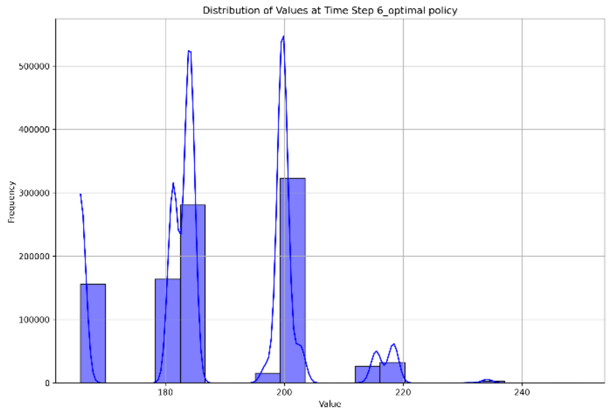


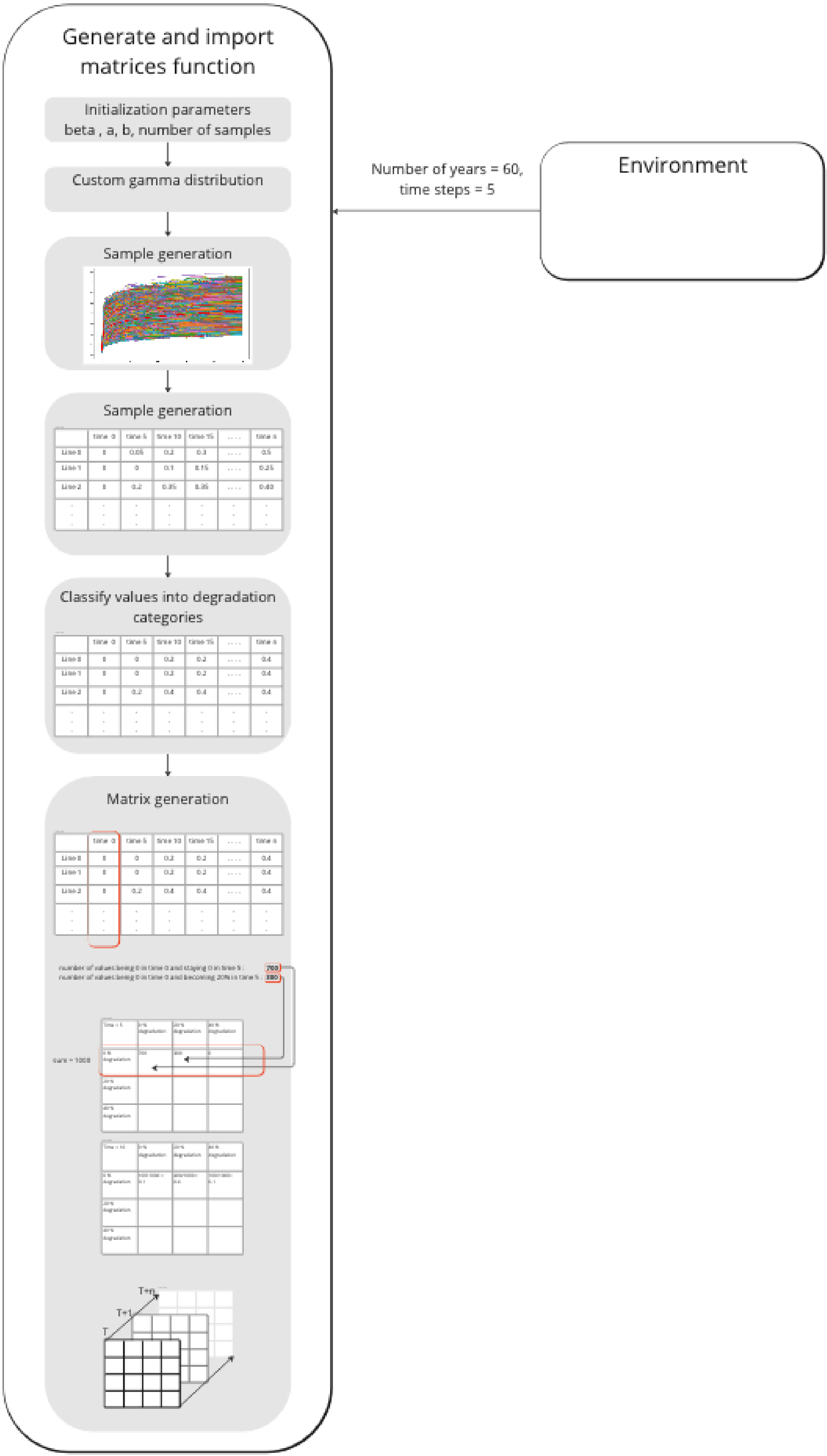


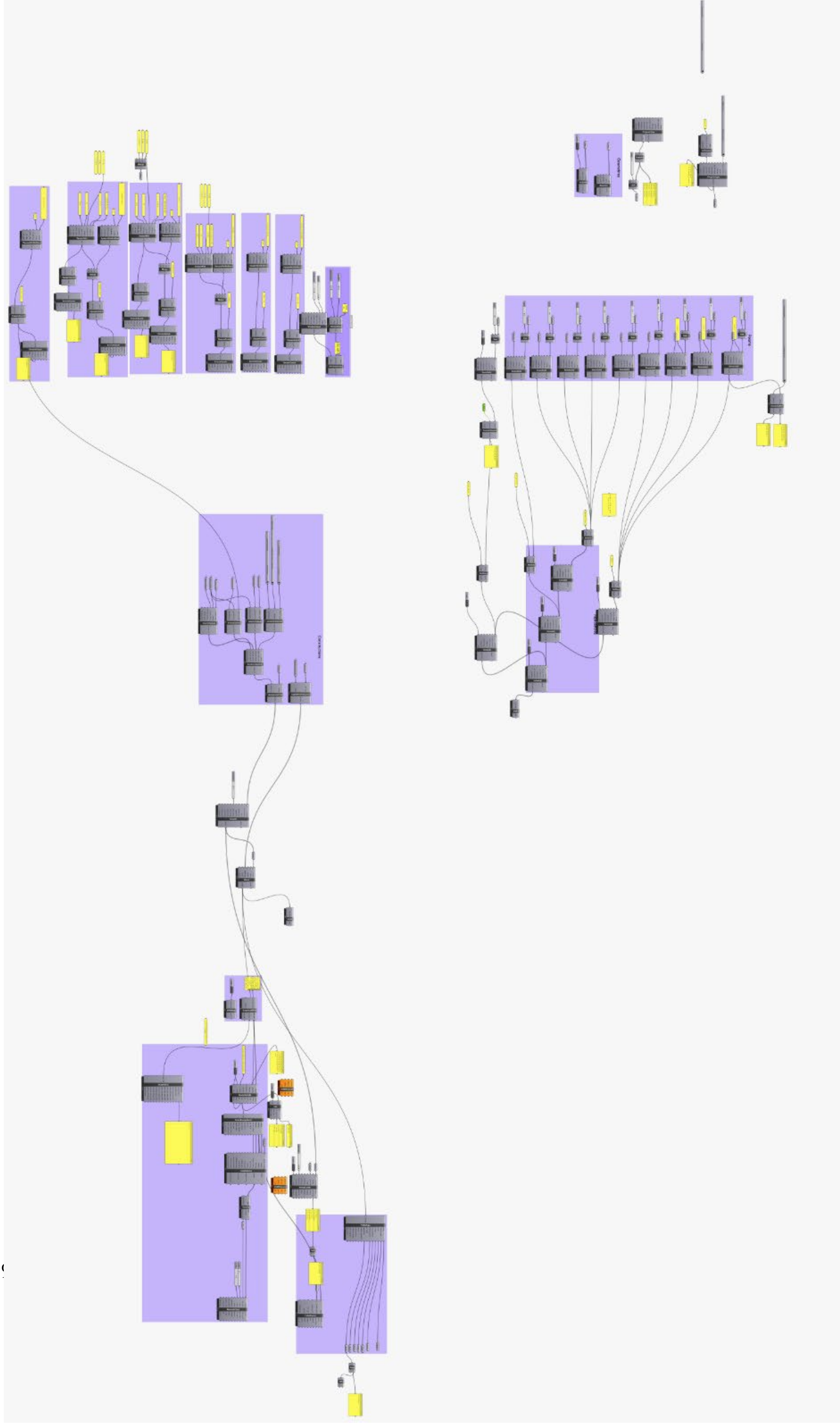
Distribution of possible energy demand at time step 5 over million episodes with optimal policy

Distribution of possible energy demand at time step 6 over million episodes with optimal policy






Generate and import matrices function
Initialization parameters
beta , a, b, number of samples
Custom gamma distribution
Sample generation
Sample generation
Classify values into degradation categories
Matrix generation
T+n
T+1
T
Number of years = 60,
time steps = 5
Environment



11. BIBLIOGRAPHY

Abbeel, P. (n.d.). *Markov Decision Processes and Exact Solution Methods: Value Iteration Policy Iteration Linear Programming*. Lecture Notes from UC Berkeley EECS .

Andriotis, C. P., & Papakonstantinou, K. G. (2018). *Managing engineering systems with large state and action spaces through deep reinforcement learning*.

Bae, M., Ahn, H., Kang, J., Choi, G., & Choi, H. (2022). *Determination of the Long-Term Thermal Performance of Foam Insulation Materials through Heat and Slicing Acceleration*. https://doi.org/10.3390/polym14224926

Bastian, Z., Blücher, M., Inhofer, P., Torres, F., Ahmed, S., & Lang, A. (2016). *Step by step retrofits with Passive House components*. https://europhit.eu/sites/europhit.eu/files/EuroPHit_Handbook_final_Optimized.pdf

Bhustali, P. (2023). Assessing the Optimality of Decentralized Inspection and Maintenance Policies for Stochastically Degrading Engineering Systems. *BNAIC/BeNeLearn 2023: Joint International Scientific Conferences on AI and Machine Learning,*.

Bouckaert, S., Pales, A. F., McGlade, C., Remme, U., Wanner, B., Varro, L., D'Ambrosio, D., & Spencer, T. (2021). *Net Zero by 2050: A Roadmap for the Global Energy Sector*.

British Standards Institution., & British Standards Institution. (1999). *Ageing of thermal insulation materials : determination of the long-term change in thermal resistance of closed-cell plastics (accelerated laboratory test methods)*. British Standards Institution.

Choi, H.-J., Kang, J.-S., & Huh, J.-H. (2018). A Study on Variation of Thermal Characteristics of Insulation Materials for Buildings According to Actual Long-Term Annual Aging Variation. *Int J Thermophys*, *39*(2). https://doi.org/10.1007/s10765-017-2318-3

Deep Variational Inference. Studying Variational Inference using DL… | by Natan Katz | Towards Data Science. (n.d.). Retrieved June 19, 2024, from https://towardsdatascience.com/deep-variational-inference-2a7e43244723

Delivering the European Green Deal - European Commission. (n.d.). Retrieved March 25, 2024, from https://commission.europa.eu/strategy-and-policy/priorities-2019-2024/european-green-deal/delivering-european-green-deal_en

Dong, H., Ding, Z., & Zhang, S. (2020). Deep reinforcement learning: Fundamentals, research and applications. *Deep Reinforcement Learning: Fundamentals, Research and Applications*, 1–514. https://doi.org/10.1007/978-981-15-4095-0/COVER

Duan, Y., Ke, Z. T., & Wang, M. (n.d.). *State Aggregation Learning from Markov Transition Data*.

Eleftheriadis, G., & Hamdy, M. (2018). *The Impact of Insulation and HVAC Degradation on Overall Building Energy Performance: A Case Study*. https://doi.org/10.3390/buildings8020023

Energy Performance of Buildings Directive. (n.d.). Retrieved March 25, 2024, from https://energy.ec.europa.eu/topics/energy-efficiency/energy-efficient-buildings/energy-performance-buildings-directive_en

Energy Saving Trust. (2015). *Trigger points: a convenient truth Promoting energy efficiency in the home*. https://www.yumpu.com/en/document/view/44329336/trigger-points-energy-saving-trust

Esser Anne, Dunne Allison, Meeusen Tim, Quaschning Simon, & Wegge Denis. (2019). *Comprehensive study of building energy renovation activities and the uptake of nearly zero-energy buildings in the EU Final report*. www.navigant.com

Fabbri, M., De Groote, M., & Rapf, O. (2016). *Building Renovation Passports - Customised roadmaps towards deep renovation and better homes*. https://www.bpie.eu/publication/renovation-passports/

Faqih, F., Zayed, T., & Soliman, · Ehab. (2020). *Factors and defects analysis of physical and environmental condition of buildings*. *5*, 19. https://doi.org/10.1007/s41024-020-00084-0

Ferreira, C., Silva, A., de Brito, J., & Flores-Colen, I. (2023). Maintainability of Buildings' Envelope. In *Springer Series in Reliability Engineering* (pp. 63–115). Springer Science and Business Media Deutschland GmbH. https://doi.org/10.1007/978-3-031-14767-8_4

Foxon, T. J. (n.d.). Technological and institutional "lock-in" as a barrier to sustainable innovation. *2002*. Retrieved January 19, 2024, from http://www.iccept.ic.ac.uk/public.html

Fritz, S., Pehnt, M., Mellwig, P., & Volt, J. (2019). *Planned staged deep renovations as the main driver for a decarbonised European building stock*.

Honeybee "ENERGY" Part 1 | Setting the Scene | Ladybug tools 1.4.0 - YouTube. (n.d.). Retrieved May 17, 2024, from https://www.youtube.com/watch?v=CoH9I4oZc_k

Hung Anh, L. D., & Pásztory, Z. (2021). An overview of factors influencing thermal conductivity of building insulation materials. In *Journal of Building Engineering* (Vol. 44). Elsevier Ltd. https://doi.org/10.1016/j.jobe.2021.102604

iBROAD - European Commission. (n.d.). Retrieved January 19, 2024, from https://cinea.ec.europa.eu/featured-projects/ibroad_en

Jan Steiger, & Eva Vahalova. (2019). *Overall retrofit plan for step-by-step retrofits to EnerPHit Standard [Passipedia EN]*. https://passipedia.org/planning/refurbishment_with_passive_house_components/overall_retrofit_plan_for_step-by-step_retrofits_to_enerphit_standard#sequential_retrofitting_of_components

Jansen, S., Mohammadi, S., & Bokel, R. (2021). Developing a locally balanced energy system for an existing neighbourhood, using the 'Smart Urban Isle' approach. *Sustainable Cities and Society*, *64*. https://doi.org/10.1016/J.SCS.2020.102496

Kantawala, N. (n.d.). *LT-Set: A Surrogate Model-Based Decision Tool for Low-Temperature District Heating Refurbishment.*

Konstantinou, T. (2014). *Façade Refurbishment Toolbox Supporting the Design of Residential Energy Upgrades.*

Kostenkentallen | RVO. (n.d.). Retrieved May 17, 2024, from https://kostenkentallen.rvo.nl/

Krachtopoulos, K. (2023). *Multi-objective Deep Reinforcement Learning for predictive maintenance of road networks.*

Littman, M. L. (1996). *Algorithms for Sequential Decision Making* [Ph.D. Dissertation, Brown University]. https://www.researchgate.net/publication/33697270

Lopes, J., Oliveira, R. A. F., Banaitiene, N., Banaitis, A., Arteconi, A., & Matas, J. (2021). *A Staged Approach for Energy Retrofitting an Old Service Building: A Cost-Optimal Assessment.* https://doi.org/10.3390/en14216929

Maia, I., Harringer, D., & Kranzl, L. (2023). *Staged renovation and the time-perspective: Which other metric should be used to assess climate-optimality of renovation activities?* https://doi.org/10.1016/j.segy.2023.100110

Maia, I., Kranzl, L., & Müller, A. (2021). New step-by-step retrofitting model for delivering optimum timing. *Applied Energy*, *290*, 116714. https://doi.org/10.1016/J.APENERGY.2021.116714

Markov Decision Process Explained | Built In. (n.d.). Retrieved June 19, 2024, from https://builtin.com/machine-learning/markov-decision-process

Mavromatidis, G., & Petkov, I. (2021). *MANGO: A novel optimization model for the long-term, multi-stage planning of decentralized multi-energy systems.* https://doi.org/10.1016/j.apenergy.2021.116585

Ministerie van Binnenlandse Zaken en Koninkrijksrelaties. (2022). *Voorbeeldwoningen 2022 bestaande bouw.*

Morato, P. G., Papakonstantinou, K. G., Andriotis, C. P., Nielsen, J. S., & Rigo, P. (n.d.). *Optimal Inspection and Maintenance Planning for Deteriorating Structural Components through Dynamic Bayesian Networks and Markov Decision Processes.*

Netherlands electricity prices, September 2023 | GlobalPetrolPrices.com. (n.d.). Retrieved May 11, 2024, from https://www.globalpetrolprices.com/Netherlands/electricity_prices/

New Energy Efficiency Directive published - European Commission. (n.d.). Retrieved March 25, 2024, from https://energy.ec.europa.eu/news/new-energy-efficiency-directive-published-2023-09-20_en

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. https://doi.org/10.1016/j.jmsy.2023.07.014

Passive House Institute. (n.d.). Retrieved January 14, 2024, from https://passivehouse.com/01_passivehouseinstitute/01_passivehouseinstitute.htm

Pavel, C. C., & Blagoeva, D. T. (n.d.). *Competitive landscape of the EU's insulation materials industry for energy-efficient buildings Revised edition*. https://doi.org/10.2760/750646

Ravichandiran, Sudharsan. (2018). *Hands-on reinforcement learning with Python : master reinforcement and deep reinforcement learning using OpenAI Gym and TensorFlow*. Packt Publishing.

Saifullah, M., Papakonstantinou, K. G., Andriotis, C. P., & Stoffels, S. M. (n.d.). *Multi-agent deep reinforcement learning with centralized training and decentralized execution for transportation infrastructure management A R T I C L E I N F O*.

scipy.stats.gamma — SciPy v1.13.1 Manual. (n.d.). Retrieved June 19, 2024, from https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.gamma.html

Sesana, M. M., Salvalai, G., Brutti, D., Mandin, C., & Wei, W. (2020). ALDREN: A Methodological Framework to Support Decision-Making and Investments in Deep Energy Renovation of Non-Residential Buildings. *Buildings 2021, Vol. 11, Page 3*, *11*(1), 3. https://doi.org/10.3390/BUILDINGS11010003

Sewnath, M. (2024). *Title ' 'Towards Zero Carbon: A Comprehensive Evaluation of Conventional Renovation Strategies for Terraced Houses, Using Life Cycle Analysis (LCA) and Life Cycle Costing (LCC) to Enhance Decision-Making Support-accompa-nied by the design of a tool Personal details*. Technical University of Delft .

Sutton, R. S., & Barto, A. G. (2018). *Reinforcement learning : an introduction* (2018th ed.).

Taki, A., & Zakharanka, A. (2023). The Effect of Degradation on Cold Climate Building Energy Performance: A Comparison with Hot Climate Buildings. *Sustainability 2023, Vol. 15, Page 6372*, *15*(8), 6372. https://doi.org/10.3390/SU15086372

Thijs Vandenbussche. (2021). *Is the EU's building renovation wave 'fit for 55'?*

Van Bueren Hein Van Bohemen, E., Itard, L., & Visscher, H. (n.d.). *Sustainable Urban Environments*.

van den Boomen, M., Spaan, M. T. J., Shang, Y., & Wolfert, A. R. M. (2020). Infrastructure maintenance and replacement optimization under multiple uncertainties and managerial flexibility. *Construction Management and Economics*, *38*(1), 91–107. https://doi.org/10.1080/01446193.2019.1674450

Van Unen, J. A. (2019). *The Energy and Comfort Performance of a Lightweight Translucent Adaptable Trombe Wall in Different Buildings and Climates A numerical study Source: Double Face 2.0 Project* [Master Thesis, Technical University of Delft]. http://repository.tudelft.nl/

Vollmer, R., Lämmle, M., Hess, S., & Henning, H.-M. (2022). *Optimization of energetic refurbishment roadmaps for multi-family buildings utilizing heat pumps*. https://doi.org/10.1016/j.enbuild.2022.112729