AGENT-BASED MODELING OF THE DUTCH HOUSING MARKET

Master Thesis

Pim Kerkhoven

Supervisor
Tillmann Miltzow

Master Thesis
Computing Science
25 October 2022
Student Nr. 3925811
Utrecht University

Universiteit Utrecht

Abstract

There currently is a crisis in the Dutch housing market. We believe the main cause of this crisis is a high housing shortage. As a result rents and buying prices are high. We also observe a lock-in effect where people cannot move out of their houses even if they want to. However, if the Netherlands would have the same m^2 per person as Germany there is enough space to house everyone. Using an agent-based model we research if policies that reduce the m^2 per person counter the effects of the housing crisis. Using this model we conclude that this is not necessarily true. It seems that reducing the m^2 per person is not the most important factor in countering the housing crisis. We believe that it is more important to create extra houses. Our model indicates that this is possible without constructing extra new-built houses.

Acknowledgements

I want to thank Till for all the ice cream and for supervising my thesis. I also want to thank Camping Maurik for the hospitality during my stay there. It really helped me finish my thesis. Next, I thank my parents for still supporting me after ten years of studying. Finally, I want to say to Lena: The wait is over. Let's go to Sweden!

Contents

Chapter 1

Introduction

The housing market is a hot topic in the Netherlands right now. Every day there are new stories about the problems in the housing market. Computer scientists can contribute to this discussion by modeling the housing market. We can use the resulting models for evaluating policies that affect the housing market.

In this thesis, we develop a model to study the current housing crisis in the Dutch housing market. In our opinion, the biggest aspect of this crisis is the large housing shortage. A housing shortage occurs when more people are searching for a house than there are houses available. Currently, there is a shortage of 279,000 houses. According to [Dutch Ministry of the Interior and Kingdom Relations 2021a], this number increases to 317,000 in 2024. After which they expect the number to slowly decline.

We believe the housing shortage is the main cause of many problems in the Dutch housing market. It leads to high buying prices and rents because there is more demand than supply. The shortage also causes a lock-in effect for households. Locked-in households want to move out of their current house but are unable to do so. For example, a lot of new graduates cannot afford to move out of their student house. The rents and buying prices of the houses they want to move to are too high.

This is a big problem for students and starters. Many of them cannot afford a new house [Dutch Ministry of the Interior and Kingdom Relations 2021b]. As a result, people apply for social housing. Which in turn causes long waiting times for social housing. For at least a quarter of the Dutch municipalities, the average waiting time is over seven years [Kraniotis and Jong 2021].

An obvious solution to the housing shortage is building more new houses and creating extra living space. However, we believe there is already enough space available. We can see this if we compare the Netherlands with countries such as Germany and England. In the Netherlands, on average, households live on $65m^2$ per person [Centraal Bureau voor de Statistiek 2019]. This is $18m^2$ more than in Germany, where the average is $47m^2$ per person [Destatis 2020].

The Netherlands has approximately 17 million inhabitants. This means there is a total of around 17 million $\cdot\, 65m^2 =$ 1105 million m^2. If the Netherlands would achieve the German average of $47m^2$ per person, they could house approximately 23 million people, with the current amount of houses. This is an extra 6 million persons, which is enough to solve the housing shortage.

It raises the following question. Can we fix the problems in the housing market by using the existing space more efficiently? We want to see if using existing space more efficiently counters the effects of the housing crisis. These effects are high rent and buying prices, a strong lock-in effect, and a high housing shortage.

We hypothesize that policies that reduce the space per person counter these effects. So they lower the prices, weaken the lock-in effect and reduce the shortage. By reducing the space a household uses more space becomes empty. If we use this emptied space to house more households, we reduce the shortage. This creates a better balance between the demand and supply of houses. Therefore, we should see a positive effect on buying and rental costs. It also leads to more choices for households. So it is easier for households that want to move to find a house. Therefore weakening the lock-in effect.

However, our hypothesis could be false. Maybe, we misjudge the influence of reducing the space per person on the housing crisis. It could be that it is not an important factor in fixing the crisis. It could also be that the policies have negative side effects. A policy could, for example, reduce the space per person and counter the effects of the housing crisis. But if it does this by placing poor people in tiny apartments then maybe it is not a good policy.

To summarize, we research the following hypothesis:

Policies that reduce the m^2 per person of households counter the effects of the housing crisis.

We start by looking at the effects of the housing crisis in Chapter 2. To do this we first look into the housing market. We give an overview of how the market developed. After this, we discuss each effect of the housing crisis.

Next, we design our model. For this, we rely on agent-based modeling. Agent-based modeling is a modeling technique used for modeling complex systems. In a complex system, complexity emerges from simple interactions between the system's components. Agent-based modeling tries to capture the emerging complexity by modeling these interactions. We discuss complex systems in more detail in Chapter 3. In Chapter 4 we show how to use agent-based models to model complex systems.

An agent-based model of the housing market simulates the interactions between households and houses. Households buy or rent houses listed on the market. Several studies have modeled the housing market using agent-based modeling. We discuss the studies that are most relevant to our model in Chapter 5. Based on this literature we design our model in Chapter 6. We specifically model the Dutch housing market. Therefore, we initialize our model with data about the Dutch housing market from the Centraal Bureau voor de Statistiek (CBS). Our model differs from models in the literature in two key ways. First, we use utilities instead of probabilities to let households make decisions. Second, this is the first model that models the social housing sector in detail. To the best of our knowledge, there are no other models that do this.

We discuss the results of this model in Chapter 7. We use these results as a baseline. We compare this baseline to six experimental models. The experimental models extend the original model with an extra policy. The aim of each of these policies is to reduce the space per person. We observe if the policies counter the effects of the housing crisis. We discuss this in Chapter 8.

We first look at a policy that reduces the space per person by taxing the amount of space a household uses. We also look at a policy that limits how much space per person a household may use. Both of these policies cannot create additional houses. We see that these policies counter some of the effects of the housing crisis. They do not counter all effects. We argue that we need to create extra houses to counter all effects of the housing crisis.

We explore four policies that do create extra houses. First, we look at a policy that allows households to share a rental house. When sharing, two households live in the same house. This policy gives good results but seems to affect only poorer households. Also, the created extra houses are all part of the rental market. Therefore, the policy does not affect the buying market.

Next, we look at a house-splitting policy. Splitting allows homeowners to convert a part of their house into a rental house. This reduces the size of their own house and creates an extra house. Therefore, the splitting policy affects both buying and rental houses. We observe that the extra houses are similar to each other. This means, households looking for other houses still have limited choices. We aim to fix this by rewarding households that split their house. This leads to an increase in the number of split houses. However, this does not have the desired effect. As we will see, the variety of houses is not increased.

Lastly, we look at a policy that splits listed houses. The results of this are good. However, we see that we need to create too many houses to counter the housing crisis. This is a problem that all our house-creating policies have.

In Chapter 9, we conclude that our hypothesis is not necessarily true. We find that reducing the space per person is not the most important factor in countering the housing crisis. It looks to be more important to create houses of appropriate sizes. We believe this can be done by modifying existing houses and without constructing extra new-built houses.

Chapter 2

Overview of the Dutch housing market

In this chapter, we give a historic overview of the Dutch housing market. We discuss the current housing crisis and its effects. We focus on two important entities: households and houses. To define these entities more formally, we rely on the definitions provided by [Centraal Bureau voor de Statistiek 2022a].

For households, we specifically look at private households. A private household is a collection of one or more people sharing the same living space. They provide for their own everyday needs in a private, non-commercial way. For example, a family of four or a 25-year-old living on their own. We do not consider institutional households. In an institutional household, housing and daily needs are provided professionally. For example, people living in nursing or care homes form an institutional household. Whenever we use the term household we mean a private household unless specified otherwise. Another important concept for households is the reference person. This is a member of the household that gives the household its characteristics. An example of such a characteristic is the age of the household.

A house is a building intended for permanent residence by one household. We distinguish between two types of houses: buy houses and rental houses. A buying house is a house that is the property of the household living in it. A house is a rental house when the household living in it does not own the house. Sometimes we count a house without a household living in it as a rental house. We do this if it is probable that the owner intends to rent out the house and owns more than two houses.

There are two types of rental houses: private and social rental houses. For social rental houses, there is a maximal rental price. There is also an income limit for households that want to live in a social house. Although social houses exist where this income limit is not a requirement. Households living in social housing can get a rent allowance from the government. The government pays rent allowance to households to help with paying the rent. The amount of rent allowance a household receives depends on its income and the amount of rent [Belastingdienst 2021]. For private rental houses, there is no maximal rent price, no income limit, and no rent allowance [Rijksoverheid 2022].

Whether a rental house is a social rental house or not depends on its monthly rental price. If it is below a certain threshold the house is a social rental house. If it is above this threshold the house is a private rental house. We call this threshold the liberalization threshold. In 2022 the liberalization threshold in the Netherlands is €763,47 per month [Rijksoverheid 2022]. Social housing corporations own most of the social rental houses [Centraal Bureau voor de Statistiek 2021a]. Whereas, private investors own more of the private rental houses [Dutch Ministry of the Interior and Kingdom Relations 2021b].

We now look at certain statistics about the properties of households and houses. We track how these statistics developed over time from 1900 up to the present day. Table 2.1 shows what each statistic looked like in 1900 and 2019.

We use 1900 as a starting point because it marks the start of the modern Dutch housing market. Around 1900 the Dutch government introduced the first housing law (Eerste Woningwet). The goal of this law was twofold. The first goal was to promote building and living in good houses. The second goal was to prohibit living in bad and unhealthy houses. The housing law led to the demolition of bad buildings. This made space available to build new good houses. Social housing corporations did this in large numbers. Therefore, the result of the housing law is a world record production of social houses. In a period of 100 years, over 2.5 million social houses were built [Lans and Pflug 2019].

Table 2.1: Comparison of housing market statistics in 1900 [Lans and Pflug 2019] and 2019 [Centraal Bureau voor de Statistiek 2021b; Centraal Bureau voor de Statistiek 2022b; Centraal Bureau voor de Statistiek 2019]

Metric	1900	2019
population size	5.104 million	17.282 million
number of households	1.113 million	7.925 million
average household size	4.59	2.18
number of houses	~1 million	7.892 million
average house size	~50 m^2	119 m^2
average persons per house	~5	2.19
m^2 per person	~10 m^2	65 m^2
percentage of houses for buying	15%	57%
percentage of houses for social rental	No data	29%
percentage of houses for private rental	No data	14%

2.1 Households and population

We first look at the population size and number of households. We plot these in Figure 2.1. For the number of households, there only exists continuous measurements since 1960. Since this year the number of households has more than doubled. The population size increased at a slower rate. This indicates a decline in average household size. When looking at Figure 2.2, we see this is the case.

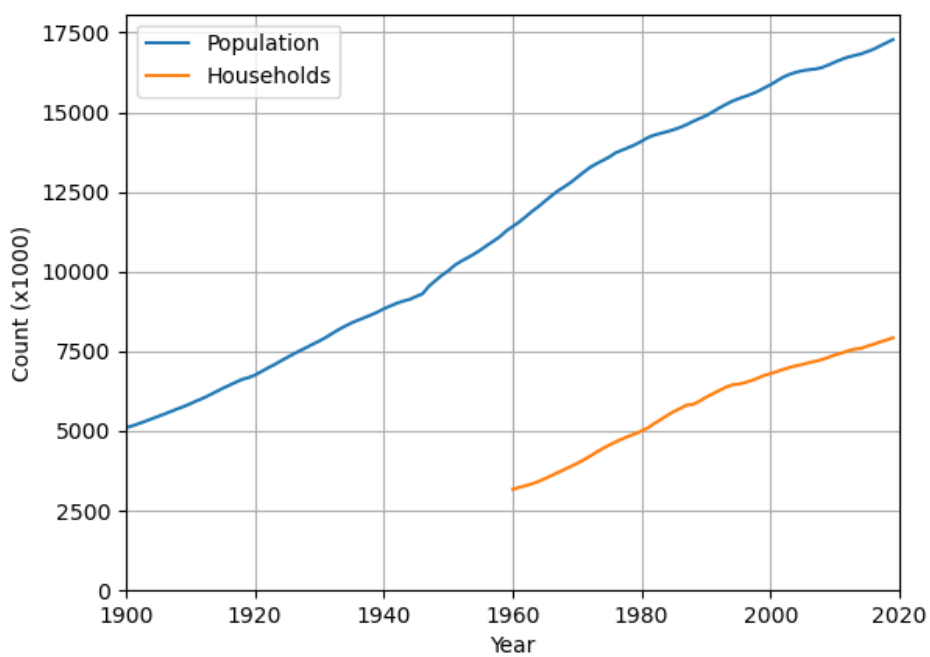

Figure 2.1: Population size and number of households since 1900 [Centraal Bureau voor de Statistiek 2021b]

The reason for this decrease is a relative increase in single households. Figure 2.3 plots the number of single households against the number of multi-person households. It also plots the total number of households. The figure shows that the number of single households increases at a faster rate than the multi-person households.

The faster increase has two main causes [Centraal Bureau voor de Statistiek 2018]. First, young people move out

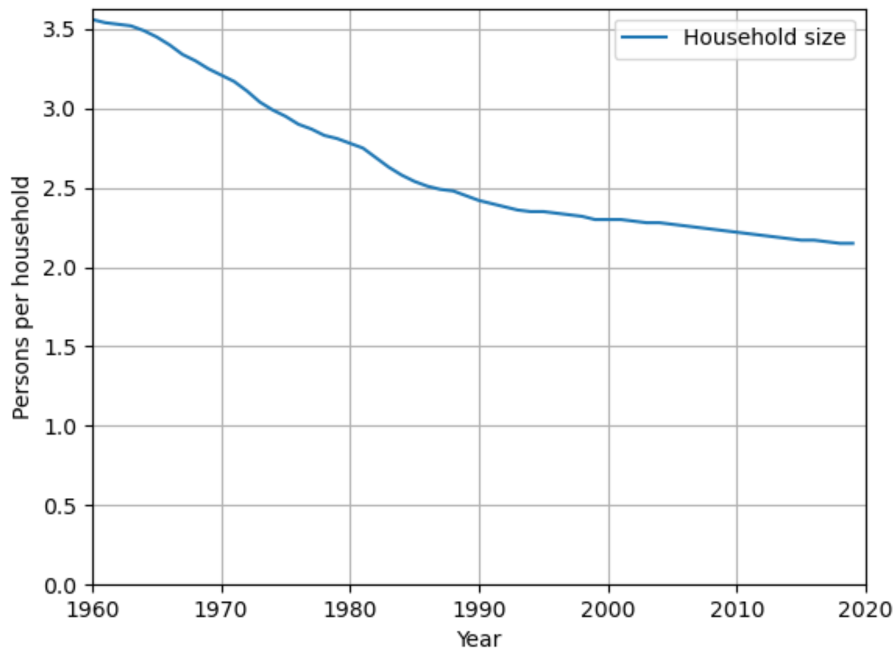

Figure 2.2: Average household size since 1960 [Centraal Bureau voor de Statistiek 2021b]

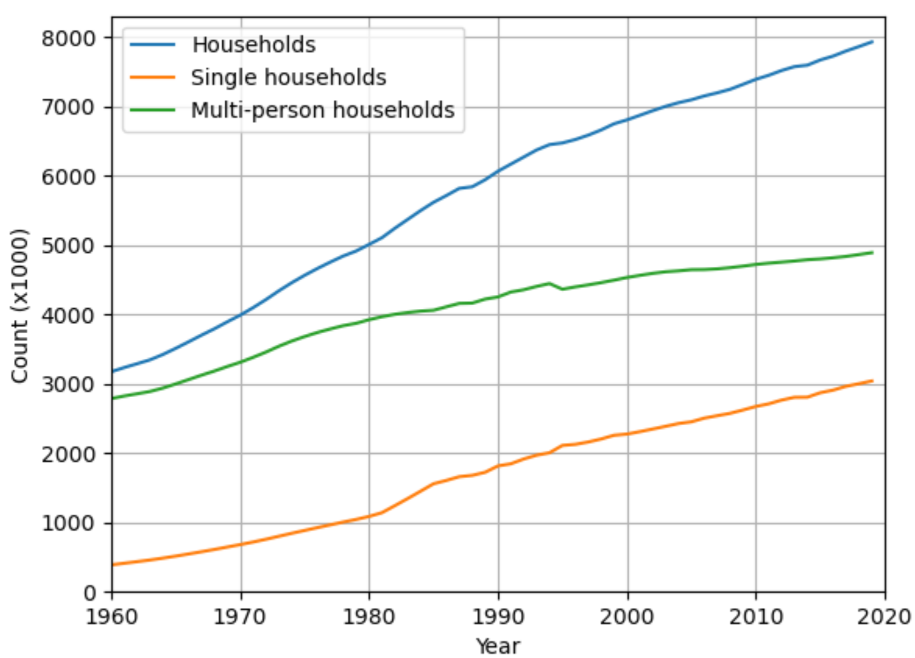

Figure 2.3: Single households versus multi person households [Centraal Bureau voor de Statistiek 2021b]

of their parents' home earlier and often alone. This results in a new household of size one. Also, the size of their parents' household decreases by one. It used to be the case that a person stayed in their parents' house until they found a partner. After a person found a partner, they moved into a house together.

We illustrate how this leads to a decreased household size with an example. Assume we have two households of size three with two parents and one child. The average household size is three. If the children move out and start to live together, we end up with three households of size two. The average household size is now two. However, if the children move out and start to live on their own, we end up with four households. Two of size two, and two of size one, giving an average of 1.5.

The other reason for the decreasing household size is the fact the elderly live longer on their own. They do not move into a nursing or care home, which would make them a part of an institutional household. The elderly also do this after their partner dies, leading to more single households [Centraal Bureau voor de Statistiek 2018].

2.2 Houses

As Table 2.1 shows there were approximately one million houses in 1900. This number went up to keep up with the number of households. We show the total amount of houses per year in Figure 2.4.

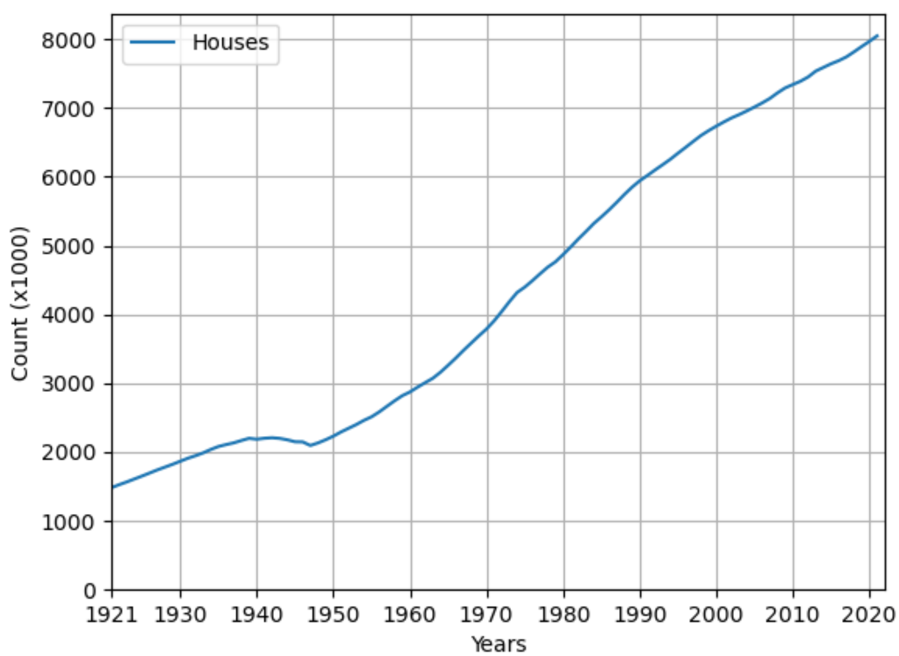

Figure 2.4: Number of houses since 1921 [Centraal Bureau voor de Statistiek 2022b]

Figure 2.5 shows the number of houses added per year. We see a decrease in the number of houses around the second world war. During the war, 25% of the houses were damaged [Sociaal en Cultureel Planbureau 1998]. Houses damaged beyond recovery were demolished, leading to a drop in the number of houses. The bookkeeping of demolished houses happened mostly in 1947. This is why we see the largest decrease in that year.

The decrease in houses led to a high housing shortage. To reduce the shortage the government needed to build new houses. However, after the war building material and workers were scarce. Also, the focus was on rebuilding other sectors first [Sociaal en Cultureel Planbureau 1998].

Therefore, the real catch-up came in the late 1960s. In those years there was a huge construction of new houses. The construction followed a quantity-over-quality approach, resulting in a lot of cheap apartments in high-rise buildings [Sociaal en Cultureel Planbureau 1998].

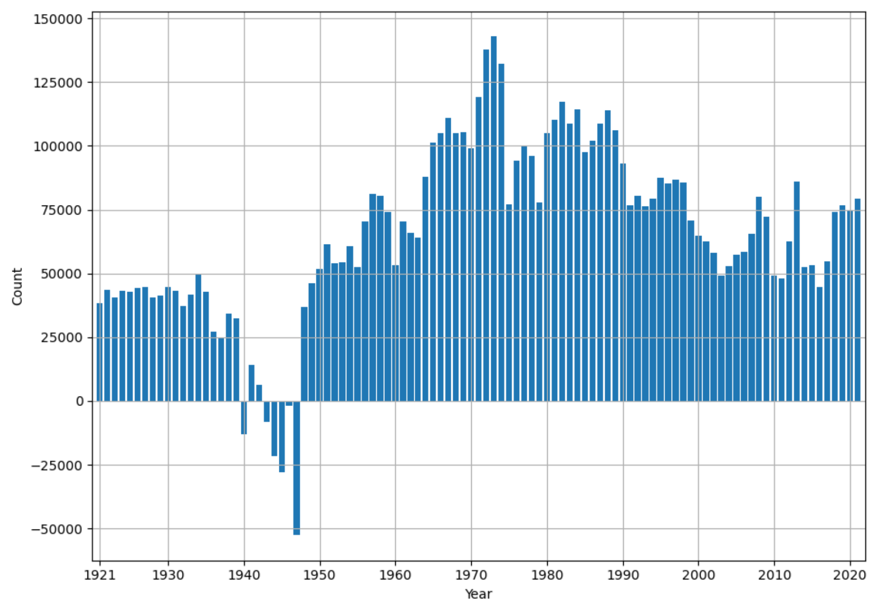

Figure 2.5: New houses per year since 1921 [Centraal Bureau voor de Statistiek 2022b]

Over time the size of houses changed. As Table 2.1 shows the average house was 50 m^2 in 1900. In 2019 this increased to 119 m^2. [Regiocontainer 2019] analyzed the size of terraced houses by build year from 1950 until 2019. They found that the average size of these houses increased from 99.4 m^2 to 120.4 m^2. At the same time, the average lot size of such houses decreased. From 187.3 m^2 for houses built in 1950 to 131.7 m^2 for houses built in 2019. So, the newer houses are bigger but have less outside space.

From the observed increase in house size and the decrease in household size, we conclude that the average space per person increases over time. Figure 2.6 shows this.

2.3 Distribution of houses over markets

In 1900, around 15% of all houses were for buying. This number increased to 57% in 2019 [Lans and Pflug 2019]. We show how this number increased over time in Figure 2.7. We see it starts to increase after the second world war.

Before the war, promoting home ownership was not seen as a government task. After the war, the government thought that owning a house could contribute to residential civilization. Owning a house should lead to more responsibility, more saved money, and more stable families [Lans and Pflug 2019]. The introduction of the National Mortgage Guarantee (NHG) in 1956 led to even more buying. This guarantee meant that the government would pay off a household's mortgage. But only if the household could not afford to do so due to unforeseen circumstances. This made banks more willing to hand out mortgages, meaning more people were able to buy.

The increase in the percentage of buying houses came at the cost of the private rental sector. This sector decreased from 60% right after the war to 11% in 2000.

2.4 Housing crisis

There currently is a housing crisis in the Netherlands. It is hard to completely understand what causes this crisis. A lot of different factors influence the housing market and people's behavior. We believe the main cause of the housing crisis is the high housing shortage. The current housing shortage is around 279,000 houses. [Dutch Ministry of the Interior and Kingdom Relations 2021a] expects this number to increase to 317,000 in 2024. After 2024, they expect the shortage to decrease to 179,000 in 2035.

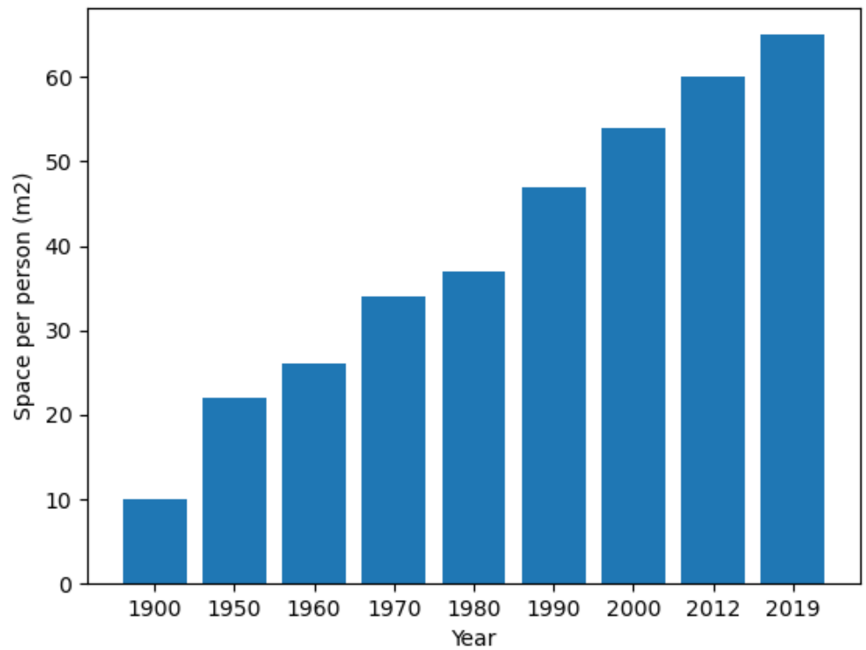

Figure 2.6: Average space per person since 1900 [Lans and Pflug 2019; Centraal Bureau voor de Statistiek 2019; Centraal Bureau voor de Statistiek 2021c; Centraal Bureau voor de Statistiek 2021b]. Note that the x-axis does not increase linearly.

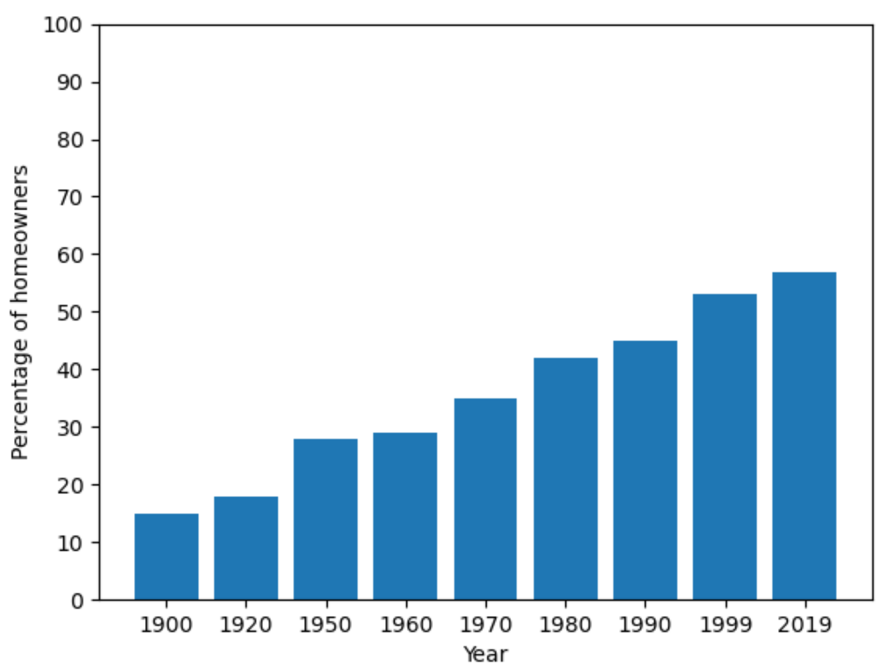

Figure 2.7: Homeownership since 1900 [Lans and Pflug 2019]. Note that the x-axis does not increase linearly.

The housing shortage should not be too low or too high [Dutch Ministry of the Interior and Kingdom Relations 2021a]. A too low shortage leads to vacant and impoverished houses, due to deferred maintenance. A too high shortage leads to long waiting times when searching for a house. As well as high rents and high buying prices. We will now look in more detail at these three effects.

We first look at the buying prices. We display the buying prices since 1995 in Figure 2.8. We see that the prices are currently the highest they have ever been since 1995 [Dutch Ministry of the Interior and Kingdom Relations 2021b]. The buying prices increased since 2013 when the prices recovered from the previous crisis in the housing market.

Figure 2.8: Buying price since 1995 [Centraal Bureau voor de Statistiek 2022c].

That crisis was different from the current crisis, as it was caused by a global financial crisis in 2008. The 2008 crisis led to prices dropping enormously. A similar effect was observed for house prices around 1980 as an effect of the oil crisis [Sociaal en Cultureel Planbureau 1998]. The current housing crisis is different in that is not caused by a financial crisis. Instead, we believe one of the causes is a lack of construction, leading to a high housing shortage.

The buying price also rises because of an increase in the number of houses bought by investors. Investors buy a house not to live in it but to rent it out. The number of investors increased because of low interest rates. These low rates made alternatives to buying a house, such as savings accounts or bonds, unattractive [Langenberg and Jonkers 2022]. Investors compete for the same houses as households that want to buy a house to live in. Given they got more active, there is more competition in the buying market. This causes the prices to increase [Langenberg and Jonkers 2022].

We now look at the rental prices. Figure 2.9 shows how rental prices have developed over the years. We notice that the rental prices never decrease. So if we only look at the price we always see an increase. However, if we correct this value for the inflation in some years the rent actually decreases [Dutch Ministry of the Interior and Kingdom Relations 2021b]. We see this in Figure 2.10, where we plot both the inflation and the rent increase.

In the private sector, the rents increase faster than in the social rental sector. In recent years, this is the sector where most houses become vacant. So most rental houses that a household can rent are part of the private rental market [Dutch Ministry of the Interior and Kingdom Relations 2021b].

Last we look at the lock-in effect. The lock-in effect consists of two components. The number of households that want to move and the time it takes them to find a new house. Throughout this thesis, we call those two concepts *want to move count* and *want to move time*.

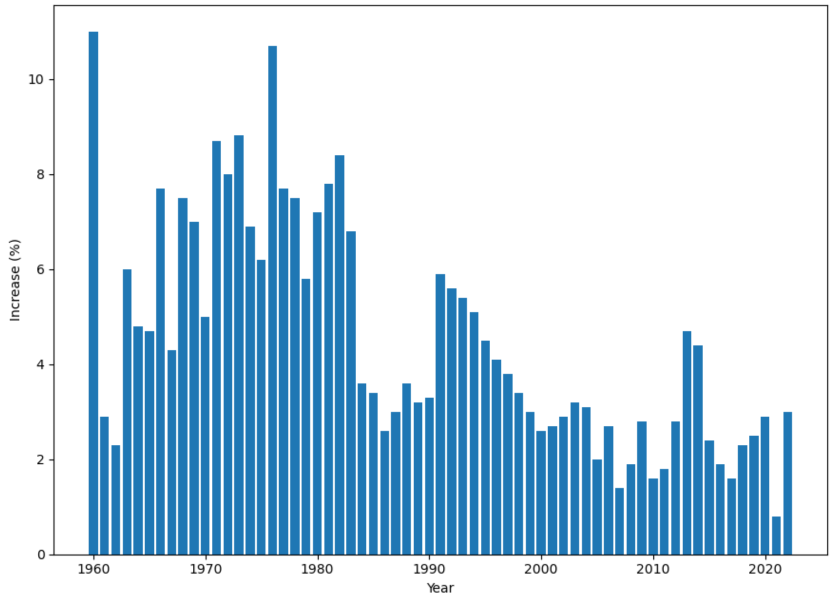

Figure 2.9: Rent increase since 1960 [Centraal Bureau voor de Statistiek 2022d].

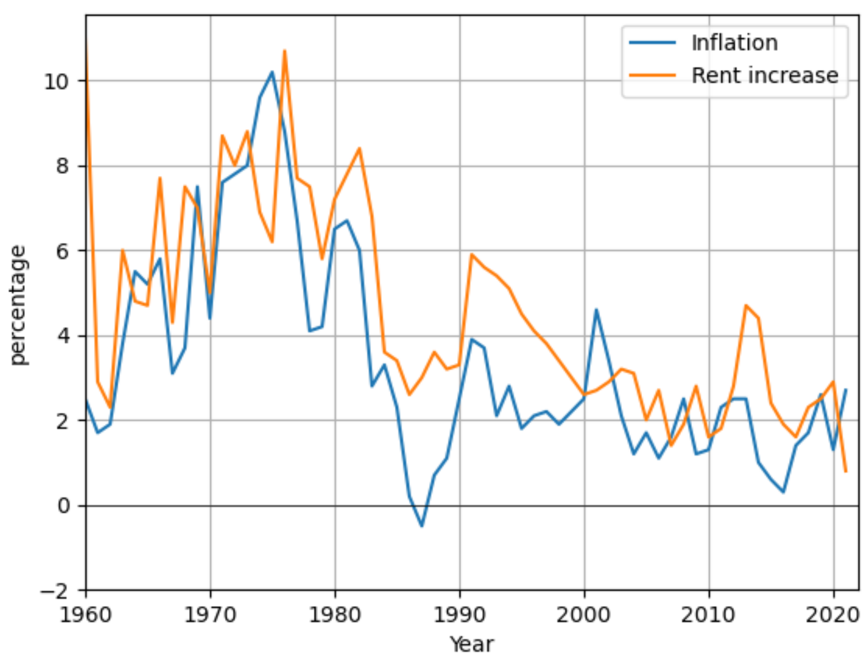

Figure 2.10: Rent increase and inflation since 1960 [Centraal Bureau voor de Statistiek 2022d; Centraal Bureau voor de Statistiek 2022e].

The percentage of households that want to move increased from 24% in 2012 to 40% in 2021 [Dutch Ministry of the Interior and Kingdom Relations 2022]. In 2021 this means around 3.2 million households indicated they want to move in the next two years. Of these 3.2 million households 1.7 million households are actively searching for a new house. This is an increase of 16% compared to 2018 [Dutch Ministry of the Interior and Kingdom Relations 2022]. So we see that since the end of the last housing crisis in 2012, the *want to move count* has increased.

We could not find an extensive dataset for the *want to move time*. We only have some data for the rental sector. On average households had to search for 10 months for a private rental house in 2021 [Dutch Ministry of the Interior and Kingdom Relations 2021b]. For a social rental house, this was 27 months. This is an increase of three months compared to 2018 [Dutch Ministry of the Interior and Kingdom Relations 2019; Dutch Ministry of the Interior and Kingdom Relations 2022]. We do not have any data to compare these numbers to. Not from other countries or earlier years. So it is hard to say if these are extreme values. Although having to search for more than a year for a new house is long.

Chapter 3

Complex systems

A complex system is a system made up of a large number of components that interact in a non-simple way [Simon 1962]. These interactions lead to behaviors and properties that characterize the system. For example, the shape of a flock of birds is caused by the movement of individual birds. Or how individual transactions between buyers and sellers lead to a market price. For a complex system, it is hard to explain the system's behavior and properties from the behavior and properties of its individual components [Sayama 2015]. In other words, the whole is greater than the sum of the parts [Newman 2011; De Domenico et al. 2019].

Complex systems show up everywhere in the real world. From the brain to economies and the global climate to stock markets [Sayama 2015]. One example of a complex system is a dog wagging its tail. From this, we observe the dog is alive. But, it is much harder to explain what processes in its body are causing the dog to be alive and wag its tail [Sayama 2015].

Another example is the already mentioned flocking of birds. Flocks of birds exhibit shapes like a V-formation or a swarm. These shapes result from the movement of individual birds. The birds try to keep moving in the same direction as their neighbors. And they also try to avoid collisions with these neighbors. We can describe the behaviors and properties at both the individual and system level. However, it is hard to explain how one leads to another. The relation between the two levels is non-trivial [Wilensky and Rand 2015].

A system is not complex when its components are independent of each other. For example, an ideal gas or a series of dice rolls. In such systems, the individual components do not interact. Therefore, there is no system behavior forming from individual interactions [Sayama 2015].

A system is also not complex when its components depend too much on each other. Such systems have tightly coupled components. This results in too few degrees of freedom in the system. We can describe such systems with a low number of variables. Examples of this are rigid bodies in physics or fixed dice rolls [Sayama 2015].

Complex systems lay between these two extremes. Their components are neither completely independent nor completely dependent.

3.1 Elements of complex systems

There are multiple definitions of what a complex system is. These definitions are not equivalent or consistent [Petty 2018]. However, they can still be informative. We will look at one such definition and then describe the most important elements in more detail.

Definition (Complex System [Sayama 2015])

> A complex system is a *network* made of a number of *components* that *interact* with each other, typically in a *non-linear* way. A complex system may arise and evolve through *self-organization*. Making it neither completely regular nor completely random. This permits the development of *emergent behavior* at *macroscopic scales*.

3.1.1 Components

A complex system consists of multiple separate parts or components. We also call these components agents or actors. An agent is a separate entity in the system with a clear boundary. This boundary shows what is part of the agent and what is not. In the case of the housing market, we often look at households as individual agents. The persons that make up a household are within the boundary of the agent. Everything that is not a part of a household, such as other households or banks, is outside the agent.

An agent can be a complex system itself. As stated above, a household consists of multiple persons. The social interactions between the members of a single household also form a complex system. And each person is, of course, also a complex system itself. For complex systems, we often observe this nesting of complex systems.

The agents in a complex system are not driven by a central controller. Each agent acts autonomously and in its own interest [Wilensky and Rand 2015]. There is not a leader bird telling the other birds where to go and in what formation to fly.

Each agent has its own state. This state describes the current values of the agent's properties. For a bird, this could be its speed, position, and direction. Over time, the state changes because of actions taken by the agent and interactions with other agents.

3.1.2 Networks and non-linear interactions

If we want to quantify a complex system, we need to specify the network(s) and the dynamics of the system [Newman 2011]. The network captures which agents interact with each other. The dynamics describe the agent behavior, so how these interactions work.

The network can be a grid connecting neighboring agents and agents on the same grid tile. It can also be a graph connecting agents through vertices. A system can have more than one network. For example, a system of social interactions can have a grid connecting neighbors. As well as a graph connecting friends and family. Agents interact with other agents through connections on both networks [Macal and North 2010].

We specify how they interact and behave in the system dynamics. The dynamics describe which action an agent can take and how these affect other agents. In the case of the housing market, this could be buying a house. This action involves two household agents and a house. One agent pays money and gains a house. The other receives money and loses that house. Examples of other actions are listing a house for sale, or placing a bid on a listed house.

In complex systems, these interactions are typically non-linear [Sayama 2015]. This can lead to a small cause having a large effect [Sayama 2015; Petty 2018]. There is no proportionality between the size of the cause and the size of the effect. A minor change in effect can have a large change in cause and vice versa [Willy et al. 2003]. We also observe randomness and chaos in the interactions in complex systems [Wilensky and Rand 2015].

For the housing market, we see an example of non-linearity when we look at the bidding process. Say a household is bidding for a house. They can place a bid of a certain value on that house. There are two outcomes for this action. If they have the highest bid, they buy the house. Otherwise, they do not buy the house. Now assume they are bidding one euro below the highest bid. Currently, they are not getting the house. However, if they would raise their bid by two euros, they get the house. So a slight change in the action, bidding two euros more, results in a completely different outcome.

3.1.3 Scales, emergence, and self-organization

We observe complex systems at two scales, the microscopic and macroscopic scale [Wilensky and Rand 2015]. At the macroscopic scale, we observe the behavior and properties of the system. This is the movement or shape of the entire flock of birds. At the microscopic scale, we observe the behavior and state of an individual agent. For the flock, this is a single bird. We observe its speed, (relative) position, and direction. As stated above, a system is complex if we cannot simply explain the macroscopic behavior from the behavior of individual agents. We call such behavior emergent. We have already seen examples of this above. We define emergence as follows.

Definition (Emergence [Sayama 2015])

Emergence is a non-trivial relationship between the behaviors of a system at the microscopic and macroscopic scale. We call macroscopic behaviors emergent when it is hard to explain them from microscopic behaviors.

Over time, the interactions between agents can produce a global pattern or structure. Because there is no global controller, we call this self-organization. There is no leader creating the pattern. The agents do this themselves [De Domenico et al. 2019]. Closely related to self-organization is emergence. However, it considers both time and scale. Whereas emergence is only about the non-trivial relationships between different scales [Sayama 2015]. We define self-organization as follows.

Definition (Self-organization [Sayama 2015])

Self-organization is a dynamical process by which a system spontaneously forms non-trivial macroscopic structures over time.

We see an example of self-organization in the flocking of birds. If we observe such a system from the start, we see that in the beginning, the birds are in random positions. We cannot observe a global pattern or shape that characterizes the flock. Over time, as birds start to move and interact with other birds we start to see the distinct shape of the flock. This is self-organization, we see a global shape created over time from the interactions of the individual birds.

3.2 Characteristics of complex systems

As stated above, there is not a single definition of what a complex system is in the literature. But there is a wider consensus on what characteristics complex systems have [Petty 2018]. We list these characteristics in Table 3.1. Non-complex systems can, of course, also exhibit some of these characteristics. However, complex systems often exhibit all characteristics. This separates them from non-complex systems.

Table 3.1: Characteristics of complex systems

Characteristic	Explanation
Non-linearity	Relations between agents are often non-linear, meaning that a small cause has a large effect [Sayama 2015; Petty 2018]
Emergence	The relationship between microscopic and macroscopic properties is non-trivial [Sayama 2015]
Nesting	The agents of a complex system are themselves a complex system [Petty 2018]
Self-organization	Over time, a complex system produces global patterns and structures [Sayama 2015]
Sensitive to initial conditions	A minor change in the starting state of the system has a large effect on how the system evolves through time [Petty 2018].
No central controller	A complex system often lacks a central controller that coordinates the process [Wilensky and Rand 2015]

3.3 Is the housing market a complex system?

We aim to build a model of the housing market to study the effect of certain measures on the housing shortage. We want to do this using agent-based modeling. This modeling approach is a good fit for modeling complex systems [Sayama 2015; Wilensky and Rand 2015]. For this technique to be effective, we need to know if the housing market is a complex system. We do this by showing that the housing market has the characteristics of a complex system. For each characteristic, we give an example of how it shows up in the housing market. We also argue that the housing market is adaptive. From this, we conclude that the housing market is a complex adaptive system.

3.3.1 Non-linearity

We showed an example of non-linearity for the housing market in Section 3.1.2. There we discussed that a small change in bidding value can result in a completely different outcome.

3.3.2 Emergence

[Kouwenberg 2015] states that boom-bust cycles characterize the housing market. These are cycles where the price first goes up drastically and then goes down hard again. It is hard to explain why these cycles occur by looking at interactions between agents. This indicates a non-trivial relationship between the microscopic and macroscopic scales. There is emergent behavior.

[Diappi 2013] names gentrification as another example of emergent behavior in the housing market. Gentrification is the process of changing a neighborhood by wealthier residents moving in. This attracts more expensive shops and businesses to the neighborhood. The cost of living increases. This forces poorer residents out of the neighborhood. This process is straightforward to observe by tracking the average wealth of residents. However, what exactly causes gentrification has divided experts for years [Diappi 2013]. This indicates emergent behavior as it points to a non-trivial relationship between the microscopic and macroscopic scales.

3.3.3 Nesting

As we showed in Section 3.1.1 nesting of complex systems happens in household agents. Another example would be a construction company. Following the same argument as for households, such companies are composed of people. The relations and interactions between these people also form a complex system.

3.3.4 Self-organization

In the housing market buying, households try to buy houses from selling households. In a simplified market, there is a constant amount of houses and households. For this scenario, over time the market price moves toward a stable value, the equilibrium. We see this as self-organization. Of course, a real housing market is not this simple. There are constant changes in the amount and properties of houses and households. This makes the market price constantly adapt to the new situation. We discuss this adaptiveness of the housing market in more detail below.

3.3.5 Sensitive to initial conditions

Markets in general, but housing markets specifically, are sensitive to initial conditions. Let's assume a static market with a fixed amount of demand and supply. In such a market prices go up when there is more demand than supply, otherwise, they go down. This is a thin line, a minor change in demand or supply can tip this balance, causing prices to rise instead of drop. Of course, in real markets, this is more complicated. Real markets are not static and have more variables involved, such as income and location. This makes it hard to predict what will tip the balance.

3.3.6 Lack of central controller

The housing market lacks a central controller. There is not a single entity telling each household where to live and at what cost. Households can decide this for themselves. There are, of course, governing bodies, but these do not control the housing market. Rather, they try to steer the market in a direction they deem correct.

3.4 Modeling complex systems

If we want to understand complex systems or predict how they will behave in the future, we need to model them. There are multiple techniques for this. One option is micro-simulation. The key idea behind micro-simulation is to update a database of samples as time progresses [Gilbert 2008]. This way, we try to simulate what a sample will look like in the future.

For the housing market, we could, for example, have a database table of households with income, age, and housing costs. We can simulate time steps by updating this table according to some update rules. For example, we can update a household's age every twelve months. Or we can update a household's income based on its current income and age. We can base these update rules on real-world data. This should lead to somewhat realistic updates.

An advantage of this approach is that we can start with real-world data. A disadvantage is that we do not simulate interactions between samples. As we stated above, these interactions are what make a system complex. Therefore, micro-simulation models fail to capture an important feature of the systems we model.

Another option is system dynamic modeling. System dynamic models describe the temporal cause-and-effect relationships between variables [Gilbert 2008]. As opposed to micro-simulation, they do model interactions between groups of agents. However, they have no direct representations of individual agents. They only deal with the macroscopic level. This makes it hard to model the differences between agents.

Differential equation models are a type of system dynamics models. An example of a model using differential equations is the Lotka-Volterra model [Wilensky and Rand 2015]. It models the size of populations of predators and preys with differential equations. There is one equation for the number of predators and one for the number of prey. The number of predators depends on the number of prey and how many predators die per time step. The number of prey depends on the birth of new animals and how many prey animals are eaten by predators. We see that these equations only model how many predator or prey animals there are. There is no representation of individual animals. If we want to model a specific type of prey that runs faster, we need to add an equation specifically for this type. We need to do this for each different type of animal we want to model. For more than ten different types this becomes infeasible.

If we do want to model the differences between agents, we can use agent-based modeling. This technique models macroscopic effects by simulating microscopic interactions between agents. This is the technique we will use for our model of the housing market. We will discuss it further in the next chapter.

Chapter 4

Agent-based modeling

Agent-based modeling tries to model emergence in a system by simulating the interactions of its agents [Gilbert 2008]. To do this, we only need to describe the behavior of the agents. We do not have to understand how this emergent behavior works [Bonabeau 2002]. This makes modeling the system easier because the individual behavioral rules are often simpler to understand than the complex interactions that occur from them [Gilbert 2008].

This is best illustrated with an example. Suppose we want to model the shape and movement of a flock of birds. We could either try to come up with equations that govern how the shape of the entire flock changes. This requires understanding complex mathematics that can describe such shapes [Wilensky and Rand 2015]. Or we could try the agent-based modeling approach. Now we need to describe the behavior and properties of a single bird agent. We need to come up with rules that govern its movement with respect to other birds. These rules are simple to formulate:

- At all times a bird tries to avoid collisions with surrounding birds.
- A bird changes its direction in the direction of surrounding birds.
- A bird changes its speed to match the speed of surrounding birds.

We observe a match between this description of agent-based modeling and how we described complex systems in Chapter 3. Both are about the relations between the microscopic and macroscopic scales. The field of complex systems is about understanding emergent macroscopic behavior from microscopic interactions. Whereas agent-based modeling is about capturing emergent behavior by modeling microscopic interactions. This resemblance makes agent-based modeling a natural technique for modeling complex systems [Wilensky and Rand 2015].

In this chapter, we will look at the benefits and drawbacks of using agent-based models. What components are part of an agent-based model? And how do we design agent-based models and what factors do we need to account for?

4.1 Benefits of agent-based modeling

4.1.1 Heterogeneity

Agent-based models are a great choice for modeling systems with heterogeneous agents [Gilbert 2008]. Agents are heterogeneous when not all agents are the same. For example, in the case of the housing market, they could differ in location, income, or age. [Wilensky and Rand 2015] claims that modeling this heterogeneity has a big impact on the result.

Because agent-based models describe each agent individually, it is easy to have heterogeneity amongst them and no agents need to be the same. With other modeling techniques, this is much harder. In system dynamics (see Section 3.4), it requires a separate differential equation for each different agent. Given all these equations depend on each other, it becomes hard to solve this for more than six different agents [Gilbert 2008].

4.1.2 Simple rules

Agent-based models are good at modeling complex interactions from simple rules. This makes them a good choice for modeling complex systems. We see this, for example, in the flocking of birds. The movement rules for a single bird are quite simple. But the macroscopic pattern of the flock movement is more complex to describe. However, by modeling each bird individually, we can capture this movement pattern in an agent-based model [Wilensky and Rand 2015].

4.1.3 Virtual field experiments

It is hard to run experiments on real-world complex systems. Especially repeating experiments and exploring different scenarios is difficult to do. Recreating complex systems with agent-based modeling enables researchers to do this [Gilbert 2008]. In this way, agent-based models serve as virtual field experiments for researchers [Sayama 2015]. This does, of course, require that the model is a good enough model of the real world. Otherwise, it would be impossible to draw correct conclusions from the experiments.

4.1.4 Emergence

Agent-based modeling is better at capturing emergence than other types of models. This is because it tries to recreate the emergence from the bottom up. This contrasts with other types of models, like system dynamics models, which try to describe the emergence directly. System dynamics models only focus on modeling the macroscopic scale. This is hard to do because the core idea of emergence is that it is hard to identify what is causing it [Bonabeau 2002].

4.1.5 Flexible

Agent-based models are very flexible. It is easy to extend them with an extra feature or behavior. Also, increasing the number of agents used in the simulation is straightforward. This allows for making incremental models. We can start with a simple model with lots of abstractions. Over time, we can make the model more complex [Bonabeau 2002]. For the housing market, we could first model households as indivisible agents. An indivisible agent is an agent that is not made up of other agents, i.e. there is no nesting of agents. Later on, we can break this abstraction and also model individual persons. We can model their individual age, income, and relationships.

4.2 Drawbacks of agent-based modeling

4.2.1 Computationally heavy

Agent-based models require a lot of computational power to run, especially if we model a lot of agents. A medium-sized number of agents, tens to millions, is the most effective [Wilensky and Rand 2015]. For more agents, it takes too much computation to simulate all interactions within a reasonable time. Of course, this also depends on the complexity of the interactions.

We should build models that take at most minutes to run. This enables easy implementation, calibration, and experimentation. For longer run times, these processes become too slow [Bonabeau 2002; Wilensky and Rand 2015].

4.2.2 Parameters

Compared to other types of models, agent-based models have lots of parameters that need tuning [Wilensky and Rand 2015]. To get to the best-performing configuration, we need to do multiple runs. This ties into the computational problem described in the previous section. We should note that [Wilensky and Rand 2015] argues that other types of models hide these parameters in assumptions. They claim that, in this way, agent-based models make these assumptions more explicit.

4.2.3 Human behavior

We often use agent-based models to simulate human behavior. This is hard to do. One reason for this is that human behavior is not always rational. This irrationality leads to not always making the best possible decisions. Next to this, the cognitive capacities of humans have limitations. This places limits on how much information a human is possible to process at a given time [Gilbert 2008]. We need to take these constraints into account when modeling human behavior. Regardless, our model of choice will have inaccuracies. These inaccuracies impact the performance and output of the model [Bonabeau 2002].

4.3 Components of agent-based models

We will now describe the components that form an agent-based model. [Macal and North 2010] states that agent-based models often consist of three major components. Agents, an environment, and relationships between agents in the form of a topology. [Wilensky and Rand 2015] states that we can model most real-world phenomena using these components. They call this the core idea of agent-based modeling.

4.3.1 Agents

Agents in the simulation correspond to actors in the real world. As discussed in Section 3.1.1, agents are separate entities. They can represent people, entire organizations, or governments [Gilbert 2008]. Agents have a state and actions they can take. Their state holds the values of different properties, such as age or income. Their actions can affect themselves, other agents, and the environment [Wilensky and Rand 2015].

The most important characteristic of an agent is that it is autonomous. It wants to achieve its own internal goals [Macal and North 2010]. This autonomy implies that an agent chooses its own actions. There is no central controller that decides for an agent. However, an agent's interactions with the outside world influence its behavior. Agents are autonomous, but they do not operate in isolation.

Besides the autonomy characteristic, [Macal and North 2010] claims there is no precise definition of what an agent is. They provide two other essential characteristics they believe an agent should have.

- Self-containment. An agent should have a clearly defined boundary. This boundary divides the world into what is part of the agent and what is not.
- Varying state. Over time, an agent's state should change. This happens because of interactions with other agents and the environment. For example, gaining money by selling a house changes an agent's wealth property.

Proto-agents

Proto-agents are a special type of agent. They serve as placeholders for future agents [Wilensky and Rand 2015]. We use them to create some type of behavior in the model without having to add an entirely new agent. Later on, we can transform the proto-agent into an actual agent, if we want more detail for this behavior. Proto-agents are not a part of the agents and environment. Other agents do not interact with them. The other way around, proto-agents affect agents and the environment.

This is best illustrated with an example. In the bird flocking example, we could introduce a bird of prey hunting other birds. We could make this bird of prey a full agent. It would also fly through the sky, have a position, and interact with nearby birds by eating them.

However, we could also first implement the bird of prey as a proto-agent. We do this to easily and quickly observe the effect of such a bird on the simulation. We implement the bird of prey as a mechanism that randomly removes birds from the simulation. Later on, we can turn this mechanism into a full agent to make the model more complex.

4.3.2 Topology and environment

The environment is the virtual world in which the agents act [Gilbert 2008]. It specifies the topology of the system, dictating which agents interact with each other. The topology can be spatial or network-based as discussed in Section 3.1.2. We can also base it on real-world data, such as GIS data [Wilensky and Rand 2015].

Another part of the environment is stationary agents. Stationary agents are agents that do not move. They have a fixed location, such as houses.

Another example is patches of grass when we model sheep eating grass on a field. We model a field as multiple patches of grass connected with a spatial topology (i.e. a grid). Each patch is an agent, growing and losing grass as sheep are eating. Sheep move over the grid and eat from the patch they stand on. The patches stay in place. Because of this stationary property, we view them as part of the environment [Wilensky and Rand 2015]. Note that this distinction is not always clear. Also, it is not the most important decision, given that we do not treat stationary agents differently from non-stationary agents.

4.4 Designing

We will now look at some general techniques for designing and developing agent-based models. [North 2018] and [Macal and North 2010] recommend using agile development methods for developing agent-based models. With these agile methods, such as SCRUM, we can make agent-based models incrementally more complex. As discussed in Section 4.1, it is easy to add complexity to agent-based models.

[Wilensky and Rand 2015] describes two approaches for designing an agent-based model: a top-down and a bottom-up approach. They state that the principle of the top-down approach is to start simple and build towards a question the model should answer. For bottom-up, the principle is also to start simple and be alert to interesting questions. From these principles, we can see the difference between the two approaches. The top-down approach tries to answer a specific question we have. The bottom-up approach tries to model some phenomenon and see what questions come to mind.

We already know our research questions, therefore we will use a top-down approach. [Wilensky and Rand 2015] and [Sayama 2015] specify some specific steps to take with this approach. We start by using the research questions to observe the scope of our agent-based model. This makes clear what it should be able to do and also what it should not do. The research questions will also drive how we make decisions in the next steps.

We use the research questions to pick our agents. Choosing what agents we need and what their properties are. We do this for both stationary and non-stationary agents. After this, we specify the topology of the system.

Now, we specify the behavior of all agents. Which actions can they take and how do interactions with other agents work. Next, we design a single time step. We describe in what order the agents take actions and which actions come first. We need to find real-world data to use for initializing the model.

Also, we pick the measures we will observe to answer our research questions. Lastly, we need a way to verify and validate the correctness of the model. Good models are validated, verified, and also replicable [Wilensky and Rand 2015]. In the next sections, we will discuss these three properties in more detail.

4.4.1 Verification

Verification checks if the implemented model corresponds to the designed model. It verifies if we correctly implemented the designed model [Wilensky and Rand 2015]. Agile development methods are a good fit for this. We start with verifying simple models. This gives us some guarantees that the basic components of more complex models work as intended. An important tool in doing this is unit testing [North 2018].

4.4.2 Validation

Validation makes sure that the model we implemented corresponds to the system we wanted to model [Wilensky and Rand 2015]. It happens on two axes, the micro-macro axis, and the face-empirical axis.

At the micro-level, we check if the agent's behavior corresponds to the behavior of real-world actors. We check if we captured the important parts of individual behavior. At the macro-side of this axis, we check the emergent behavior. We validate that the emergence in our model resembles the emergence in the real-world system. This way we confirm that we indeed captured the important parts of the system. On the other axis, the empirical side checks if the data the model generates corresponds to the data of the real-world system. We can check this by running statistical tests to check how well the generated data fits the real data. We use face validation to check if the model properties are similar to the system properties [Wilensky and Rand 2015].

4.4.3 Replication

Replication is about documenting our design. Others can use this documentation to implement our model and verify our results [Wilensky and Rand 2015]. Agent-based models are hard to replicate. They require the correct implementation of a lot of detailed processes. The connections between these processes are non-trivial. If we want another researcher to be able to replicate our models, we need a good way to describe the models.

To this end [Grimm et al. 2006] proposed the Overview, Design Concepts, and Details (ODD) protocol. The idea of the ODD protocol is to always structure the information on agent-based models in the same way. The protocol has seven elements, grouped into three blocks.

The first block gives an overview of the model. It gives a quick idea of the model's focus. After reading this block, someone should be able to implement a skeleton of the model. The elements of this block are purpose, state variables and scales, and process overview and scheduling. The purpose informs about what the goal of the model is. Why did we build the model? Or what problem does it try to solve? The next step is describing the state variables and scales. The state variables describe the agents of the model and their properties. The scales inform about the length of time steps or the size of spatial dimensions. The last element of the overview block describes the process overview and scheduling of these processes. The process overview gives a short description of all processes in the model. Examples of processes are buying or selling a house. Scheduling describes the order in which these processes take place.

The second block discusses design concepts. This block has one element, also called design concepts. It discusses the general concepts underlying the model design. It links the concepts we use to the field of Complex (Adaptive) Systems. One example of this is describing what properties we want to emerge from our model. Another example is describing how agents observe their environment.

The last block describes the details of the models. After reading this block, it should be possible to completely reimplement the model. Its elements are initialization, input, and submodels. Initialization describes how we set up the model at the start of a simulation run. Input describes the data needed to run the model. The submodels give a detailed explanation of all the processes.

Chapter 5

Agent-based models of the housing market

We now turn to building an agent-based model of a housing market. In the next chapter, we will propose our own model. But first, we look at existing models. We do this to get an overview of what is possible and what choices we need to make for our own model. We start with an overview of existing models. After this, we discuss the most relevant components of housing market agent-based models in more detail.

5.1 Overview

Only a few studies use agent-based models to study the housing market. Of these studies, the most influential are [Geanakoplos et al. 2012], its follow up [Axtell et al. 2014], and [Baptista et al. 2016].

[Geanakoplos et al. 2012] and [Axtell et al. 2014] study the Washington D.C. housing market during the 2007 housing bubble and its aftermath. These studies underpin most of the subsequent studies. We see those as foundational studies. They are not the first to build an agent-based model of the housing market. They are the first to use micro-level empirical calibration. This is a technique used by most of the later studies, as it gives good results. The technique calibrates the behavior of agents based on real-world data. They use this to calibrate parameters in bidding price equations based on transaction data. Bidding price equations determine what price an agent bids for a house.

[Geanakoplos et al. 2012] and [Axtell et al. 2014] have a detailed representation of the buying and selling market. They do not model the rental market in detail. [Baptista et al. 2016] does include a more advanced rental market. This study builds a model of the British housing market.

[Evans et al. 2021] models the Greater Sydney region in Australia. The authors try to predict the future housing market price. This differs from previous studies. Previous studies focussed more on explaining the historic behavior in housing markets. These studies also explore ways to influence this behavior. This way, such studies evaluate the effect of certain policy decisions.

[Evans et al. 2021] tries to improve on the mentioned studies by capturing the spatial dimension of the housing market. This creates submarkets in the global market based on the location of a house. They also use these submarkets to limit the number of houses agents can view when searching for a new home. This limits the cognition of an agent.

[Yun and Moon 2020] models the Korean housing market. The authors also try to limit agent cognition. They extend the models of [Axtell et al. 2014] and [Baptista et al. 2016] with a brochure mechanism. This brochure shows a limited selection of houses for the agent to bid on.

[Zhuge and Shao 2018] couples a housing market model with an advanced demographic model. They split each time step into two phases. First is the demographic phase and then the housing market phase. The demographic phase models the entire population. It captures factors such as marriage, divorce, emigration, immigration, work, birth, and death.

None of the discussed studies models the different renting sectors. They only model the private renting sector. They do not model sectors like social or student housing. [Baptista et al. 2016] have a mechanism they call social housing. However, they use this as a fallback for homeless households. They only place a household in social housing if the household cannot find a house. This assumes an unlimited supply of social houses.

5.2 Spatial models

[Evans et al. 2021] states that most existing models of the housing market do not capture the spatial dimension of the housing market. However, the authors have the opinion that the housing market is a spatial market. This spatial structure leads to different submarkets, each with its own market price. For example, when the city center is more expensive than the suburbs.

Many studies do not capture this spatial component because it increases the complexity of the model. Although it would be desirable [Baptista et al. 2016]. There are some studies, like [Ge 2014] and [Ustvedt 2016], that use a grid-based structure. These grid-based approaches come with their own limitations.

First, because of the added complexity, grid-based models often use a small-scale grid. This small grid is only useful for modeling cities. Large computation times make it infeasible for simulations of entire countries [Baptista et al. 2016].

Next, the existing grid-based studies assume monocentric cities (i.e. cities with a single center). In monocentric cities, there is a relation between the value of a house and its distance to the city center. The closer to the center a house is, the more expensive it is. The monocentric assumption makes it impossible to model environments with multiple centers (polycentric) [Evans et al. 2021].

Lastly, grid-based approaches run into problems when including the construction of new houses in the model. This is because of the finite size of the grid. This leads to no space being available for new houses when the grid is already full [Yun and Moon 2020].

[Evans et al. 2021] uses a graph-based approach. Modeling the Greater Sydney area, each node is a district in this area. The edges show which districts lay next to each other. Each district is now modeled similarly to other non-spatial models such as [Baptista et al. 2016] and [Axtell et al. 2014]. Of course, with a flow of information between districts through the network. This flow also limits what houses a household views when looking for a new house. Households have a higher chance of seeing nearby houses than those that are far away. [Evans et al. 2021] calls this the outreach factor. The outreach factor causes imperfect information for households. As stated in Section 4.2.3 imperfect information is important when modeling human behavior.

5.3 Modeling human behavior

A common behavioral theory in modeling financial decisions is the utility maximization theory [Zhuge and Shao 2018]. The utility measures the value of a decision. For the housing market, we can base this value on monthly payments, house quality, and more.

Utility maximization theory assumes that, when faced with a choice, agents always choose the alternative with the highest utility [Zhuge and Shao 2018]. To do this, agents need the ability to measure their current utility and the utility of alternatives. Existing studies do this by moving households into the places that have the highest value for the lowest price. [Baptista et al. 2016]

Utility maximization theory is widely applied to model decision-making. It is a key element of theoretical approaches explaining human behavior [Howes et al. 2014]. However, there are two main points of critique.

The first critique has to do with bounds on information processing. When faced with too many options, it is impossible for a human to process all these options. Humans do not have the cognitive capabilities to view all houses listed for sale in a reasonable time. Therefore, they probably do not bid on the house that would give them the highest utility. Often, decisions are suboptimal. Humans pick the best option from a selection of all listed houses.

There is an easy way to tackle this in our models: limit the options given to the agents [Howes et al. 2014]. How to limit options is a modeling choice. The choice depends on the domain. For the housing market, we could remove options at random, but probably this is not realistic.

A more realistic choice is the brochure concept introduced by [Yun and Moon 2020]. In this model, every household has preferences for what type of house it wants to live in. These preferences are region and maximum and minimum price. When a household is looking for a new house, it receives a brochure. The brochure contains an arbitrary amount of houses matching the household's preferences.

Another critique of the theory is that humans might not be as rational as the theory assumes. We can never model all components that influence the utility of an option. And even if we could model all influences, sometimes behavior is irrational. We can mitigate this by including a random component in our utility calculations. We discuss this in the next section.

5.3.1 Random utility maximization

An influential extension of utility maximization theory that uses a random component is random utility maximization (RUM). Like utility maximization theory, RUM also assumes human decisions try to maximize a utility. This happens within constraints of available resources and cognition [Arentze et al. 2013]. For the housing market, these resources are, for example, the income and wealth of a household. The cognition determines which listed houses a household views as an option. We see that limiting human cognition is already a key part of RUM.

With RUM, the utility function includes a random component. This random component captures limitations in models. These limitations stem from limited information about how the world works. We do not understand all workings of the human brain. This results in modelers not being able to fully model the decision-making process. To overcome this, we make the utility function stochastic. Besides this, the random part also accounts for parts of the system that we did not model [Arentze et al. 2013]. For example, we can choose not to model connectivity to the internet of houses. However, this may very well influence a household's decision when they decide which house is the best. We can view the random component as capturing this and other unmodeled factors. This is, of course, only a very rough estimation of such factors. We acknowledge they exist and account for them.

With a random component, the utility function consists of two parts. A deterministic or observed part. And a random or unobserved part. The observed part is usually a linear combination of some variables. Like, monthly costs, leftover money, house location, and house quality. The unobserved part follows a distribution. A normal distribution is a good choice for this as it is natural for behavior [Susmel 2014].

Both parts of the utility function have parameters we need to estimate. We can do this manually through experimentatation. We alter the parameters until we get a model that produces realistic data.

A more sophisticated method uses a local search algorithm. The key idea behind this method is to run a simulation with a set of parameters. We check how well the resulting data fits some real-world data. Next, we update the parameters in the direction that improves this fit [Susmel 2014].

5.4 Modeling demographics

Most models have some form of demographic modeling. This tracks how the population evolves over time. [Baptista et al. 2016] describes a general approach most models follow. It models age, income, and wealth. As this model models households and not persons, the age refers to the household reference person. Often this is the oldest (working) person in a household [Insee 2019].

Age influences the likelihood that a household dies. The older the household, the more likely it is to die. Models remove dead households from the simulation.

[Baptista et al. 2016] has the concept of inheritance. This inheritance transfers the belongings of a dead household to a random household. The belongings include owned houses and wealth.

Besides death, [Baptista et al. 2016] also has the birth of new households. This resembles children leaving their parents and divorces. New households get an age, income percentile, and starting wealth. The income percentile remains constant for the lifetime of a household. This percentile determines the income a household gets based on its age. So whilst the income percentile is fixed, the actual income changes with age.

As time progresses, a household's wealth can increase by saving money. It saves all money it does not spend on housing and non-housing consumption. The non-housing consumption consists of essential consumption and non-essential consumption. Essential consumption is the same amount for each household. It represents the cost of staying alive, i.e. buying food. Non-essential consumption varies per household, with wealthier households spending more.

[Zhuge and Shao 2018] has the most detailed population model. They model age, death, income, education, emigration, employment, birth, friends, marriage, divorce, and immigration. Changes in any of these categories can cause actions in the housing market. Getting a new job may lead to buying a more expensive house. Losing a job can lead to losing a house. Or a divorce causing the search for a new house.

5.5 Construction

Construction is often implemented as a proto-agent in models. There is no explicit modeling of construction companies. Instead, at the end of each time step, construction balances the number of houses with the population size. For example, [Baptista et al. 2016] tries to keep the ratio of houses to households constant.

More explicit modeling would have construction companies that try to maximize their profit. The amount of profit is their utility. They make money by selling houses and lose money by building them. They aim to build the house types with the highest demand at the most popular locations.

5.6 Changing houses

Changing houses consists of two parts. First, when do households decide to move out of their current house? Second, how do households get a new house?

5.6.1 Moving out

The moment households decide to move out, depends on their current situation. Are they renting or do they own their home? When renting, the general modeling approach is to have contracts. Contracts determine how long a household rents a house. When the contract expires, the household looks for a new house [Baptista et al. 2016].

For moving out of a bought house, multiple approaches exist. One of these we discussed in the previous section, where demographic changes trigger movement. Another, often used, approach is to use a movement probability. [Baptista et al. 2016] calculates a probability based on how long a household is living in its current house. In this model, households sell their home, on average, every 11 years.

5.6.2 Bidding process

We now look at how households get a new house. Usually, these processes are the same for both the renting and buying markets. In our descriptions of approaches, we will focus on the buying market. However, we can easily extend these approaches to the rental market. We discuss three different approaches, the double auction, buyer-seller matching, and first-come first-serve.

Double auction

[Baptista et al. 2016] uses a double auction mechanism. The double auction consists of two phases. Before these phases take place, all house buyers give their bid price. Also, all house sellers give their asking price. For this discussion, it is not relevant how we determine these prices.

Now the first phase of the auction takes place. The model matches buyers and sellers. It matches a buyer with the seller that sells the best quality house it can afford. This results in some sellers getting matched to one or multiple buyers. Some sellers will not find a match. This happens if their asking price is too high or their house quality is too low. Also, buyers will not find a matching seller if their bid is too low.

Now the second phase happens. In this phase, sellers sell their house to buyers. Sellers matching with only one buyer, sell their house to this buyer. Sellers matching with more than one buyer, get to increase their price. Out of the buyers that can still afford the house after this increase, one is randomly selected to buy the house.

Buyer-seller matching

Buyer-seller matching is a more simplistic approach than the double auction. It is used by [Evans et al. 2021]. Every buyer buys the most expensive house it can afford. To do this, again, all buyers and sellers list their price. We now sort these prices from high to low for both buyers and sellers. For each bidder, starting with the highest one, we pick the most expensive house it can still afford. We immediately sell this house to this bidder. We continue until we find a bidder that cannot afford any of the houses. [Evans et al. 2021] also includes a 20% random chance of rejection when a house gets sold. They do this to model rejections because of external influences.

First-come first-serve

An even simpler approach is the first-come first-serve approach. [Yun and Moon 2020] uses this in combination with their brochure approach. Every time steps the model iterates through all buyers. A buyer now receives a brochure. If the brochure contains an affordable house, the buyer will buy it. In case there are multiple affordable houses, it buys the house with the best quality. If there is more than one of those, the buyer buys the cheapest. Each time step, the model randomizes the order it iterates over the buyers. It does this to not let the same household have the first choice every time.

Chapter 6

Base Model

In this chapter, we discuss the *base model*. The *base model* is the model to which we compare our experiments. For each experiment we extend the *base model* with a policy, creating a new model. To compare different models we run each model multiple times. This gives us a set of samples for each model. We analyze the differences between the different models in Chapter 8. But first, we describe how the *base model* works.

For this description, we use the ODD protocol as we discussed in Section 4.4.3. We do not cover the submodels element in this chapter. Instead we cover this element in Appendix A. The basis for our model is [Baptista et al. 2016]. However, we extend it by adding a real social rental market. We also make our model utility-based. Therefore, agents make decisions based on the utility of a future situation.

6.1 Purpose

We want to use the *base model* and the experimental models that use it as a basis to answer our hypothesis. This hypothesis is: policies that reduce the m^2 per person have a major role in countering the effects of the housing crisis. Our models must be capable of measuring these effects. These effects are high rent and buying prices, a strong lock-in effect, and a high housing shortage. The model must also measure the space per person a household uses. We need a way to compare the experimental models to the *base model*.

6.2 State variables and scales

Each time step in the model is one month. Some of the actions in the model repeat every month. Others run only once per year. The model starts running in 2012 and runs until 2060. To have a single model run not take up too much time, we scale the number of agents. For our runs, we use a scale factor of 12,500. This means one household in the model represents 12,500 households in the real world. The same applies to houses.

6.2.1 Households

Household agents are the moving parts of the model. They represent the population and move in and out of houses. Households represent a group of people living together, this can be someone on their own or a family. Households have the following properties: age, size, income percentile, income, wealth, and waitlist time. They also have a contract, that states in which house they live. Lastly, households have a *want to move* property. This is a binary variable that indicates if they want to move out of their current house.

Like in [Baptista et al. 2016], the household age represents the age of the household reference person. We assume households have a minimum age of 15 and a maximum age of 100. Households have a size assigned at their creation. The size indicates how many people are part of the household. A household's size can change during its lifetime. We describe how this work in Section 6.3.2.

When a household's size changes it reevaluates its current housing situation. The reevaluation happens with a utility function. This function scores the housing situation. The score depends on the quality of the house and the amount of money a household saves whilst living in the house. The utility also depends on the space per person of the household in the house. If a household's size changes both the amount of money it saves and the space per person change. This affects the utility. If the utility after the size change is lower than before, the household wants to move.

Households also use the utility function to make decisions on which house to buy or rent. We discuss this utility function in more detail in Section 6.4.1.

We assign households to an income percentile at birth. Households stay in this percentile for their entire lifespan. As in [Baptista et al. 2016], this fixed percentile means we do not model shocks like unemployment. Together with a household's age, the income percentile determines their income. Households have an income that puts them in the correct percentile for their age. As a result, a household's actual income changes as it ages. We base our calculations on real-world data. At birth, households also get some starting wealth. This wealth also corresponds to their age and income percentile. As a household saves and spends money during its lifetime, its wealth changes over time.

We also assign households a waitlist time at their birth. This waitlist is either their age minus the minimum age of 15. Or it is zero. We pick one of these options from a distribution. We use the waitlist time to decide which household gets a social house. We only need it when more than one household applies for the same social house. This waitlist time increases every year.

Households have a contract. This contract registers their current living situation. A contract can be either a buying, rental or social rental contract. All contracts store the monthly payment. This payment is either for the mortgage or the rent. A buying contract also stores how much mortgage there is left to be paid.

Table 6.1: State variables households

State variable	Value	Unit
Age	$15 \leq \text{age} \leq 100$	years
Size	$1 \leq \text{size} \leq 5$	amount of people
Income percentile	$0 \leq \text{percentile} \leq 100$	percentile
Income	$\text{income} \geq 0$	euros
Wealth	*integer*	euros
Waitlist time	$\text{waitlist time} \geq 0$	years
Contract		
Want to move	*true* or *false*	

6.2.2 Houses

Houses are the other important part of the model. Houses are stationary agents, so as described in Section 4.3.2 they are part of the environment. They have a size, a quality, and a type. The quality is a proxy for the location and the condition of a house [Baptista et al. 2016]. We choose to model size explicitly, instead of including it in the quality parameter. We do this to track how much space households use on average. The quality of a house cannot change during a model run. In the *base model* the size of a house also cannot change.

A house's type indicates whether it is for social rent, private rent, or buying. Social rental houses can become private rental houses and vice versa. Rental and buying houses cannot be converted into each other. We base the percentage of houses for rent and buying on real-world data.

Houses that are for buying can have an owner. This owner indicates which household currently owns the house.

Table 6.2: State variables houses

State variable	Value	Unit
Size	> 0	m^2
Quality	$0 \leq \mathit{score} \leq 10$	score
Type	*social rent*, *private rent* or *buy*	
Owner	a household	-

6.2.3 Markets

A market agent keeps a list of all listed houses in a specific submarket. These submarkets are the rental and buying market. The buying market lists all buying houses. We list social and private rental houses on the rental market. Households interact with these markets to rent, buy or sell a house.

A market stores all transactions. For each sale or rental agreement, it stores the size and quality of the house and the price of the transaction. The price is either the monthly rent or the buying value. We use these transactions to fit a linear regression model. This model predicts the value of a house based on its size and quality. When we list a new house on a market, we use the prediction model to predict its market value.

Table 6.3: State variables markets

State variable	Value
Listed houses	list of houses with their list price
Type	*rent* or *buy*
Transactions	list of sales
Market price regression model	

6.2.4 Bank

The bank is a proto-agent in our model. There is one bank. From this bank households loan money for a mortgage. The bank always has enough money to award the mortgage. It only checks if the household has a high enough income to repay the mortgage. All mortgages have the same run-time and interest rate.

6.2.5 Government

Like the bank, the government also is a proto-agent. It manages the liberalization threshold and pays out rent allowance. To pay out the rent allowance the government needs money. To get this money it raises taxes. Each household pays the same percentage of its income towards this tax.

The liberalization threshold determines which houses are for social rent. Houses with a rent price below the threshold are for social rent. Houses with a price above the threshold are for private rent. The threshold is set such that the percentage of social houses does not vary too much.

6.2.6 Construction

Construction is also a proto-agent. The construction agent creates new houses. The amount of houses created for each year is the same for every model run. This makes it easier to compare different runs and models. We draw both the size and quality of a new house from a distribution. These distributions are also the same for all models and runs.

6.3 Process overview and scheduling

We now describe the processes in the model. We divide the processes into two categories. Processes that run every month and those that run only at the end of each year. Figure 6.1 shows the processes and how the model schedules them. In this section, we only give a short description of the processes in the figure. We give the full description of all processes in the model in Appendix A.

6.3.1 Monthly processes

Gain wealth

Every month households gain wealth by saving money. The amount of money a household saves is non-linear in income. The higher income it has, the more it relatively saves.

How much money a household saves depends on its size, its housing cost, and the current tax rate. These are the obligatory costs. The size of a household determines its essential costs. This is money spent on essential items such

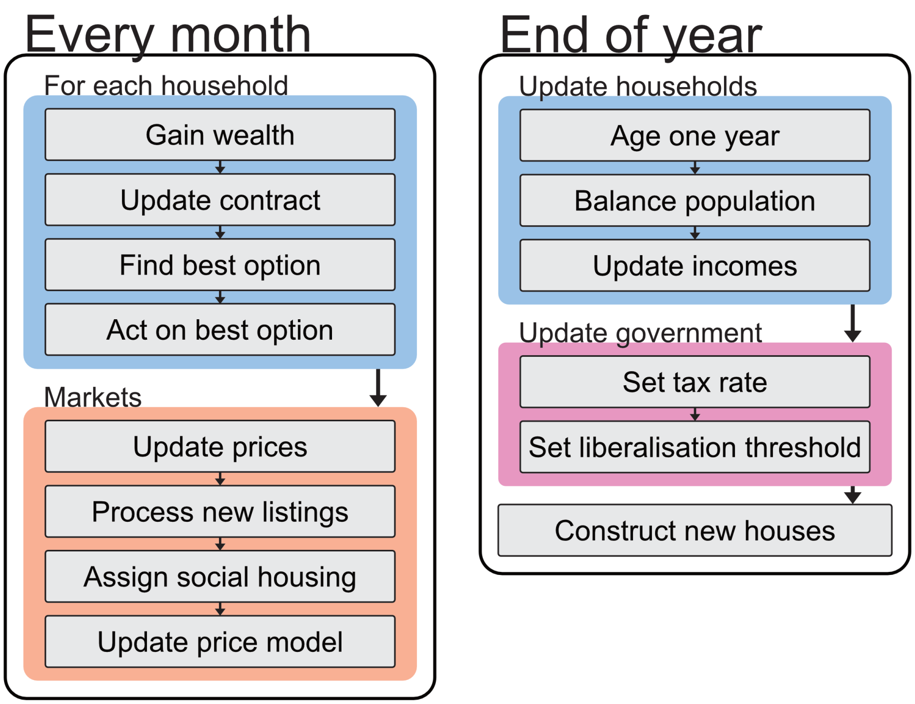

Figure 6.1: Schedule of processes in the model.

as food or clothing. It increases linearly with a household's size. The housing costs are what the household pays either for rent or their mortgage. A household also pays a portion of its income for taxes. How much depends on the current tax rate.

If a household's income is not high enough to cover the obligatory costs, it uses its wealth to cover the gap. If it has no wealth remaining, the household is evicted.

If a household's income is higher than the obligatory costs, it spends a part of the remaining money. This resembles money to spend on, for example, eating out or leisure activities. This is where we introduce the non-linearity. As the amount of remaining money goes up, the amount spent increases slower.

Update contract

Each month we update the contracts for all households. For each contract, we increase the run time by one month. For buying contracts, we decrease the remaining mortgage that needs to be paid off. We do this each month for a contract until the remaining mortgage reaches zero. After this, the cost of the contract is set to zero.

Look at markets

Every month households look at the buying and rental market to see if they can improve their utility. They determine the best option for both markets using random utility maximization (see Section 5.3.1). They compare these best options and act on the one with the highest utility in the next phase.

When reviewing houses, households do not see all listed options. Like [Yun and Moon 2020], we use a brochure to limit the number of options. The maximum number of options a brochure shows is a parameter we control. If the number of listed houses on the market is above the brochure limit, we make a random selection.

For the buying market, the brochure only lists houses with an asking price that is lower than the maximum mortgage a household can get.

For the rental market, we make two brochures. One for social rental houses and one for private rental houses. A household selects the best option from both brochures. The best option for the rental market is the option with the highest utility of these two best options.

Act on best option

How a household acts on the best option depends on its current situation. If the best option is not an improvement, the household does nothing. Otherwise, the household moves into the best option. There are two exceptions to this.

First, if a household is currently buying its house, it first has to sell the house before moving into a new house. In this case, we use the fact that there is a better option available as a trigger. This trigger tells the household that there exists a better house and that the household can improve its utility. The buying household wants to do this. Therefore it lists its house for sale. When the house is sold, the household becomes homeless and starts looking for a new house.

The other exception happens when a household wants to move into a social house. In this case, the household puts itself on the list of households interested in the social rental house. At the end of the month, we move the household with the longest waiting time into this social rental house. So, a household does not necessarily move into this house.

Update listed prices

A house listed on the buying market lowers its asking price by 3% every month it is not sold. The asking price can drop below the point where the selling household can no longer pay off its current mortgage. Note that a household may use its wealth to help pay off the mortgage.

Houses listed on the rental market decrease their rent by 3% every month they are not rented out. This only applies to houses that are for private rent. The listed value of social rental houses does not decrease. If the value of a private rental house decreases below the liberalization threshold it remains a private rental house. It does not become a social rental house.

Process new listings

We wait with putting newly listed houses on the market until the end of the month. When listing a house we first determine its market value using the market price model. We then list the house 4% above this value on the market.

For rental houses, this is also when we decide if a house is for social or private rent. If the market rent price of a house is below the current liberalization threshold, we list the house for social rent. Otherwise, we list the house for private rental. As stated above, this affects whether the rent price decreases each month or not.

Assign social housing

We resolve how to assign social housing at the end of each month. This means that all households had the chance to show an interest in the social houses. For social houses with at least one interested household, the household with the longest waiting time moves into the house.

Update price model

At the end of each month, we refit the parameters of the market price regression model, for both the rental and buying market. We do this by using the last 100 transactions as input data.

6.3.2 Yearly processes

Ageing

Every year all households get one year older. We also increase the waitlist time by one year. Households that get older than 100 years die. When a household dies, we list the house it occupied on the market.

Balance population

For each year in a model run, we have a joint distribution for the age and size of households. Each distribution tells us what percentage of households have a certain age and size at the start of a year. We call an age-size combination a slot. We discuss the dataset that provides the information for every slot in Section 6.5.

At the end of each year, we modify the current population to match the next year's distribution. We do this by moving around households between slots. We take households from slots that have too many households and put them into slots with too few households. We only move households around by changing their size. We never move a household from one slot to another by changing its age.

For example, if we have two 22-year-old households. One of size four and one of size three. And the distribution tells us we need two 22-year-old households of size three. We make the four-person household a three-person household by decreasing its size by one. After a household's size changes, we compare its new utility with its old utility. If the new utility is lower than the old utility, we mark that the household wants to move.

It might be impossible to achieve the correct distribution by moving households around. We then create and delete households from slots to get to the correct distribution. For slots with too few households, we create new households with the correct age and size. We delete households from slots with too many households. Deleted households die. We list their house on the market.

Updating income

Every year, we update a household's income based on its age and income percentile. Based on [Baptista et al. 2016], we draw the actual income from a joint age-income distribution. We do this in such a way that every household falls into its assigned percentile.

Set tax rate

At the end of each year, we calculate the new monthly tax rate for the next year. As stated above, the government uses the raised taxes to pay out the rent allowance. To calculate the tax rate, we first calculate how much rent allowance we award to all households in a month. We divide this number by the total monthly income of all households together. This gives us the tax rate for the next year. This tax rate does not raise the exact needed amount of money each month, but it is a good enough approximation.

Set liberalization threshold

We use the liberalization threshold to determine if a rental house is a social or private rental house. We want to have between 50% and 75% of all rental houses to be for social rent. If at the end of the year this is not the case we change the liberalization threshold. If less than 50% of all rental houses are for social rent, we raise the threshold. This way the rent of more houses will be below the threshold, turning them into social rental houses. If more than 75% of all rental houses are for social rent, we lower the threshold.

Construct new houses

At the end of each year, we build new houses. As stated above we draw the size and quality of a new house from two distributions. One part of the new houses become available for the rental market. The other part becomes available on the buying market. We fix this ratio for all runs. We immediately add new houses to their respective markets. We set the price the same as for newly listed existing houses. This means the price will be above the current market price for such a house.

6.4 Design concepts

6.4.1 Human behavior

Households make decisions by calculating and comparing utilities. Given a list of potential houses, they calculate the utility for each option. They also calculate the utility of their current situation. The utility depends on how much money a household saves each month when living in the house. It also depends on the space per person in the house and the house quality. After calculating all utilities, they pick the option with the highest utility.

We calculate a utility with random utility maximization. As described in Section 5.3.1, the utility function consists of a deterministic and a stochastic part. We use the following function for the deterministic part:

$$\begin{aligned} \text{deterministic} = 0.0005 \cdot \text{saves} + 0.1 \cdot \sqrt{\text{space per person}} \cdot \text{house quality} \\ -\text{movement cost} - \text{homeless penalty}. \end{aligned}$$

Where *saves* is the amount of money a household saves each month. The amount depends on a household's income. As well as the housing costs of the house under consideration. We described how to calculate this amount in Section 6.3.1.

Space per person is how much space per person the household has in the house. We calculate it by dividing the house size by the household size. The *house quality* is the quality of the considered house. As stated in Section 6.2.2, the quality is a proxy for the location and condition of a house.

Movement cost is a cost subtracted from the utility. It models a barrier on moving into a new house. The goal is to stop households from changing houses each month. Without this cost, a household moves every time it finds a better option. With this cost, the utility of other houses is lower if a household recently changed houses. The cost depends on how long a household lives in its current house and decreases over time. When calculating the utility of the household's current house, the movement cost is zero. We use the following formula to calculate the movement cost:

$$\text{movement cost} = \begin{cases} 1.5 & \begin{array}{l} \text{, if house is not current house} \\ \text{and not homeless} \\ \text{and time in current house} = 0 \end{array} \\ 1.5 * \frac{12}{\text{time in current house}} & \begin{array}{l} \text{, if house is not current house} \\ \text{and not homeless} \end{array} \\ 0 & \text{, otherwise} \end{cases}.$$

Where *time in current house* is the number of months a household is living in its current house.

We subtract the *homeless penalty* if a household is homeless. The value of this penalty is five. When calculating the utility of a homeless household, the space per person and quality is zero. The housing costs of this house are also zero.

To get the utility we combine the deterministic utility with a value $\mathcal{X}$ we draw from a normal distribution $\mathcal{N}(0, 1)$. The final utility function now looks like this:

$$\text{utility} = \text{deterministic} + \mathcal{X}.$$

6.5 Initialization and input

The initialization of the model takes two steps. Creating the agents and calibrating the model. For creating the agents we rely on real-world data. This data provides us with the correct distributions for household size and age, income, wealth, and house size.

Creating households

To create the household we use a dataset with a joint distribution of household size and age. We do not only need such a distribution when creating the households. We also need this data for balancing the households at the end of each year. Therefore, we need data for each year from 2012 until 2060. To get this we combine historic data [Centraal Bureau voor de Statistiek 2021d], with a prognosis for the future [Centraal Bureau voor de Statistiek 2021e].

For each year, this dataset gives us data on each five-year age group from 15 to 20 years up to 95 to 100 years. Each age group is divided into 5 size groups. From size 1 up to 5. We call each age-size combination a slot. A slot thus consists of a five-year age range and a household size. For each slot, the data gives us how many households to create.

When creating a household for a slot, we randomly pick an age that lies in the age range of the slot. We assign the new household the size of the slot. Next, we assign the household an income percentile. We pick the percentile from a uniform distribution between 0 and 1. Based on this percentile and a household's age we give it an income and starting wealth.

The datasets for wealth ([Centraal Bureau voor de Statistiek 2021f]) and income ([Centraal Bureau voor de Statistiek 2021g]) are very similar to each other. The data consists of ten-year age ranges, starting at 15. For each age range, the data give us which percentile of households has a certain average income or wealth. Based on this average, we can generate the actual income or wealth. We discuss this in more detail in Appendix A

Creating houses

Houses require a size and a quality. We could not find a dataset for the quality of houses. Therefore we pick it from a uniform distribution between 1 and 10. For house size, there exists a dataset [Centraal Bureau voor de Statistiek 2021c]. This dataset gives us how many houses there are in a certain surface area class. Each class provides us with a lower and higher bound for the size of the houses in the class. The classes and their bounds are 2-15m^2, 15-50m^2, 50-75m^2, 75-100m^2, 100-150m^2, 150-250m^2, 250-500m^2, 500-1,0000m^2. When creating a house in a certain surface class, we pick its size randomly between the lower and higher bound of the class.

Lastly, we assign the house to either the buying or rental market. 45% of all houses are for rent and 55% are for buying. We base these percentages on the real-world distribution of rental and buying houses, as discussed in Chapter 2.

Calibration

We calibrate the model, by running the model for a certain amount of months. During the calibration, we do not advance time. We run the same month over and over again. Thus we only run the actions in the *every month* column of Figure 6.1. Note that we skip the gain wealth action. By calibrating we aim to get an initial assignment of households to houses.

This process generates new transactions each month. We use these transactions to calibrate the market price prediction models. This way these models better predict the market prices at the end of the calibration with respect to our data.

Construction

As stated in Section 6.2.6, the number of constructed houses for each year is the same for every run. We use two datasets to make the list of constructed houses per year. One for years in the past and one for years in the future.

For past years, we use historic data of how many houses were constructed in a given year [Centraal Bureau voor de Statistiek 2022b]. For future years, we base how many houses we construct on the expected housing shortage for that year [Dutch Ministry of the Interior and Kingdom Relations 2021a]. By combining the expected shortage with the number of households and houses in the model, we calculate how many houses we need to construct to achieve the expected shortage in the model.

6.6 Analyzing results

6.6.1 Metrics

At the end of each year, we gather metrics about the model. We need these metrics to analyze our policies in Chapter 8. For most of the metrics, we track the average per year. For the other metrics, we create a histogram of their distribution for the last year of a run.

Buying price

To see how the buying price develops over time we use the market price prediction model of the buying market. Using this model we predict the value of a reference house at the end of each year. This reference house has a size of 120 m^2 and a quality of five. We use this size because the average house size in the Netherlands is 120 m^2. We do not have data on the real-world quality of Dutch houses, but in the model, the average quality is five. Therefore, we pick this value for the quality of the reference house.

We could also use the average sale price for the past year to track the buying price, but this has a disadvantage. Remember that the buying price depends on house size and quality. If in a year only large, high-quality houses are sold. We expect the average price to be higher than when we only sell small, low-quality houses. Therefore, when the average transaction value goes up, we do not know the cause. Does the price go up because buying a house got more expensive or because only expensive houses are sold?

Rental price

To track the average rent, we calculate the average amount households pay for rent. We do not track the average rent of houses that got rented out in the last year. As with buying price, this value could be biased because only expensive houses were rented out.

Housing shortage

The housing shortage is the number of houses we need such that there is a house for each household. This number can be negative. This happens when there are more houses than households. We scale the shortage by the number of households. This way we can more easily compare shortages between years [Dutch Ministry of the Interior and Kingdom Relations 2021a].

$$\text{shortage} = \frac{\#\text{households} - \#\text{houses}}{\#\text{households}}.$$

Space per person

For the space per person, we look at both the average and the histogram. We calculate the space per person by dividing the size of the house a household lives in by its size. For the average, we calculate this for all households and then take the average. For the histogram, we do not average over the model runs but instead accumulate over all runs. So a histogram shows the result of all runs together. We show an example of such a histogram in Figure 6.2.

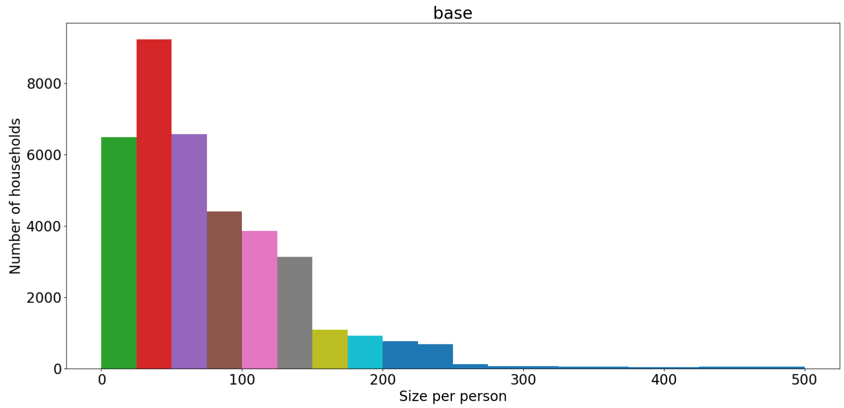

Figure 6.2: Example of space per person histogram

Lock-in effect

For the lock-in effect, we look at three metrics. Homelessness, *want to move count* and *want to move time*. Homelessness tracks how many households are homeless. *Want to move count* measures how many households want to move. Homelessness and *want to move count* are related. If a household is homeless then it also wants to move.

Want to move time measures how long households that want to move have to search before they move into a new house. For example, say the average *want to move time* is three months. This means that on average households, that want to move, spend three months searching for a new house.

The *want to move time* is the most important metric in analyzing the lock-in effect. For example, say we have 100 households that want to move and the average *want to move time* is zero months. This is a less strong lock-in effect than when ten households want to move, and the average *want to move time* is 16 months.

Utility

We measure the average utility to see how a model changes how happy households are with their house. If a model counters all effects of the housing market but also lowers the utility it might not be such a good model.

House size histogram

Like the space per person histogram, we make a histogram of the size of all houses. This way we can track how a model changes the distribution of houses. We can also observe which types of houses are added.

Listed houses

We track how many houses are listed on a market each year. We distinguish between buying and rental houses. We also track the size of the listed houses each year.

6.6.2 Statistical test

We need a method to see if our experimental models counter the effects of the housing crisis. We do this by comparing the relevant metrics with the *base model*. To quantify the differences we need a statistical test.

Most metrics show how the value of the metric develops over time. We run each model 50 times. This means we get 50 time series per metric. To check if two models give similar results for a metric, we need to compare the sets of time series. We could not find a method in the literature that does this directly. Therefore, we manipulate the sets of time series to get data we can compare with a statistical test.

For this manipulation, we first calculate the average time series of this metric for one of the models. Next, for both models, we calculate the distances between this average and the observed time series. For this, we use the euclidian distance measure [Lee and Kam 2014].

After calculating the distances, we have two sets of distances. One with the distances between the time series of model one to the average of model one. The other has distances between the time series of model two and the average of model one.

We use a statistical test to check if these two sets of distances are from the same distribution. If they are, we say that the models are similar for that metric. If they are not, we state that the models are different for that metric. The statistical test we use is the Mann-Whitney U test [Mann and Whitney 1947].

If we need to compare two distributions, for example in the case of the space per person distribution, we also use the Mann-Whitney U test.

6.6.3 Table

To give an overview of our results, we compile the results of our experiments into a table. This table shows for the space per person, the effects of the housing crisis and the utility, how the value of the experimental model compares to the *base model*.

This makes it easy to see how the model performs. We use symbols to indicate how the models differ from each other. We use the following symbols: - -, -, ?, =, +, and ++. Table 6.4 shows the meaning of each symbol. We give these symbols a color to indicate if the change they represent is positive, neutral, or negative. We use green if the change is positive. Orange if the change is neutral. And red if the change is negative. Table 6.5 shows for each metric which change results in which color.

We determine what symbol we assign to a metric based on the statistical test described above and a visual comparison. If the statistical test says two metrics are equal, we show in the table that the metrics are equal. If the test says two metrics are different, we pick how they are different after visually comparing the averages and standard deviations of the two metrics.

Table 6.4: Explanation of symbols

Symbol	Explanation
- -	Result of experiment much lower than base model
-	Result of experiment lower than base model
?	Result of experiment different from base model
=	Result of experiment similar to base model
+	Result of experiment higher than base model
++	Result of experiment much higher than base model

Table 6.5: Color for each type of change per metric

Metric	Positive	Neutral	Negative
Average m^2 per person	-, - -	=, ?	+, ++
Space per person distribution	?		=
Buying price	-, - -	=, ?	+, ++
Average rent	-, - -	=, ?	+, ++
Number of homeless households	-, - -	=, ?	+, ++
Number of households that want to move	-, - -	=, ?	+, ++
Average want to move time	-, - -	=, ?	+, ++
Housing shortage	-, - -	=, ?	+, ++
Average utility	+, ++	=, ?	-, - -

Chapter 7

Results of the base model

In this chapter, we look at the results of the *base model*. We discuss the space per person and the effects of the housing crisis. We use the values of these metrics as a baseline. In the next chapter, we compare our experimental models to this baseline. We also discuss some shortcomings of the model and how they influence the results.

The results we present in this chapter are the averages of 50 runs of the *base model*. The scale factor we use is 12,500. So every household in the model represents 12,500 households in the model. The same applies to houses. This means we have around 600 households and houses in a model run. This is a small scale, which can cause a high variance in the results of the model. One household changing houses has a bigger impact when there are 600 households than when there are 60,000.

7.1 Space per person

Figure 7.1 shows the average space per person in the *base model*. The figure also plots the standard deviation around this average. We plot this standard deviation in all subsequent figures as well. For the space per person, we see the amount increase from 70 m^2 at the start to 80 m^2 at end of the model. In Section 2.2, we saw that in the real world the space per person is around 65 m^2 in 2019. So our model is similar to this.

7.2 Buying price

As discussed in Section 6.6.1, we track the buying price by predicting the market price of the same house each year. We plot this value in Figure 7.2. We see the price increase from around €100,000 at the start to around €200,000 at the end. This is not comparable to the real-world prices we discussed in Section 2.4.

We cannot use the price data in our model as a prediction for what prices in the real world will do. But, we can use it as a baseline for comparison with other models. This way we can check how other models influence the buying price. We do this in the next chapter where we analyze the results of our experimental models.

There are several reasons why the price data is not accurate. One reason is a lack of real-world transaction data. With such data, we could calibrate the market prediction models to be closer to the real world. However, different to [Baptista et al. 2016], we do not have access to real-world transaction data. Such a dataset has a record of all transactions during a certain period. For each transaction, it lists the value and the properties of the house. With such a dataset it is straightforward to calibrate the prediction models. Because we do not have such a dataset, we use a different approach. We calibrate the prediction models, with transactions generated by the model itself.

Another reason for the lower buying price is that we do not model investors. Investors are agents that buy houses to rent them out. Investors are either wealthy households or companies. Without investors, we miss a class of agents that competes in the buying market. So there is less competition in the market, leading to lower prices. A side-effect of not having investors is that paid rent disappears. Because nobody owns the rental houses, the paid rent leaves the model.

The last cause for the low buying price is the absence of a bidding system. In our model, we use a first-come first-serve approach for the auction. We sell the house to the first household that wants to buy it at its current

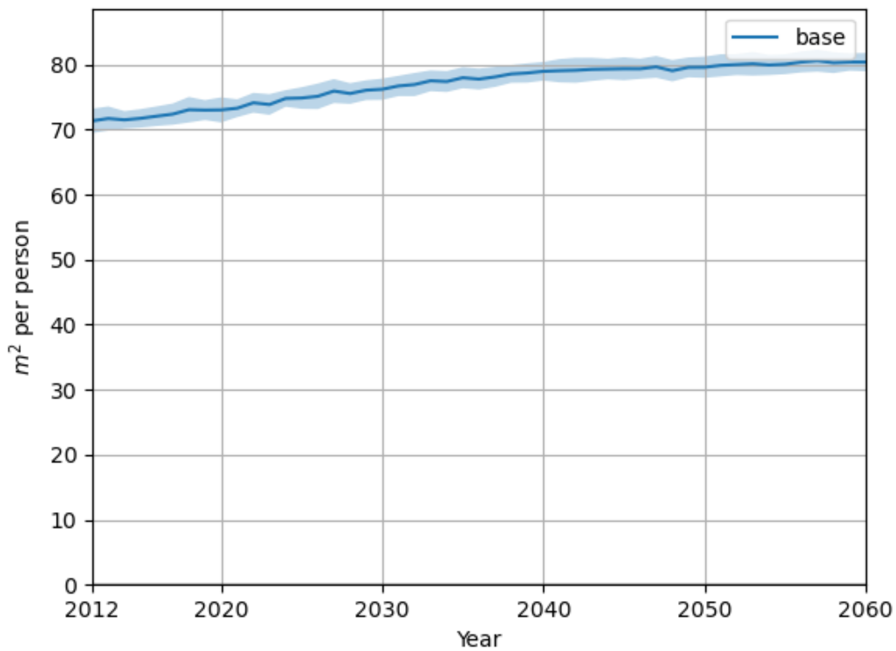

Figure 7.1: Average space per person in base model

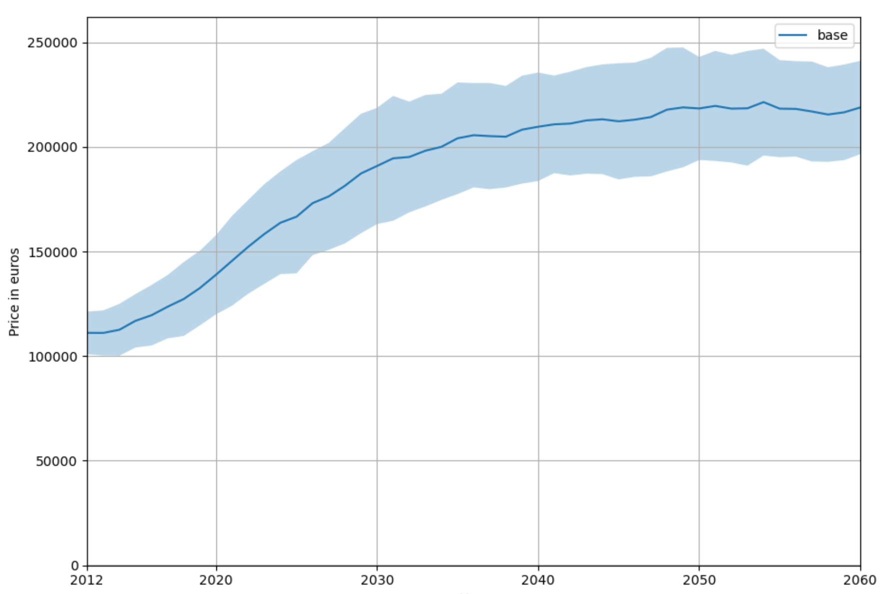

Figure 7.2: Buying price in base model

listed value. If we allow households to compete against each other for the same house, households bid against each other. As a result, houses sell above their listed value, leading to higher transaction values. This in turn leads to higher predictions for the buy value.

7.3 Rent price

As we can see in Figure 7.3, the average rental price remains constant throughout a model run. However, in Section 2.4, we saw that rent always increases in the real world. As discussed there, an increase in rent often follows the inflation rate. We do not model inflation, therefore there is no need to increase rent prices each year to keep up with inflation. This leads to the rent prices being somewhat constant. As with buying prices, this works fine for our model. We do not want an accurate prediction of rental prices. We do want to be able to compare the rents of different models with each other.

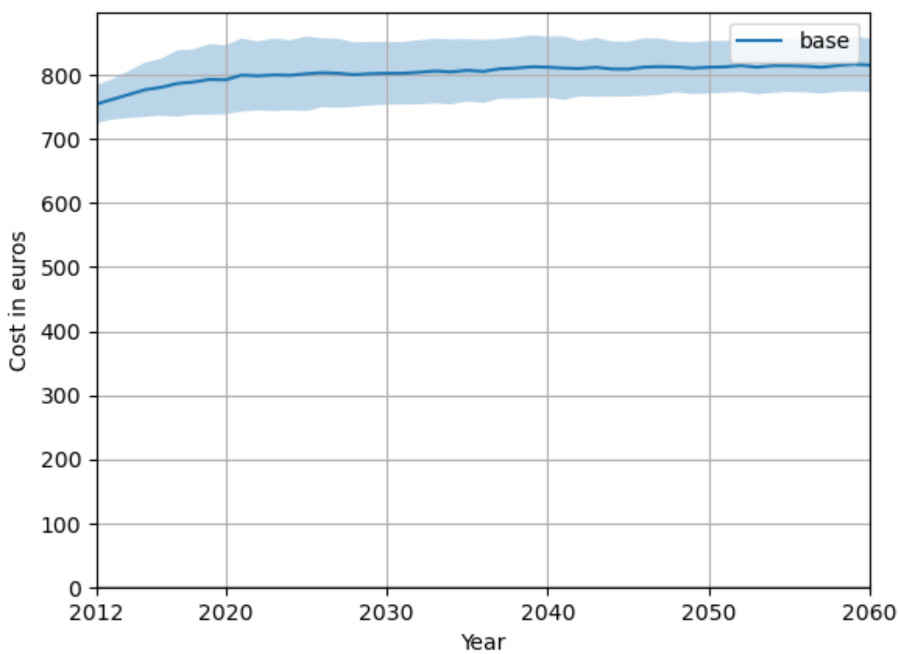

Figure 7.3: Average rent in base model

7.4 Lock-in effect

We measure the lock-in effect using three metrics: homelessness, *want to move count*, and *want to move time*. We plot those three statistics in figures 7.4, 7.5 and 7.6.

We see that there are around 50 households that want to move. This translates to 625,000 households in the real world. If we compare this to the real-world data described in Section 2.4, we see that this is much lower than the 1.7 million households that are actively searching for a new house. We discuss why this is below.

For the *want to move time*, we see it is around three months. This is also lower than what we observed in Section 2.4. Both the *want to move count* and *time* are not comparable to the real world. We cannot use these values to make predictions about the real world. We can use the metrics as a baseline, to see how our experimental models affect their value.

In our model, there are only two reasons for a household to want to move. It is homeless or its utility decreased after a change in size. This is not an accurate model of the real world. There are many reasons why a household decides it wants to move. For example, changing jobs, promotion, or inheriting wealth. These are all factors we do not model.

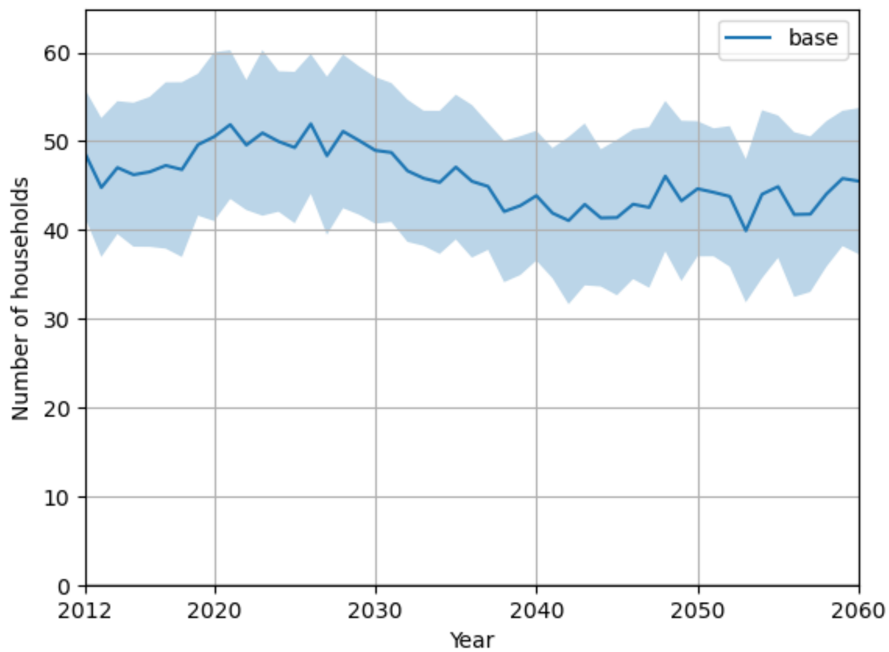

Figure 7.4: Homelessness in base model

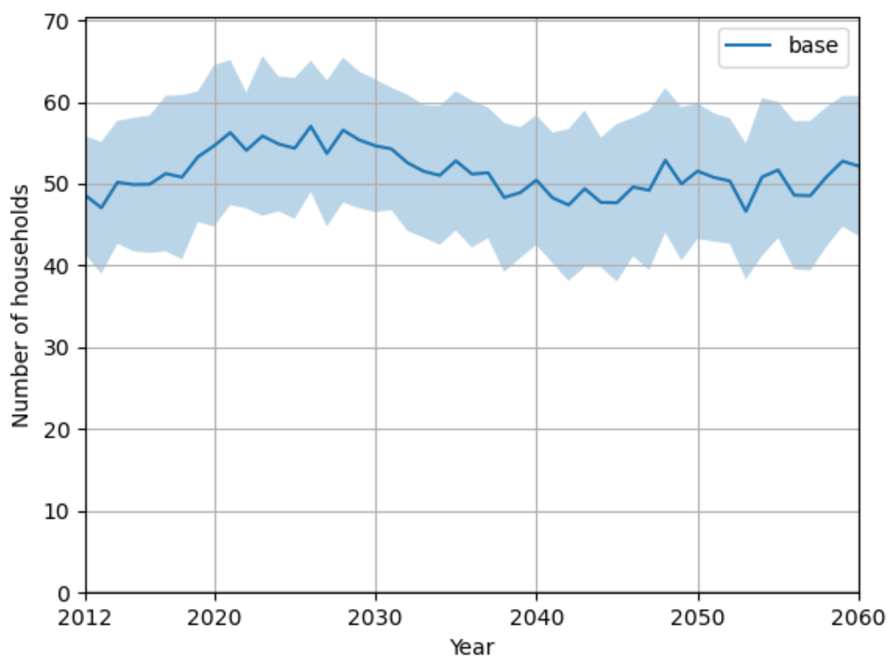

Figure 7.5: Want to move count in base model

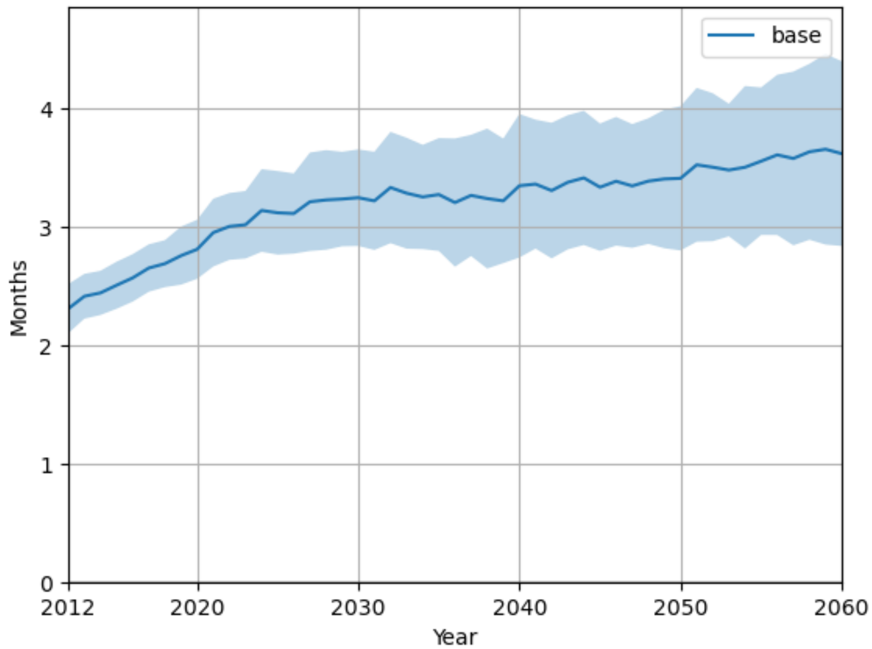

Figure 7.6: Want to move time in base model

We only model households. We do not model the persons they consist of and the relations between them. Therefore, we cannot capture events such as breaking up, getting a child, getting into a relationship, or someone dying. We use the changing household size as a proxy for all these kinds of events.

We also do not model income shocks. A household always stays in the same income percentile. Its income changes based on its age, but never drastically. This also means households never lose their job, or get a promotion. If we would model such changes, this can lead to situations where a household wants to move. For example, when a household loses its job, this leads to the household no longer being able to afford its current house.

7.5 Housing shortage

The housing shortage for the *base model* increases first to 4% and then decreases to around 1.5%. Figure 7.7 shows this. As discussed in Section 6.5 we base the number of constructed houses on real-world data. The number of households in the model also follows real-world data. Therefore, the shortage is correct with respect to those numbers.

The number of created houses each year is the same for all model runs. This means that the state of the rest of the model does not influence the amount of constructed houses. For example, we do not build more houses if there is high demand. Or fewer houses, if there is less demand. We made this decision because it is one less variable to take into account when comparing models. However, a more realistic model would have a dependence between the markets and the construction agent.

7.6 Utility

Lastly, we look at the utility of the model. Figure 7.8 shows how this number develops over time. We see the average utility increase a bit over time.

How we calculate utilities has two shortcomings. First, the utility function cannot look into the future. It does not take into account that buying a house, results in a household owning the house. The utility function treats rental and buying houses the same. For example, say we have a buying and rental house of the same size and value. Now

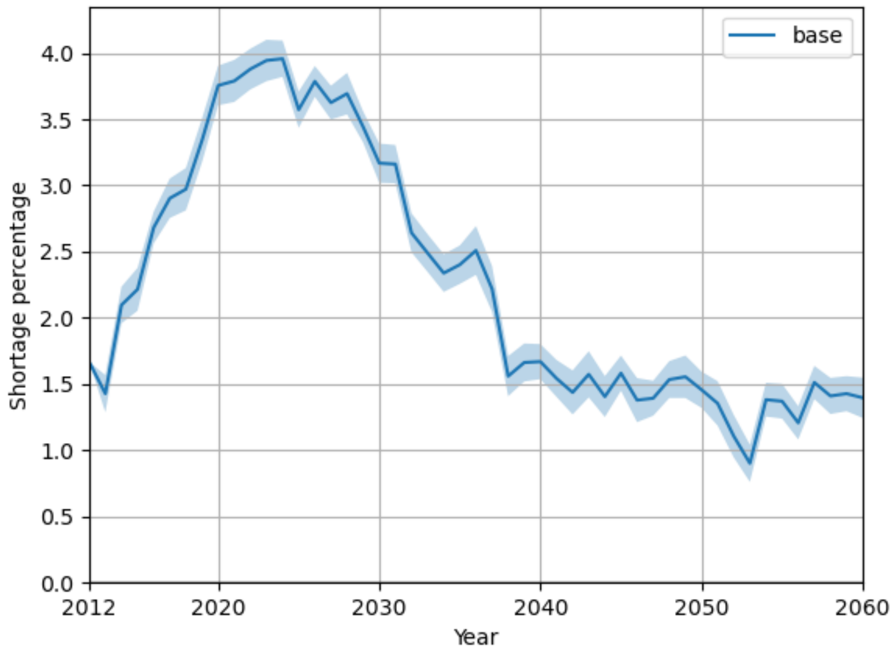

Figure 7.7: Housing shortage in base model

assume that the mortgage and rent for these houses are the same. If we ignore the stochastic part of the utility function, then the utilities are the same for a household.

However, when the household pays off the mortgage it no longer has any housing costs. If it was renting the house it would still have to pay rent. Also, the buying household gains wealth when it sells the house, which is also an advantage. If we would model these factors they should tip the utility in favor of the buying house. However, we only look at the utility of the houses for the next month because this is simpler to model.

The other shortcoming is that all households use the same utility function. With the same parameters. As a result, all households give the same importance to the components of the function. So how much money they save, how big their house is, and what the quality of the house is. We can imagine that in the real world this is not the case. For example, some households value saved money more than others. Or find quality more important.

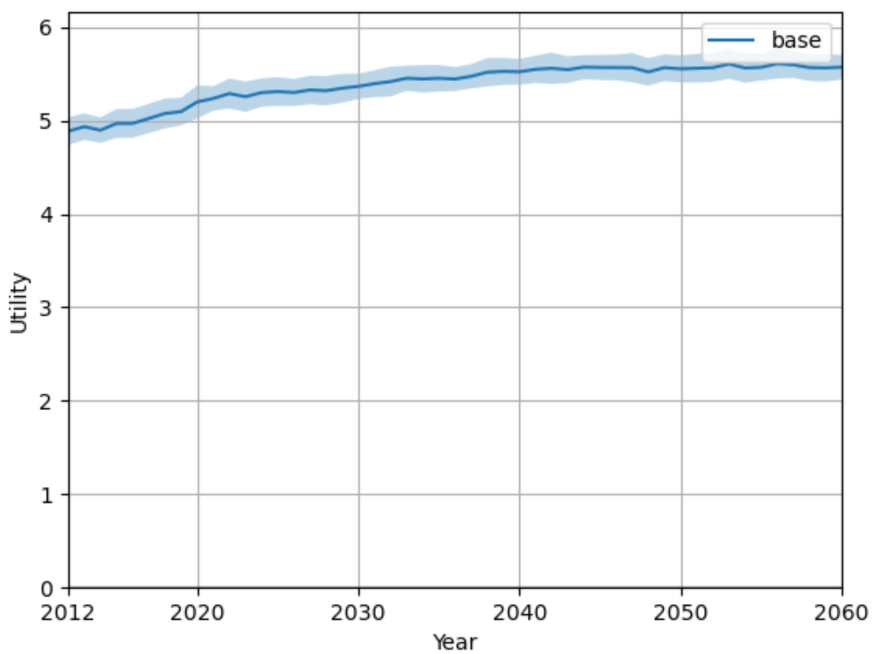

Figure 7.8: Utility in base model

Chapter 8

Results

To explore if our hypothesis is valid we experiment with ten different policies. In this chapter, we discuss the six most relevant policies. The other policies do not give us any extra information in validating our hypothesis.

Each policy should be capable of reducing the m^2 per person. We compare the results of each experiment to the *base model* as described in Chapter 6. This way we see if the policies counter the effects of the housing crisis. As discussed in Chapter 2, the effects of the housing crisis are high prices, high shortage, and a strong lock-in effect.

We look for the following changes in these effects and the space per person. For all effects, we want to see a reduction. The buy and rent price should be lower than in the *base model*. The housing shortage should decrease. This happens when a policy creates extra houses for households to live in. We add these extra houses to the houses already created through construction. We also want to observe a less strong lock-in effect. As discussed in Section 6.6.1, we analyze the lock-in effect by looking at homelessness, *want to move count* and *want to move time*. Of these factors, the *want to move time* has the biggest influence on the strength of the lock-in effect.

For the space per person, we want to see a reduction. Otherwise, the policy is not relevant to our hypothesis. We also look for a change in the distribution of the space per person. This distribution should become more centered around the mean.

Ideally, the policies do not reduce the average utility. The utility measures how households score their housing situation. If a policy results in a drop in this score, this means that households value the new situation less.

We observe these changes not only in the entire population. But also in subgroups of the population. We do this to see which groups a policy affects most. Or if there are groups that the policy does not affect. For example, a policy could only affect poor households and not rich households. We give an overview of the subgroups and their properties in Table 8.1.

Table 8.1: Subgroups of population

Group name	Properties
rich singles	household size = 1 and income percentile $\geq 80\%$
starters	household age ≤ 35
pensioners	household age ≥ 60
poor young families	household age < 50 and household size $>$ 2 and income percentile $< 33\%$
super poor	income percentile $\leq 20\%$
poor	$20\% <$ income percentile $\leq 40\%$
middle	$40\% <$ income percentile $\leq 60\%$
rich	$60\% <$ income percentile $\leq 80\%$
super rich	income percentile $> 80\%$

8.1 Taxing m^2 per person

The first policy we look at introduces a tax for each m^2 per person a household uses. The idea of this tax is that a household pays for the space it uses. The more space it uses, the more tax it pays. We believe this will motivate households to live in smaller houses. As the utility of larger houses decreases more than the utility of smaller houses. This way the policy should reduce the m^2 per person. We calculate the m^2 per person for a household in a certain house as follows,

$$m^2 \text{ per person} = \text{house size/household size.}$$

We experiment with three different taxation functions. A linear one, a quadratic one, and a function that does not tax the first 30 m^2 a household uses. For each function, we picked the coefficients after some initial experimentation.

We use a linear function because it is the simplest function that has an increasing tax as the space a household uses increases. The function we use is:

$$\text{tax} = 2 * m^2 \text{ per person.}$$

For the quadratic function we pick the following function:

$$\text{tax} = 0.075 * (m^2 \text{ per person})^2.$$

We use a quadratic function because the tax for using a low amount of space is less than for a linear function. After the intersection point of the two functions, the tax is higher than the linear function. For our functions this point is $26\frac{2}{3}m^2$. We suspect this motivates households even more to reduce the amount of space they use.

It might be unfair to tax a household if it only uses a small amount of space. Every household needs some amount of space to live in. The goal of the tax is not to have households pay money, but to limit how much space they use. We look at a taxation function that does not tax the first 30 m^2 per person a household uses. The function taxes every meter over this amount quadratically. It has the following form:

$$\text{tax} = 0.075 * max(0, m^2\text{per person} - 30)^2.$$

We add the space per person tax on top of the tax that pays for the rent allowance costs. However, for the rent allowance tax, we put the raised money back into the model in the form of rent allowance. We do not do this with the money raised by the space per person tax. This results in money leaking from the model. A direct effect of this is a lower average utility because households save less money. It also influences the buy and rent prices. The leaking money makes it hard to analyze the differences between the *base model* and this model. Does an effect happen because of the tax or because of the leaking money?

To fix this problem, we also put the money we raise from the space per person tax back into the model. We equally divide all raised money over the households. As a result, households that pay a less than average amount of tax gain money. Households that pay an above average amount lose money.

The taxation policy does not include a way to create new houses. Therefore we do not expect a change in the housing shortage.

Analysis

We show an overview of the results of this policy in Table 8.2. It shows that the linear taxation function has almost no effect compared to the *base model.* This taxation function only has a little effect on the *want to move time.* This function does not reduce the m^2 per person. For our hypothesis, we are only interested in policies that reduce the space per person. Policies that do not lower the space per person do not help us with answering our hypothesis. Therefore, the taxation policy with the linear taxation policy provides no insight into if our hypothesis is valid.

Table 8.2: Overview of results for taxation policy

	linear	quadratic	minimal needed space
Average m^2 per person	=	- -	- -
Space per person distribution	=	=	=
Buying price	=	++	++
Average rent	=	=	-
Number of homeless households	=	=	=
Number of households that want to move	=	=	=
Average want to move time	-	-	-
Housing shortage	=	=	=
Average utility	=	=	=

The other two taxation functions have similar effects. We focus the rest of our analysis on those two functions. Both functions reduce the space per person. The space per person drops by around 5 m^2 for both functions.

We see an increase in buying price, which means this policy does not counter this effect. For the rental price, we also do not see the desired effect by looking at the table. The quadratic function does not reduce the rent. The minimal needed space function does reduce it only slightly. However, if we look at the rent in more detail, we see that for rich singles, starters and pensioners the rent decreases. In contrast for poor young families, the rent increases. Because this goes paired with an increase in utility, this is not necessarily bad.

The reason both utility and rent increase is the fact that poor young families gain money from the space per person tax. We see this when looking at the amount of money they save each month in figure 8.1. Poor young families use a below average amount of space per person. Therefore they gain more money from the taxes than they pay. Because poor young families have more money, they can afford to live in better houses. These houses are more expensive, which is why we see an increase in rent.

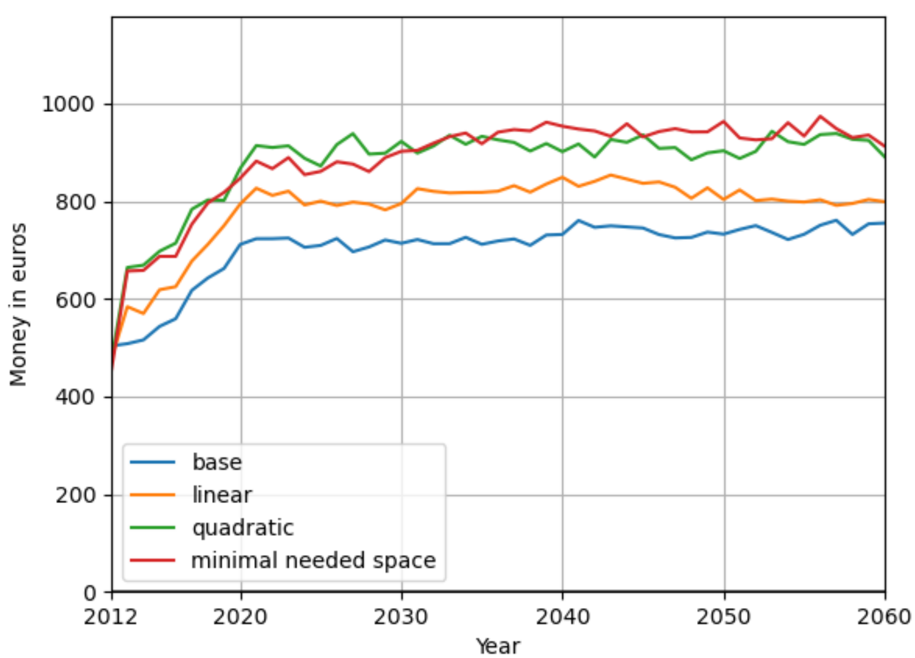

Figure 8.1: Average saved money per month of poor young families for taxation policies.

All taxation functions reduce the *want to move time*. The homelessness and the *want to move count* do not change.

Therefore, the strength of the lock-in effect decreases slightly. For homelessness, we see that the poorer a household is the more likely it is to be homeless. This effect is also present in the *base model*.

Because this policy does not create any new houses, the shortage remains the same as in the *base model*. When looking in more detail into the houses, we see that a lot of houses are listed on the market. We display this in Figure 8.2. We also look at the average size of the listed houses. Figure 8.3 shows this size increases for the quadratic and minimal required space function.

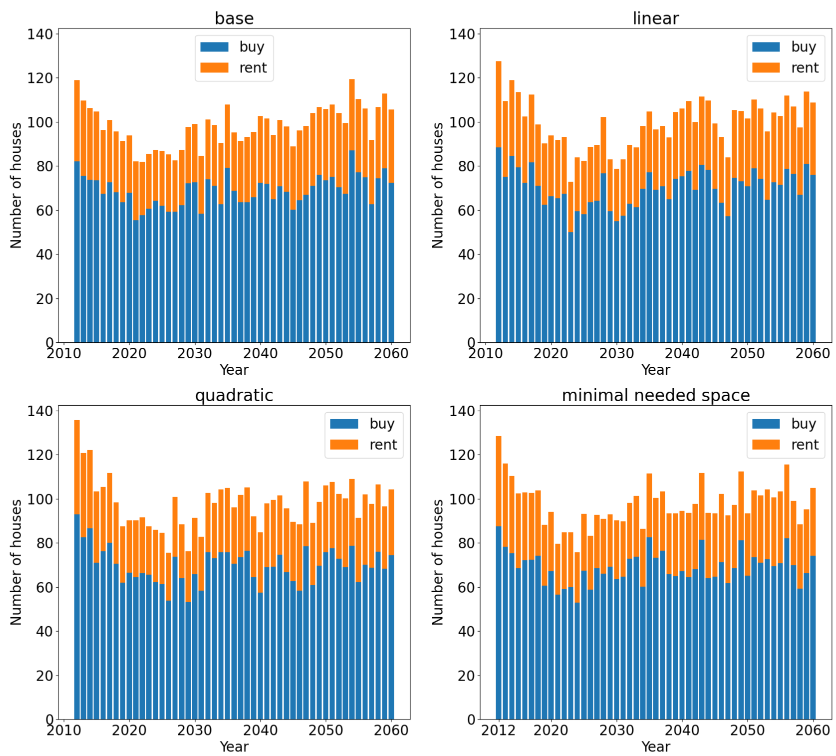

Figure 8.2: Number of listed houses for taxation policies.

Conclusion

With the correct taxation function, this policy reduces the m^2 per person. In this case, the policy does counter some of the effects of the housing crisis.

A careful analysis of the rent price reveals a positive change in this price. This analysis points out the importance of looking at the distribution of rents over the population. Only looking at the average rent leads us to believe this policy did not affect the rents.

In general, it is often useful to look at the distribution of certain metrics of households with respect to their income percentile. Unfortunately, we did not track metrics this way when gathering our data. So we have to deduce such statistics from the data we did measure.

We observe a slightly weakened lock-in effect. However, we also observe no change in the *want to move time* or homelessness. For homelessness, we observe that more poor than rich households are homeless. We see there are enough houses listed to house all the homeless households. Homeless households do not move into these houses because they cannot afford to do so. This already is the case in the *base model*. However, this effect gets worse for this policy. The listed houses are more expensive. This has two causes. First, the space per person tax makes housing more expensive.

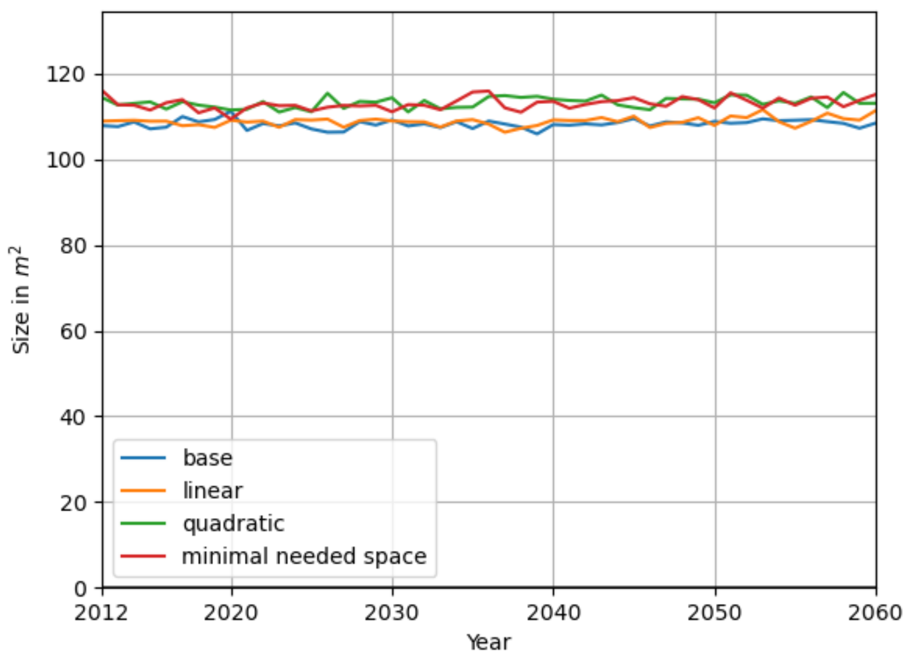

Figure 8.3: Average size of listed houses for taxation policies.

Next, we see that the size of listed houses increases. Given the linear nature of the market price prediction models, bigger houses are more expensive. We conclude there is a mismatch between homeless households and listed houses. Homeless households tend to be poor. Whilst listed houses are more expensive than in the *base model*. To effectively reduce homelessness, we need a policy that targets this mismatch.

As stated above the taxation policy does not counter all effects of the housing crisis. There is an increase in buying price. We have no sound explanation as to why this is the case.

We also observe no change in the housing shortage. This is what we expect, given that the policy does not create extra houses. If we want a policy that decreases the shortage, it is obvious what we should do. We should design a policy that adds houses to the model. Given that we fix the number of constructed houses for all models. We should create these houses from the already available space.

8.2 Limiting space per person

The second policy we explore puts a limit on the amount of space per person a household uses. Households cannot move into a house in which their space per person exceeds this limit. We explore different limits, namely 30, 55, 80, and 105 m^2 per person,

For this policy, it is of course obvious how it will reduce the m^2 per person. As with the previous policy, this policy does also not create new houses. Therefore it can not reduce the housing shortage.

Analysis

Looking at Table 8.3, we see that all limits reduce the space per person. We show these reductions in Figure 8.4. We see that the lower the limit is, the lower the space per person is. This generalizes to most of the metrics in this experiment. The lower the limit is, the stronger the observed effect is.

Table 8.3: Overview of results of limiting space per person policy

	30 m^2 limit	55 m^2 limit	80 m^2 limit	105 m^2 limit
Average m^2 per person	- -	- -	- -	- -
Space per person distribution	?	?	?	=
Buying price	- -	- -	- -	-
Average rent	- -	- -	- -	- -
Number of homeless households	++	++	++	+
Number of households that want to move	++	++	++	-
Average want to move time	++	++	++	- -
Housing shortage	=	=	=	=
Average utility	- -	- -	- -	- -

We see a reduced buying price for all limits. For the 105 m^2 limit, this is only a small reduction. The rent price decreases for all limits.

The lock-in effect shows interesting results. We see that homelessness gets worse for all policies. In particular, for the lower three limits, homelessness increases a lot. We see the effects of the increased homelessness in the *want to move count*. With only the 105 m^2 limit performing slightly better than the *base model*.

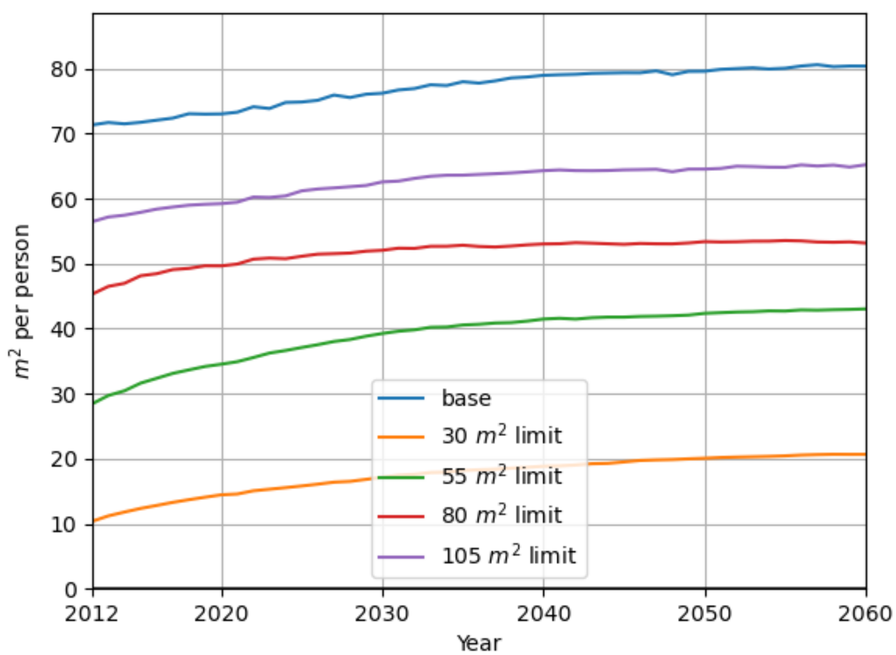

Figure 8.4: Space per person for limit policies.

The *want to move time* follows a similar trend as the other two lock-in effect metrics. It gets worse for the lower three limits. For the 105 m^2 limit the *want to move time* decreases.

As this policy does not create any new houses, the housing shortage does not decrease. For the three lowest limits, we see an increase in the number of listed houses in Figure 8.5. There also is an increase in the average size of the listed houses For all policies, we see a drop in utility. We show this in Figure 8.6.

Conclusion

All limits decrease the space per person. However, not all limits counter the effects of the housing crisis. As expected the housing shortage does not change for all limits. All limits have a positive effect on buying and rental prices.

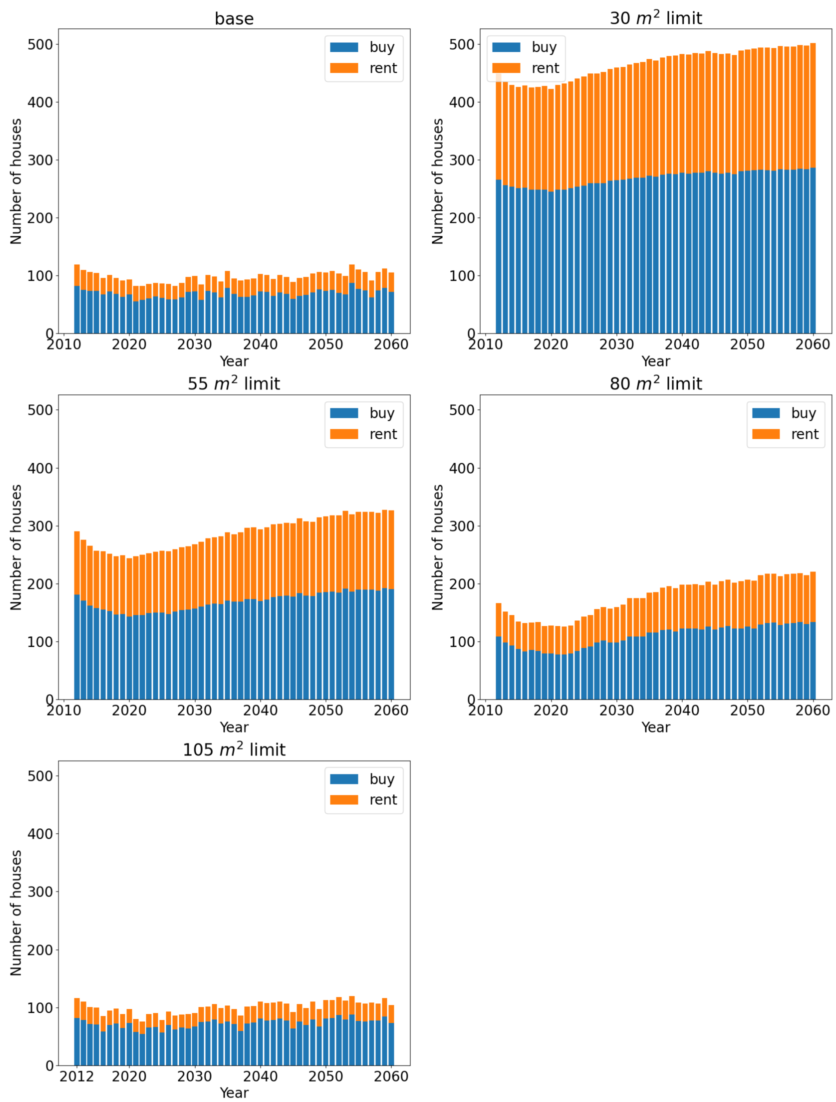

Figure 8.5: Number of listed houses for limit policies.

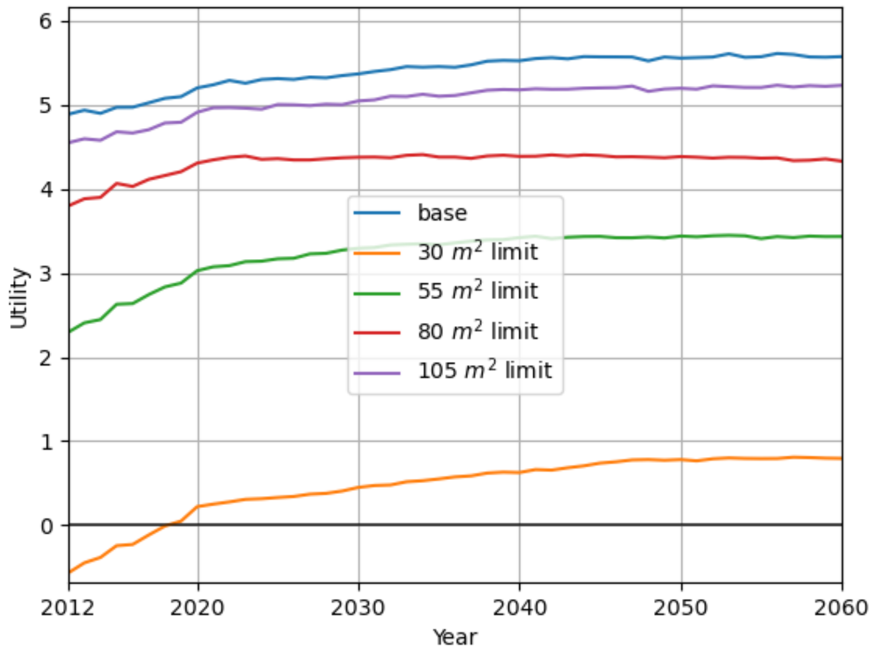

Figure 8.6: Utility for limit policies.

When looking at the lock-in effect, we see that homelessness increases for all limits. Like the previous policy, this is because homeless households cannot move into listed houses. For the previous policy, homeless households could not afford to move into the listed houses. For this policy, the m^2 per person limit prevents households from moving into a house. This leads to households having fewer choices, essentially making the market smaller. A way to fix this is to modify the existing houses, such that more households can move into those houses.

As a result of the reduced choice of houses, it becomes harder to find a house. This leads to a higher *want to move time* for the lowest three limits. Which leads to a stronger lock-in effect for these limits. Only for the 105 m^2 limit, this has a positive effect. Although homelessness increases, we see a decrease in *want to move count* and *time*.

105 m^2 per person is a big limit. Such a limit only targets excessive usage of space. From this experiment, we conclude that targeting excessive usage is effective. However, in this instance, it also comes with a decrease in utility.

This experiment shows that reducing the space per person as much as possible is not necessarily a good idea. It is more important to have houses available that households can move into.

8.3 Sharing rental houses

We now experiment with a house-sharing policy. This policy allows households to share rental houses with one other household. When two households decide to share they both pay 50% of the rent. Both social or private rental houses can be split.

A currently renting household decides if it wants to share. It does this by comparing utilities. It compares their current utility with the utility of sharing their house. For this, a household assumes that they share with a household of the same size. When calculating the sharing utility, the calculation of the space per person changes. It is now equal to the size of the house divided by the sum of the sizes of the two households. This causes a drop in space per person. The amount of saved money increases, because the paid rent goes down. If the utility of the imagined sharing situation is higher than the current situation, the household lists its house for sharing.

We count houses listed or used for sharing double when calculating the housing shortage. We do this because such a house can house two households. When measuring house sizes, we measure a shared house as two houses with half the size of the original house.

Analysis

Table 8.4 shows a general overview of the results of this experiment. There is a reduction in m^2 per person of around 10 m^2 in 2060. We see that the space per person distribution becomes more centered around the mean. Figure 8.7 shows this.

Table 8.4: Overview of results for sharing policy

	sharing
Average m^2 per person	- -
Space per person distribution	?
Buying price	-
Average rent	- -
Number of homeless households	- -
Number of households that want to move	- -
Average want to move time	?
Housing shortage	- -
Average utility	++

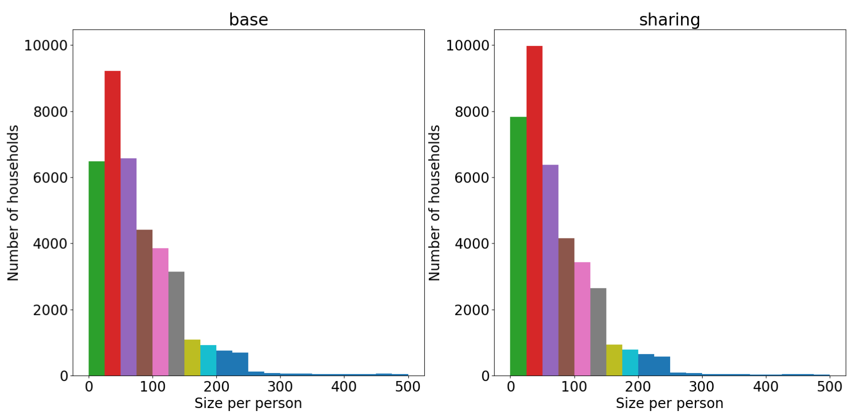

Figure 8.7: Space per person distribution for sharing policy.

Both the buy and rent price decrease. However, if we look at the value of buying transactions, we see an increase in value. We show this in Figure 8.8. As discussed in Section 6.6.1, this indicates that more expensive houses are sold, instead of houses getting more expensive.

Homelessness and *want to move time* decrease drastically. Where in the *base model* around 50 households are homeless. For the sharing policy, this decreases to 10 over time. It is unclear if the *want to move time* improves for the sharing policy. It decreases for the first 30 years but increases for the last 20 years (Figure 8.9).

This policy does reduce the housing shortage. As explained above we count houses available for sharing twice. Figure 8.10 shows how many houses are for sharing. There are so many shared houses that the shortage becomes negative. So there are more places to live for households than there are households.

When we zoom in on the different groups we see that super rich households are not affected by this policy. There is no reduction in space per person for this group. We do see a reduction in the space per person for poor and super poor households. The utility for these groups goes up and fewer of them are homeless. We observe that this policy is affecting more households that already live in a small space. This raises the question if this is wanted behavior.

When looking at the house size distribution in Figure 8.11, we see an increase in the number of small houses. Small houses are houses with a size of less than 100 m^2. We do not observe much of a change in the number of houses bigger than 100 m^2.

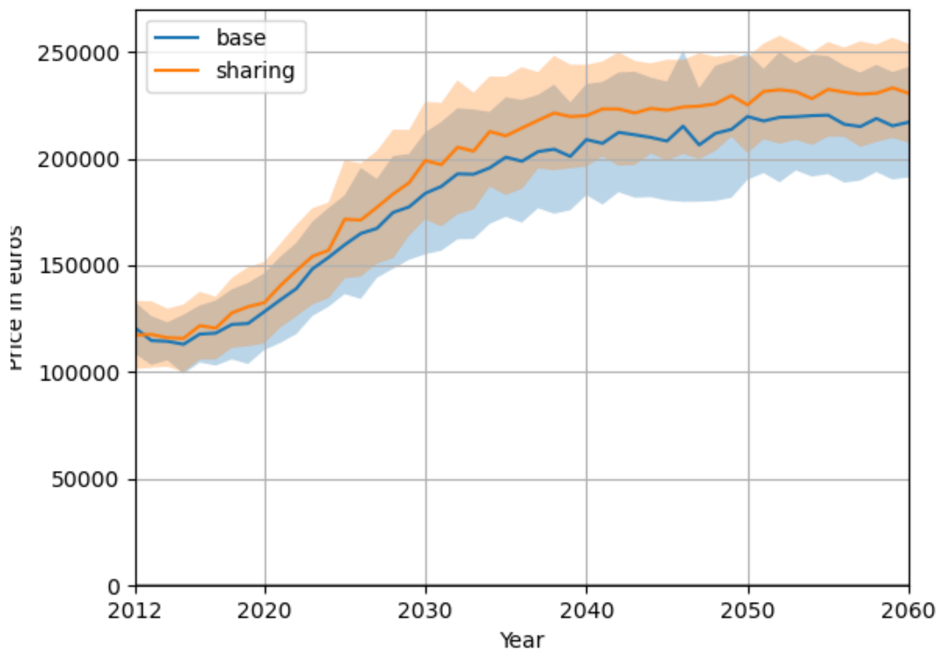

Figure 8.8: Value of buying transactions for sharing policy.

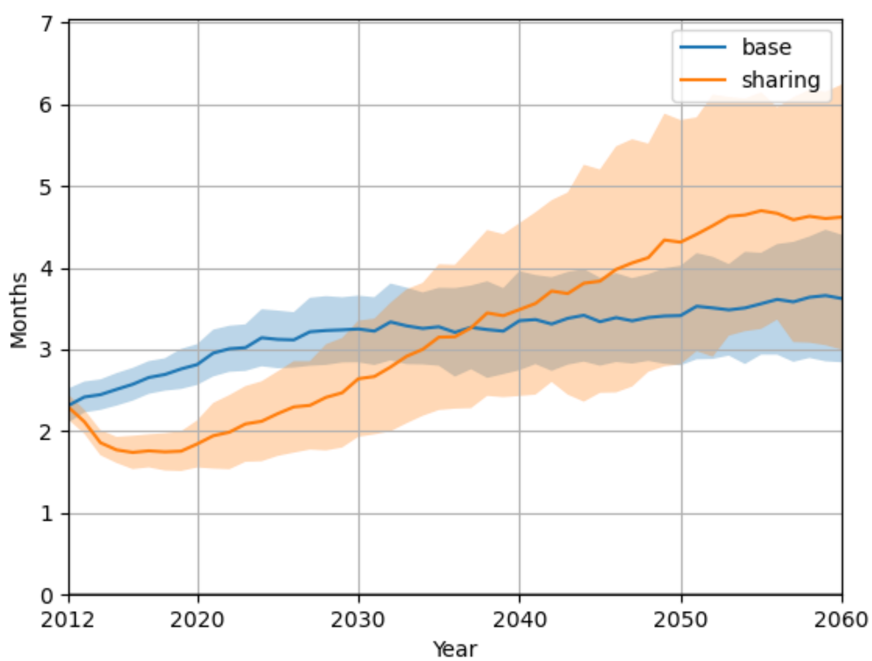

Figure 8.9: Want to move time for sharing policy.

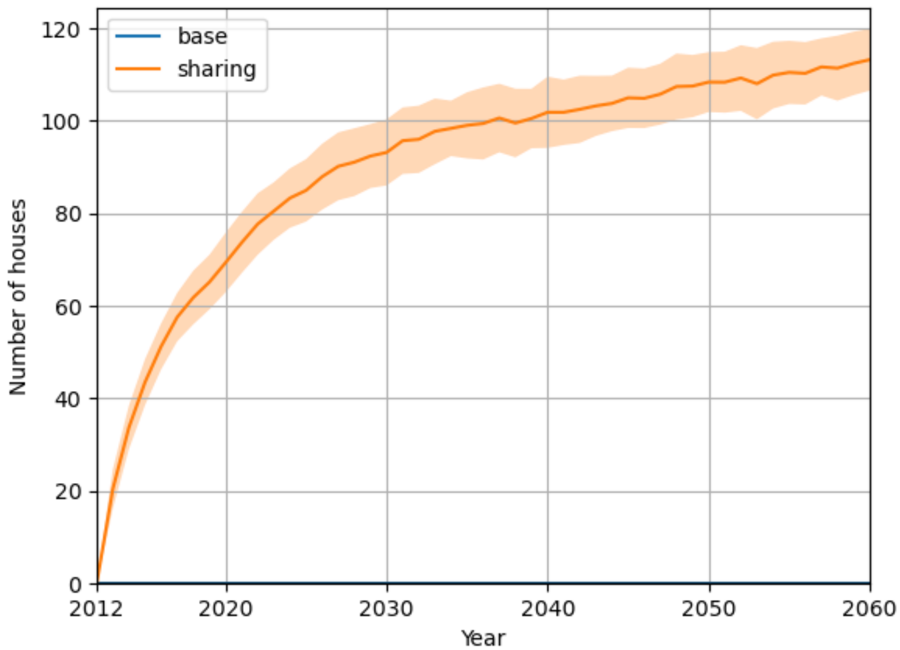

Figure 8.10: Number of shared houses for sharing policy.

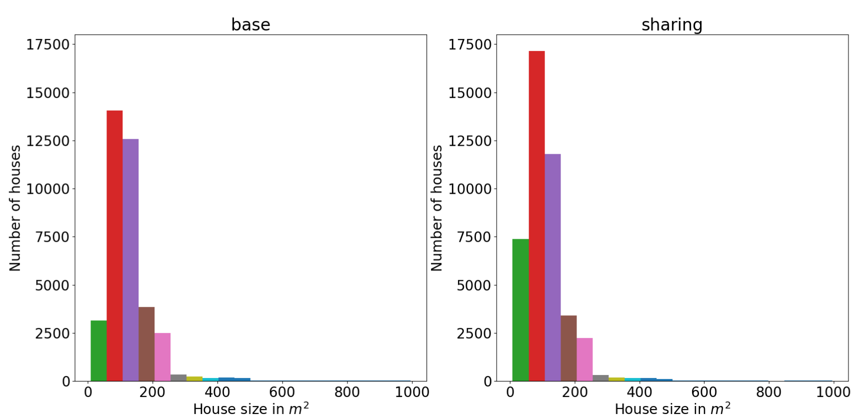

Figure 8.11: House size distribution for sharing policy.

Conclusion

By allowing households to share their house, this policy reduces the space per person. We see that the policy counters most effects of the housing crisis. Only for the lock-in effect, it is the question if it is countered. We see a decrease in homelessness and *want to move count*, but the *want to move time* is harder to analyze.

We first look at homelessness. In the previous two policies, we saw no decrease in homelessness. This was due to a mismatch between households and listed houses. This policy reduces homelessness, by creating lots of small houses. These are houses that poorer households can afford. As we saw before, poorer households are more often homeless. Because the sharing policy creates houses poor households can afford, homelessness reduces. In other words, this policy fixes the mismatch.

We observe that the *want to move time* decreases first and then starts to increase. The amount of shared houses we add is the reason for this. The added shared houses cause the shortage to be negative. This means there are more places to live than households. Therefore, every household that wants to move can find a new place. This is true for homeless households, leading to the observed reduction in homelessness. However, this is not necessarily true for households that want to move and already own a house. They need to find a house that is better than their current house.

We see that the added shared houses are small, meaning they produce a low utility. Households that want to move often do not want to move to such a house. As the number of shared houses grows, it becomes harder to find a non-shared house. This is due to the brochure mechanism. For a brochure, we select ten houses listed on the market at random. When more and more houses on the market are shared houses, the chance of showing a non-shared house decreases. More often households will receive a brochure with only shared houses. Therefore, it takes longer before such households are able to move. Resulting in the increase in *want to move time* as the number of shared houses increases.

The sharing policy seems to affect poorer households more than richer households. This is probably not desired. We want to spread the burden of fixing the crisis over the entire population.

We believe there are three reasons why rich households are not affected as much. First, they are buying more. As sharing only targets the rental market, buying households are not affected. Second, renting rich households do not need to share. They can easily afford a place on their own. Lastly, as we saw most of the houses added through sharing are small. Rich households can afford to live in bigger more expensive houses. So when given the choice they will pick such a house.

Compared to the previous policies, this policy does reduce the housing shortage. However, these places are only added to the rental market. The buying market is not affected. We also see that the added places mostly have the same size. In other words, there is not a huge variety in the sizes of the added houses. To tackle this problem we think we need to create a higher variety of houses. And affect both rental and buying houses.

8.4 Splitting buy houses

With this policy, we look at splitting houses. House owners can split their house into two parts. These parts have the same quality as the original house. The sum of their sizes is equal to the size of the original house.

Households that split keep living in one part and rent out the other part. They earn the rent that a household pays when living in the rental part. The process for deciding to split a house is similar to that for deciding to share. It is also based on comparing utilities. A household compares the utility of its current situation with a splitting situation. For the splitting situation, it assumes it rents out the rental part at its current market price.

There is a cost associated with splitting. This cost covers the remodeling of the house to make it suited for two households to live in. This is a one-time cost of €50,000 euro paid from a household's wealth. We cannot include one-time costs in utility calculations. Therefore, we assume a household wants to earn this money back within a year. So we subtract $€50,000/12 = 4,166\frac{2}{3}$ from the saved money in our utility calculations. This is a big amount. Therefore, it looks like splitting is only an option for rich households.

Similar to house sharing this policy creates extra houses from the existing space. But again the created houses are only added to the rental market. However, the policy also affects buying houses. Because splitting changes a buy house, this house is smaller after the split.

Analysis

Table 8.5 displays the results for the splitting experiment compared to the *base model*. There is only a slight reduction in space per person. It drops around 1 to 2 m^2 in 50 years. We observe no change in the distribution.

Table 8.5: Overview of results for splitting policy

	splitting
Average m^2 per person	-
Space per person distribution	=
Buying price	=
Average rent	-
Number of homeless households	- -
Number of households that want to move	- -
Average want to move time	-
Housing shortage	-
Average utility	++

The prices are not affected too much. The buying price does not change. The rent price shows a small decrease. The lock-in effect changes for the better. We observe a decrease in both the *want to move count* and homelessness. These are nearly halved. The *want to move time* reduces slightly.

Figure 8.12 shows the number of split houses created. These created houses cause the shortage to decrease. After 20 years there are more houses than households, causing a negative housing shortage. When we look at the sizes of the created houses in Figure 8.13. We see an increase in small houses with a size of less than $100m^2$.

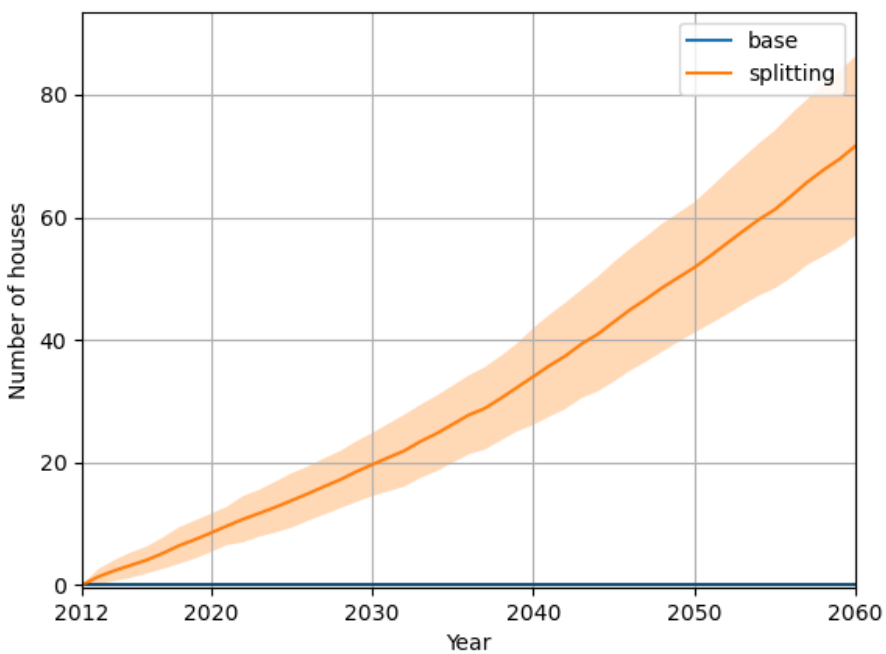

Figure 8.12: Number of split houses for splitting policy.

When looking at the subgroups, we see that they are affected in line with the general effects. However, it is hard to observe this given that the changes are small. We observe that homelessness does not decrease for rich and super rich households. It does decrease for middle, poor, and super poor households.

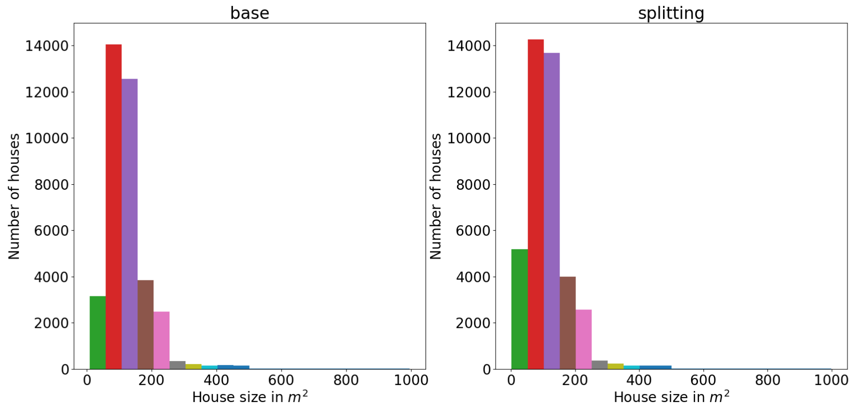

Figure 8.13: House size distribution for splitting policy.

Conclusion

This policy does reduce the space per person. The policy counters most effects of the housing crisis. Only the buying price does not improve. However, although there is a negative housing shortage we still observe homelessness. This is because the policy does not reduce homelessness for richer households.

We believe the reason for this is that the houses we add are rental houses. A lot of those added houses end up as social rental houses. It is more difficult for rich households to move into a social house. Only 15% of all social houses are available for rich households. Therefore the new houses are not suited for rich households.

This policy does create a lot of houses. However, as with the sharing policy, most of those are small. Again the variety in the size of created houses is not huge. We suspect this policy suffers from the same problems as the sharing policy. Such as the problem with the brochure. However as this policy adds fewer houses than the sharing policy, this *brochure problem* is less prominent.

We see households keep deciding to split their house. Even when there are more than enough split houses. This is because when deciding to split households always assume they can rent out the new part. This is not fair. Households should take into account how likely it is that they rent out the new part.

8.5 Rewarding splitting

This policy extends the splitting policy by rewarding households that split their house. This reward lowers the one-time cost households pay for splitting their house. This makes households with low wealth more likely to split. This also affects the decision to split. Because the reward increases, the utility of the splitting scenario increases. It does this by lowering the amount needed to split, resulting in more saved money. We experiment with rewards of €10,000, €20,000, and €30,000 euros.

Compared to normal splitting we expect a higher variety in the type of houses created. This should result in more people being able to find a house. Therefore reducing homelessness and lowering prices. Also *the want to move time* should go down as there are more houses to choose from.

Analysis

We compare the results of the rewarding splitting experiment with the results of the splitting experiment. We show those results in Table 8.6. We see all rewards reduce the space per person. The higher the reward is the higher the reduction in space per person is. In general, we observe that the higher the reward is, the stronger the observed effect is.

Table 8.6: Overview of results for rewarding splitting policy

	€10,000 reward	€20,000 reward	€30,000 reward
Average m^2 per person	- -	- -	- -
Space per person distribution	=	?	?
Buying price	=	=	-
Average rent	=	-	-
Number of homeless households	- -	- -	- -
Number of households that want to move	- -	- -	- -
Average want to move time	-	?	?
Housing shortage	- -	- -	- -
Average utility	=	++	++

Only the €30,000 bonus model lowers the buying price. If we look at the rent, we see that the two highest bonuses reduce it slightly.

For the lock-in effect, we see that the homelessness and *want to move time* decrease. Again, the higher the reward the bigger the decrease. For the *want to move time* we see a small reduction for the €10,000 bonus. For the €20,000 and €30,000 bonus, we see a similar effect as with the sharing policy. With a decrease in the first 30 years and an increase in the last 20 years.

The higher the bonus the more houses are created (Figure 8.14). This results in a reduced shortage. Figure 8.15 compares the distributions of house sizes for all models. We see an increase in the number of houses with a surface of less than $50m^2$. For only the buy houses, we see a decrease in the number of 100 to 150 m^2 houses and an increase in the number of 0 to $50m^2$ houses. This indicates that most of the houses we split are 100 to $150m^2$ houses. We show this in Figure 8.16.

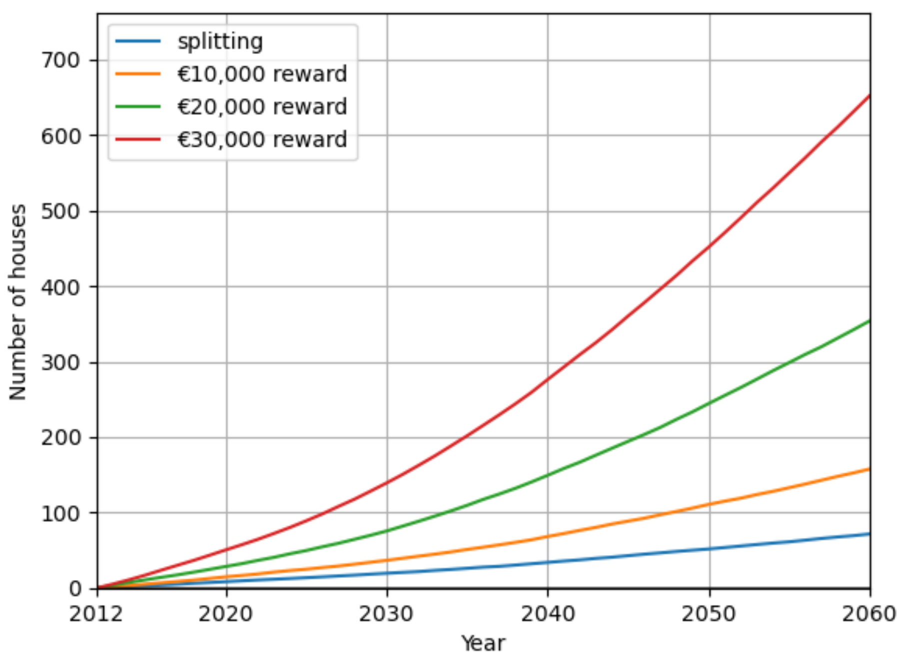

Figure 8.14: Number of split houses for rewarding splitting policy.

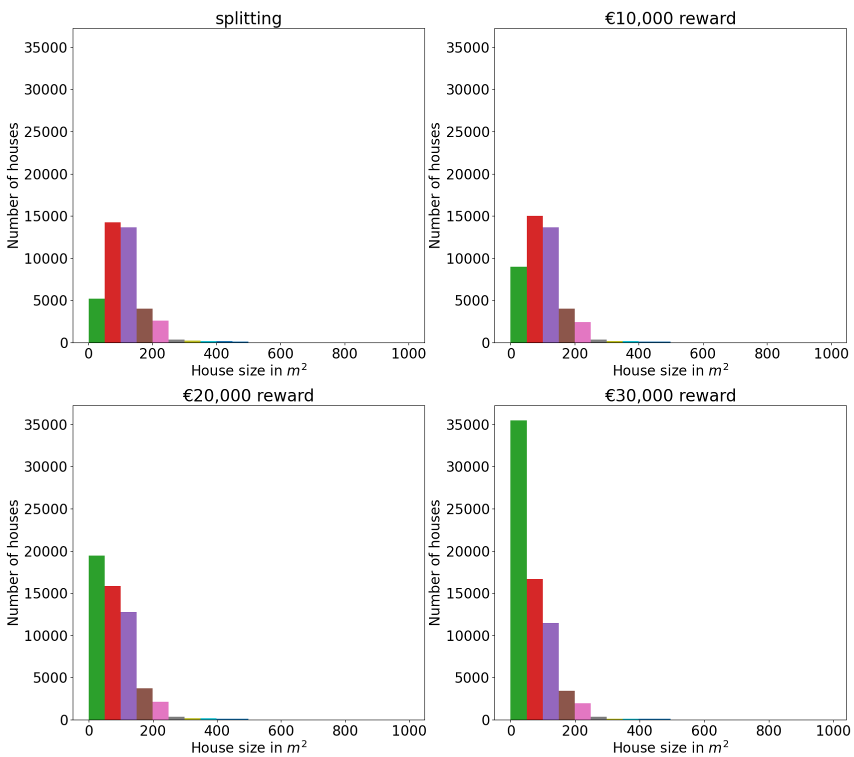

Figure 8.15: House size distribution for rewarding splitting policy.

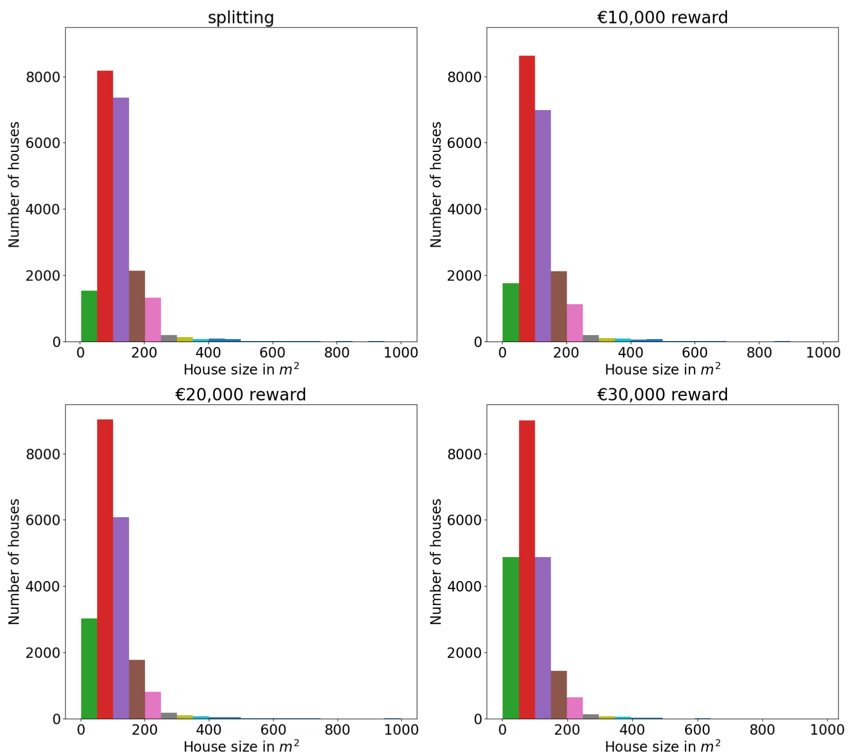

Figure 8.16: House size distribution of buying houses for rewarding splitting policy.

Conclusion

We see that the €30,000 reward is effective in countering the effects of the housing crisis. Only for the *want to move time* we first see a decrease and then an increase. As with the sharing policy, we suspect this is due to how the brochure works.

Again this policy produces much more houses than needed. The variety in created houses is not better than before. Only small houses are added. Also, these houses are created from relatively small houses. The big houses do not change. This is something we want to address. We already have enough small houses. There is no need to create even more.

8.6 Split big listed houses

For this experiment, we allow the government to split houses listed on the market. The government splits houses in which a household of average size would have too much space. We experiment with different limits for what is too much space. These limits are 55, 80, and 105 m^2 per person.

As before, splitting a house costs money. Also, the government first has to buy houses listed on the buying market before it splits them. To cover the costs of splitting and buying the government raises extra taxes.

Houses are split into two equal-sized halves and have the same quality as the original house. If the original house was a buy house, we list both halves on the buying market. If the original house was a rental house, we list the halves on the rental market. This means this policy creates houses in both markets.

Analysis

We show the results for this policy in Table 8.7. We see that all policies reduce the space per person. The lower the space per person limit is the higher the reduction is.

Table 8.7: Overview of results for splitting listed houses policy

	55 m^2 limit	80 m^2 limit	105 m^2 limit
Average m^2 per person	- -	- -	- -
Space per person distribution	?	=	=
Buying price	- -	- -	-
Average rent	++	-	-
Number of homeless households	- -	- -	- -
Number of households that want to move	- -	- -	- -
Average want to move time	++	-	- -
Housing shortage	- -	- -	- -
Average utility	++	++	++

The buying price decreases for all limits. The lower the limit is the bigger the decrease is. We see that the 80 and 150 m^2 limits decrease the rent. The 55 m^2 limit increases the rent.

When looking at the lock-in effect, we observe that all limits decrease homelessness. The lower the limit, the bigger the reduction. For the 80 and 105 m^2 limits, we see this also in the *want to move count*. However, for the 55 m^2 limit we see a decoupling between homelessness and *want to move count*. We plot the *want the move count* in Figure 8.17. The *want to move time* improves for the two higher limits. For the 55m^2 limit, the *want to move time* increases.

The lower the limit, the more houses are added to the model. We plot how many houses are added in Figure 8.18. This directly translates to the housing shortage. All limits reduce the shortage to below zero.

When looking at the house size distribution in 8.19, we see that the policy creates a lot of similar houses. We suspect this is because we split each house into two halves of the same size. It might be better to split the houses into unequal parts.

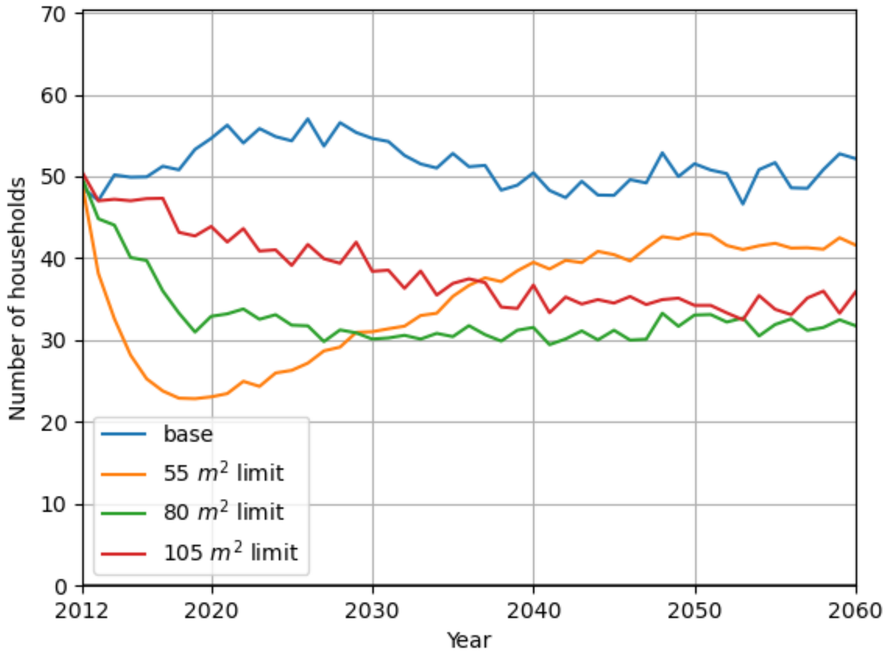

Figure 8.17: Want to move count for splitting listed houses policy.

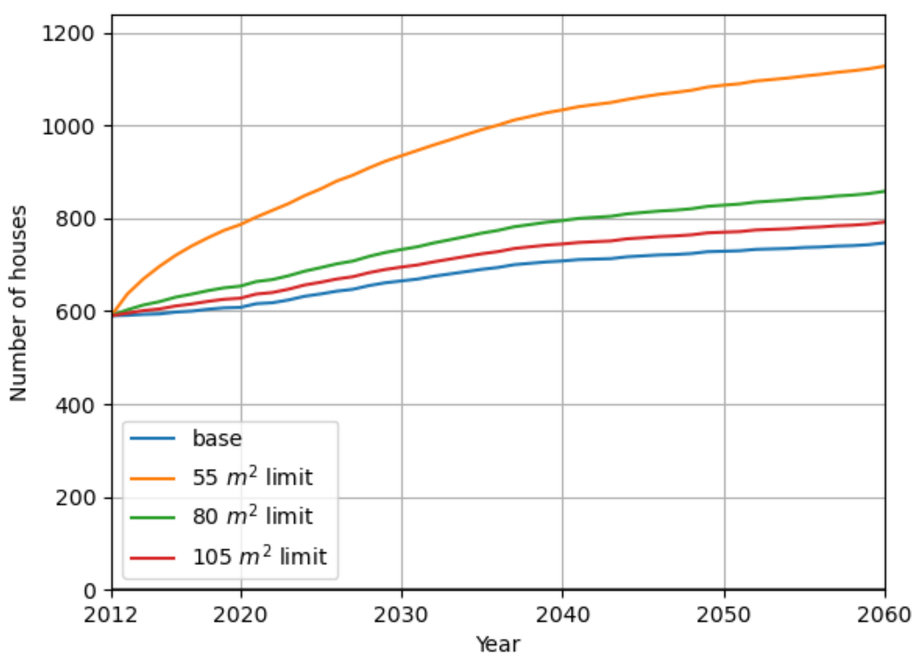

Figure 8.18: Number of houses for splitting listed houses policy.

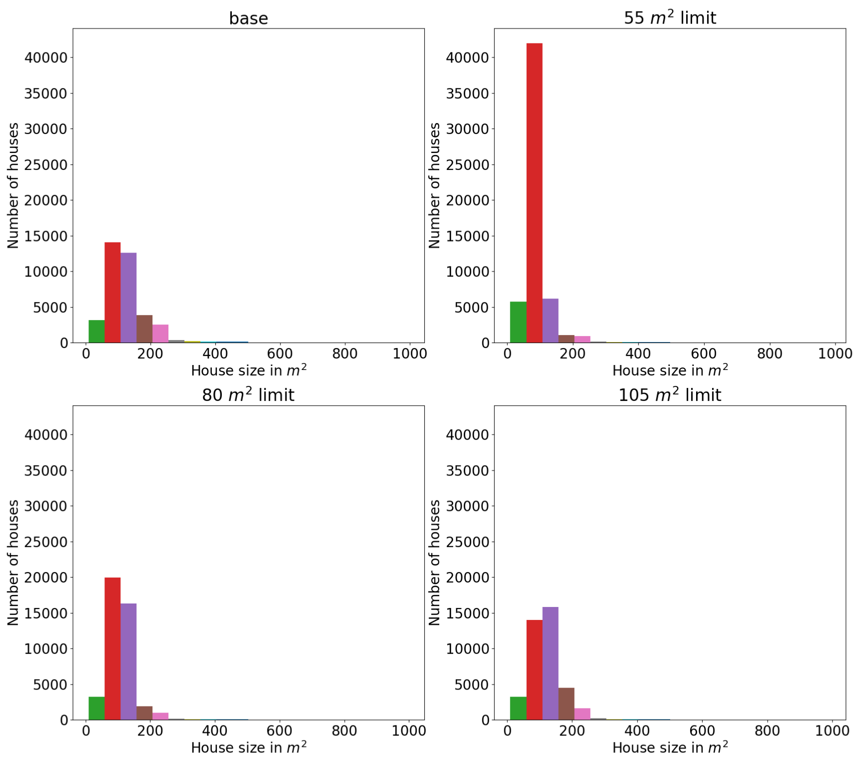

Figure 8.19: House size distribution for splitting listed houses policy.

Conclusion

All limits reduce the space per person. Only the 80 and 105 m^2 limit counter all the effects of the housing crisis. It looks like the 55 m^2 limit creates too many houses. We see that it starts to suffer from the same problems as the splitting and sharing policies. As discussed, this is a problem with the brochure mechanism. When we look at the *want to move time* for the 80m^2 limit, we see an indication that the *brochure problem* occurs there as well.

For all limits, there are more than enough houses created to house all households. Yet there still is too much homelessness for all three limits. So there still is a mismatch between homeless households and listed houses.

We believe the mismatch effect is more prominent, because of the brochure mechanism. In essence, the brochure mechanism limits the number of houses a household has to review before making a decision. This is good when there are a lot of houses that would suit the household. If there are only a few houses that match with the household, this is not as good. It reduces the chance that a household will find the houses it can move into.

To fix this we either remove the brochure mechanism or improve it. Removing the brochure mechanism means a household has to review all listed houses. This might not be computationally feasible. To improve the brochure mechanism, we only include houses in a brochure that have a high chance of being suitable for the household we give the brochure to.

Another problem we observe, not only for this policy, is that we create too many houses. We keep adding houses even when the shortage is negative. This happens because when creating a new house we do not take into account how many houses there are. Also, we only allow the splitting houses. We do not allow the merging of houses. So the number of houses can only increase and never decrease.

Chapter 9

Conclusion

Policies that reduce the m^2 per person do not necessarily counter all the effects of the housing crisis. It appears to be more important to create extra houses. Based on the sharing and splitting policies, we suspect that we can create the extra houses without building more new-built houses. In other words, we can create these houses by modifying existing houses to house more people.

The results of the sharing and splitting policies indicate that it is important to have variety in the created houses. We believe we should create houses that target all layers of the population. If we create houses that target only a specific subgroup of the population, we do not solve the problems of other subgroups.

The taxation and limiting policies only reduce the space per person. These policies do not create extra houses. Such policies do not seem to counter all the effects of the housing crisis. This leads us to believe that reducing the space per person is not the most important factor we need to address to fix the housing crisis. Creating extra new houses appears to be a bigger factor in fixing the housing crisis. The sharing and splitting policies do this by modifying existing houses. Such policies seem to have a reduction in the space per person as a side effect.

The sharing and splitting policies lead us to believe that is possible to fix the housing crisis by creating houses from existing houses. We did not research this hypothesis directly. Therefore, this is an interesting hypothesis for a follow-up study.

Such a study should consider the effect that sharing or splitting a house has on a household. Our model does not consider the human impact of sharing or splitting. People may find it intrusive to share or split their house. They might value their privacy and do not want to live so near to another household. We also assume that every household is equally willing to split or share. This is probably not realistic. Our model does not consider these soft criteria when households decide if they want to share or split.

Bibliography

Arentze, T.A., Kowald, M., and Axhausen, K.W. 2013. An agent-based random-utility-maximization model to generate social networks with transitivity in geographic space. *Social Networks 35*, 3, 451–459.

Axtell, R., Farmer, D., Geanakoplos, J., et al. 2014. An agent-based model of the housing market bubble in metropolitan washington, d.c.

Baptista, R., Farmer, J.D., Hinterschweiger, M., Low, K., Tang, D., and Uluc, A. 2016. *Macroprudential policy in an agent-based model of the UK housing market.* Bank of England.

Belastingdienst. 2021. Berekening huurtoeslag 2021.

Bonabeau, E. 2002. Agent-based modeling: Methods and techniques for simulating human systems. *Proceedings of the National Academy of Sciences 99*, suppl 3, 7280–7287.

Centraal Bureau voor de Statistiek. 2018. Honderd jaar alleenstaanden. https://www.cbs.nl/nl-nl/achtergrond/2018/26/honderd-jaar-alleenstaanden.

Centraal Bureau voor de Statistiek. 2019. Bouwen en wonen. https://longreads.cbs.nl/trends19/economie/cijfers/bouwen-en-wonen/.

Centraal Bureau voor de Statistiek. 2021b. Bevolking, huishoudens en bevolkingsontwikkeling; vanaf 1899. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/37556/table?dl=53FFD.

Centraal Bureau voor de Statistiek. 2021d. Particuliere huishoudens naar samenstelling en grootte, 1 januari. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/37975/table?ts=1645603409617.

Centraal Bureau voor de Statistiek. 2021g. Inkomen van huishoudens; inkomensklassen, huishoudenskenmerken. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/83932NED/table?ts=1645603609140.

Centraal Bureau voor de Statistiek. 2021f. Vermogen van huishoudens; huishoudenskenmerken, vermogensklassen. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/83835NED/table?ts=1645603771556.

Centraal Bureau voor de Statistiek. 2021e. Prognose huishoudens naar type; leeftijd, burgerlijke staat, 2022-2070. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/85084NED/table?ts=1645604981407.

Centraal Bureau voor de Statistiek. 2021a. Voorraad woningen; eigendom, type verhuurder, bewoning, regio. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/82900NED/table?ts=1645605239934.

Centraal Bureau voor de Statistiek. 2021c. Voorraad woningen; woningtype, oppervlakteklasse, regio. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/83704NED/table?ts=1645605829383.

Centraal Bureau voor de Statistiek. 2022b. Voorraad woningen; standen en mutaties vanaf 1921. https://opendata.cbs.nl/#/CBS/nl/dataset/82235NED/table?dl=5B52.

Centraal Bureau voor de Statistiek. 2022c. Bestaande koopwoningen; verkoopprijzen prijsindex 2015=100. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/83906NED/table?dl=610D0&ts=1665663235125.

Centraal Bureau voor de Statistiek. 2022d. Consumentenprijzen; huurverhoging woningen vanaf 1959. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/70675ned/table?ts=1665663810978.

Centraal Bureau voor de Statistiek. 2022e. Consumentenprijzen; prijsindex 1900 = 100. https://opendata.cbs.nl/#/CBS/nl/dataset/71905ned/table.

Centraal Bureau voor de Statistiek. 2022a. Begrippen. https://www.cbs.nl/nl-nl/onze-diensten/methoden/begrippen/.

De Domenico, M., Brockmann, D., Camargo, C., et al. 2019. Complexity explained.

DeHypotheekSite. 2019. Hoe wordt je maximale hypotheek berekend? https://dehypotheeksite.nl/hypotheekinformatie/maximale-hypotheek/hoe-wordt-je-maximale-hypotheek-berekend.php.

Destatis. 2020. Building activity. https://www.destatis.de/EN/Themes/Society-Environment/Housing/Tables/dwellings-germany.html.

Diappi, L., ed. 2013. *Emergent phenomena in housing markets: Gentrification, housing search, polarization.* Physica, Heidelberg.

Dutch Ministry of the Interior and Kingdom Relations. 2019. Ruimte voor wonen: Resultaten van het

WoonOnderzoek nederland 2018.

Dutch Ministry of the Interior and Kingdom Relations. 2021b. Staat van de woningmarkt.

Dutch Ministry of the Interior and Kingdom Relations. 2021a. Het statistisch woningtekort nader uitgelegd. https://www.woningmarktbeleid.nl/onderwerpen/berekening-woningbouwopgave.

Dutch Ministry of the Interior and Kingdom Relations. 2022. Wonen langs de meetlat: Resultaten van het WoonOnderzoek nederland 2021.

Evans, B.P., Glavatskiy, K., Harré, M.S., and Prokopenko, M. 2021. The impact of social influence in australian real estate: Market forecasting with a spatial agent-based model. *Journal of Economic Interaction and Coordination*.

Ge, J. 2014. Who creates housing bubbles? An agent-based study. *Multi-agent-based simulation XIV*, Springer Berlin Heidelberg, 143–150.

Geanakoplos, J., Axtell, R., Farmer, J.D., et al. 2012. Getting at systemic risk via an agent-based model of the housing market. *American Economic Review 102*, 3, 53–58.

Gilbert, N. 2008. *Agent-based models*. SAGE Publications, Inc.

Grimm, V., Berger, U., Bastiansen, F., et al. 2006. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling 198*, 1, 115–126.

Howes, A., Lewis, R.L., and Singh, S. 2014. Utility maximization and bounds on human information processing. *Topics in Cognitive Science 6*, 2, 198–203.

Insee. 2019. Reference person of the household - main processing (population census). https://www.insee.fr/en/metadonnees/definition/c1944.

Kouwenberg, R.C.J., Roy AND Zwinkels. 2015. Endogenous price bubbles in a multi-agent system of the housing market. *PLOS ONE 10*, 6, 1–10.

Kraniotis, L. and Jong, W. de. 2021. Sociale huurwoning? In zeker een kwart van de gemeenten wacht je meer dan 7 jaar. https://nos.nl/op3/artikel/2377995-sociale-huurwoning-in-zeker-een-kwart-van-de-gemeenten-wacht-je-meer-dan-7-jaar.

Langenberg, H. and Jonkers, W. 2022. Achtergrond bij de huizenprijsstijgingen vanaf 2013. https://www.cbs.nl/nl-nl/longread/de-nederlandse-economie/2022/achtergrond-bij-de-huizenprijsstijgingen-vanaf-2013?onepage=true.

Lans, J. van der and Pflug, M. 2019. *Canon volkshuisvesting*.

Lee, R.K.-W. and Kam, T.S. 2014. Time-series data mining in transportation: A case study on singapore public train commuter travel patterns. *International Journal of Engineering and Technology 6*, 431–438.

Macal, C.M. and North, M.J. 2010. Tutorial on agent-based modelling and simulation. *Journal of Simulation 4*, 3, 151–162.

Mann, H.B. and Whitney, D.R. 1947. On a Test of Whether one of Two Random Variables is Stochastically Larger than the Other. *The Annals of Mathematical Statistics 18*, 1, 50–60.

Newman, M.E.J. 2011. Resource letter CS–1: Complex systems. *American Journal of Physics 79*, 8, 800–810.

North, M. 2018. Hammer or tongs: How best to build agent-based models? In: 3–11.

Petty, M.D. 2018. Modeling and validation challenges for complex systems. *Engineering Emergence*.

Regiocontainer. 2019. Onderzoeksanalyse: Wat gebeurt er met de woon- en perceeloppervlakte van woningen? https://www.regiocontainer.nl/woon-en-perceeloppervlakte.

Rijksoverheid. 2022. Wat is het verschil tussen een sociale huurwoning en een huurwoning in de vrije sector? https://www.rijksoverheid.nl/onderwerpen/huurwoning-zoeken/vraag-en-antwoord/wat-is-het-verschil-tussen-een-sociale-huurwoning-en-een-huurwoning-in-de-vrije-sector.

Sayama, H. 2015. *Introduction to the modeling and analysis of complex systems*.

Simon, H.A. 1962. The architecture of complexity. *Proceedings of the American Philo- sophical Society 106*, 6, 467–482.

Sociaal en Cultureel Planbureau. 1998. *Sociaal en cultureel rapport 1998*.

Susmel, R. 2014. Discrete choice models. https://www.bauer.uh.edu/rsusmel/phd/ec1-18.pdf.

Ustvedt, S. 2016. An agent-based model of a metropolitan housing market - linking micro-level behavior to macro-level analysis.

Wilensky, U. and Rand, W. 2015. *An introduction to agent-based modeling: Modeling natural, social, and engineered complex systems with NetLogo*. The MIT Press.

Willy, C., Neugebauer, E.A.M., and Gerngross, H. 2003. The concept of nonlinearity in complex systems. *European Journal of Trauma 29*, 1, 11–22.

Yun, T.-S. and Moon, I.-C. 2020. Housing market agent-based simulation with loan-to-value and debt-to-income. *Journal of Artificial Societies and Social Simulation 23*, 4, 5.

Zhuge, C. and Shao, C. 2018. Agent-based modelling of purchasing, renting and investing behaviour in dynamic housing markets. *J. Comput. Sci. 27*, 130–146.

Appendix A

Submodels

Initialization

Create households

We create households based on age-size slots. Each slot has an *age range*, a household size, and a count. The *age range* consists of a minimum and a maximum age. The count states how many households are part of this slot in the real world. To get the correct number of households for the model we need to divide this number by the *scale factor*.

Using the slots we get the age and size of a household. We then pick an income percentile for the household. Using the income percentile and the age, we give the household an income and a starting wealth. The procedure is the same for both income and wealth. Only the data we use is different. We describe how we pick the income and wealth in more detail below.

Lastly, we decide if we base the household's waitlist time on its age, or if it is zero.

```
households = []
for age_range, size, count in slots:
    scaled_count = round(count / SCALE_FACTOR)

    for i in 0..scaled_count:
        age = randint(age_range.min, age_range.max)
        income_percentile = random(0, 1)

        income = get_money(age, income_percentile, income_data)
        wealth = get_money(age, income_percentile, wealth_data)

        waitlist_time = age if random() < 0.75 else 0

        household = Household(age, size, income_percentile, income,
                              wealth, waitlist_time, Homeless)

        households.append(household)
```

Picking the income and wealth follows the same procedure. The amount it returns depends on the age and income percentile of a household. It also depends on the dataset we provide it with. This is a dataset either for income or for wealth. These datasets have data for different age ranges. To find a household's money, we first need to pick the data for the age range the household falls in. For every age range, we have data on which percentage of households fall into a certain income class. For each income class, the data provides us with a mean amount of money households in that class earn. We find the correct class for a household and pick an amount of money based on the mean for that class.

```
def get_money(age, income_percentile, data):
    income_classes = pick_income_classes_for_age(age, data)
```

```
    total_percentage = 0
    for percentage, mean in income_classes:
        if total_percentage <= total_percentage + percentage:
            return pick_money(mean)

        total_percentage = total_percentage + percentage
```

Create houses

Like the age-size slots for households, we have size slots for houses. These slots have a *size range* and a count. The *size range* gives the minimum and maximum size for houses in that slot. The count says how many houses, we need to be in the slot. Again we need to scale this number by the *scale factor*. For house quality, we have no dataset. Therefore, we pick the quality from a uniform distribution between 1 and 10. We also have the *percentage for rent* variable. This variable indicates which percentage of houses is for rent. We list each house on its respective market.

```
houses = []
for size_range, count in slots:
    scaled_count = round(count / SCALE_FACTOR)

    for i in 0..scaled_count:
        size = randint(size_range.min, size_range.max)
        quality = random(1, 10)

        if random() < PERCENTAGE_FOR_RENT:
            house = RentalHouse(size, quality)
            rental_market.list(house)
        else:
            house = BuyingHouse(size, quality)
            buying_market.list(house)

        houses.append(house)
```

Bank

Calculate maximal bid for household

When creating a brochure for the buying market, we show a household only houses that it can afford. For this, we need to know the maximum value the household can bid on a house. For the maximum bid, we first have to calculate the maximum mortgage for the household. We base our calculation of the maximum mortgage on [DeHypotheekSite 2019]. We do not describe this process here.

Because of interest rates, the maximum bid is a percentage of the maximum mortgage. For our models, the interest rate is always 5%. Therefore, the maximum bid is 95% of the maximum mortgage.

Households can use their wealth to raise the maximum bid. Households with wealth greater than zero, use 30% of their wealth to increase the bid.

```
def maximal_bid(max_mortgage, wealth):
    max_house_value = (1 - INTEREST_RATE) * max_mortgage

    if wealth >= 0:
        return max_house_value + 0.3 * wealth

    return max_house_value
```

House list value to monthly mortgage payment

We use the monthly amount a household pays for housing in our utility function. For rental houses this is easy, we use the monthly rent. For buying houses we need to convert the buying value to a monthly mortgage payment.

To do this we first determine for which amount the household needs a mortgage. This depends on whether the household pays part of the buying value from its wealth. As discussed, households with a positive wealth spend 30% of it when buying.

This leaves an amount of money for which the household needs a mortgage. We multiply this amount by the interest rate to get the mortgage value. To get to the monthly payment, we divide this amount by the number of months the household has to repay the mortgage.

```
def house_value_to_monthly_payment(house_value, wealth):
    if household_wealth > 0:
        house_value = house_value - 0.3 * wealth

    if house_value <= 0:
        return 0

    needed_mortage = house_value * (1 + INTEREST_RATE)

    return needed_mortage / (MORTGAGE_DURATION_IN_YEARS * 12)
```

Government

Set liberalization treshold

We set the liberalization threshold such that between 50% and 75% of all rental houses are social rental houses. If the percentage of social rental houses is lower than 50%, we raise the threshold by 5%. This way houses with a higher rent also become social rental houses. Therefore, more rental houses will be social rental houses. If the percentage is higher than 75%, we lower the threshold by 5%. Now the reverse effect occurs. Less rental houses become social rental houses.

```
def set_liberalization_threshold(number_of_rental_houses, number_of_social_houses):
    percentage_social = number_of_social_houses / number_of_rental_houses

    if percentage_social < 0.5:
        LIBERALIZATION_THRESHOLD *= 1.05
    elif percentage_social > 0.75:
        LIBERALIZATION_THRESHOLD *= 0.95
```

Set tax rate

To set the tax rate, the government first calculates how much rent allowance it pays in total each month. It also sums all incomes of the households. We only include incomes larger than zero in this summation.

To set the tax rate, the government calculates which percentage the rent allowance is of the total income. This percentage is the new tax rate.

```
def set_tax_rate(households):
    total_rent_allowance = 0
    total_income = 0

    for household in households:
        if household is renting:
            total_rent_allowance += calculate_rent_allowance(household, rent_paid_by_houseshold)
        if household.income > 0:
            total_income += household.income
```

```
    tax_rate = total_rent_allowance / total_income
```

Calculate rent allowance for household

We use the same procedure as the Dutch government for our calculation of the rent allowance [Belastingdienst 2021]. We generalize this procedure with respect to the liberalization threshold. We do this because the liberalization threshold changes over time in our model. Note that only renting households are eligible to receive rent allowance.

```
def calculate_rent_allowance(household, rent):
    # Households that have too much wealth do not get rent allowance
    if (household.size == 1 household.wealth > WEALTH_LIMIT_SINGLE_HOUSEHOLD)
    or (household.size > 1 and
            household.wealth > WEALTH_LIMIT_MULTI_PERSON_HOUSEHOLD):
        return 0

    quality_discount_limit = 0.58 * LIBERALIZATION_THRESHOLD

    if household.age < 23:
        max_rent_limit = quality_discount_limit
    else:
        max_rent_limit = LIBERALIZATION_THRESHOLD

    # If the rent is too high, households do not get rent allowance
    # For households younger than 23 this limit is lower.
    if rent > max_rent_limit:
        return 0

    # Yearly, income we use in our calculations. If income is less than zero,
    # we set this number to zero.
    test_income = max(0, household.income * 12)

    # MIN_INCOME_LIMIT, A_FACTOR, B_FACTOR and TARGET_AMOUNT are parameters
    # that depend on the properties of a household. They are set by the government.
    if test_income <= MIN_INCOME_LIMIT:
        basic_rent = minimal_basic_rent
    else:
        basic_rent = A_FACTOR * test_income ** 2 + B_FACTOR * test_income + TARGET_AMOUNT

    if household.size <= 2:
        capping_limit = 0.84 * LIBERALIZATION_THRESHOLD
    else:
        capping_limit = 0.9 * LIBERALIZATION_THRESHOLD

    # Rent allowance consists of three parts
    part_a = max(0, min(rent, quality_discount_limit) - basic_rent)
    part_b = max(0, min(rent, capping_limit) - max(basic_rent, quality_discount_limit)) * 0.65
    part_c = max(0, rent - max(basic_rent, capping_limit)) * 0.40

    return part_a + part_b + part_c
```

Market

List buying house

When listing a buying house, we first calculate its market value. We list the house 4% above this value on the market.

```
def list_buying_house(house):
    value = MARKET_PRICE_PREDICTION_MODEL.predict(house)
    LISTINGS[house.id] = (value, 1.04 * house)
```

List rental house

When listing a rental house we first predict its rental price. We do this with the market price prediction model. If the predicted rent is below the liberalization threshold, we list the house as a social house. Otherwise, we list the house as a private rental house. In this case, we list the house with a rent 4% above its current market value.

For social rental listings, we also determine if we show the listing to houses with a high income. 15% of social listings are available for high-income households.

```
def list_rental_house(house):
    rent = MARKET_PRICE_PREDICTION_MODEL.predict(house)

    if rent <= LIBERALIZATION_THRESHOLD:
        if random() <= 0.15:
            is_available_for_high_income_households = True
        else:
            is_available_for_high_income_households = False

        LISTINGS[house.id] = SocialListing(value, house, is_available_for_high_income_households)
    else:
        LISTINGS[house.id] = PrivateListing(1.04 * value, house)
```

Make brochures

Brochures display a selection of all listed houses on a specific market. For the buying market, we first get such a selection by selecting listed houses at random. We filter out all houses with a value higher than the maximum bid of a household.

```
def make_buying_brochure(max_bid):
    brochure = random_sample(LISTINGS, size=BROCHURE_SIZE)

    return filter(lambda listing: listing.value <= max_bid, brochure)
```

For the private rental market, we select as many houses as we need to fill the brochure from the listed houses. We do not filter these houses. So, potentially households can rent a house with a rental price higher than their income. However, they will only do so if this increases their utility. We do not expect this to happen often.

```
def make_private_rental_brochure(private_listings):
    return random_sample(private_listings, size=BROCHURE_SIZE)
```

For the social rent market, the social houses we use for the brochure depend on a household's income. For households with a low income, we select from all listed social houses. For households with a too high income, we only select from listed social houses that are for households with a high income.

```
def get_social_brochure(household, social_listings):
    if (household.size == 1 and household.income > INCOME_LIMIT_SINGLE_HOUSEHOLD)
          or (household.size > 1 and household.income > INCOME_LIMIT_MULTI_PERSON_HOUSEHOLD):
        # Only select listings available for households with a high income
        social_listings = filter(lambda listing: listing.is_available_for_high_income_households, social_l

    return random_sample(social_listings, size=BROCHURE_SIZE)
```

Get best option from brochure

We select the best option from a brochure by calculating the utility of each listing in the brochure. We then select the listing with the highest utility as the best option.

```
def get_best_option_from_market(brochure, household):
    best_option = (-infinity, None)

    for listing in brochure:
        # Note that for buying listings we first convert the listing.value
        # to a monthly mortgage payment
        monthly_payment = listing.value

        listing_utility = utility(household, listing.house, monthly_payment)

        if listing_utility > best_option.utility:
            best_option = (listing_utility, listing)

    return best_option
```

Assign social housing

For social listings with at least one household interested in it, the household with the longest waiting time gets the house.

```
def assign_social_housing(social_listings_with_interested_households):
    for listing in social_listings_with_interested_households:
        longest_waiting_household = max(listing.interested,
            key=lambda household: household.wait_list_time)
        move(longest_waiting_household, listing.house)
```

Update prices

Each month a buying house is not sold or a private listing is not rented out, we decrease its listed value by 3%. Note, that we do not decrease the listed value of social rental houses.

```
def update_prices(listings):
    for listing in listings:
        listing.value = 0.97 * listing.value
```

Household

Saved money

We described how households calculate how much money they save in Section 6.3.1. In the listing below, we show how this process works in detail. First, we determine how much tax a household pays. Next, we determine how much essential costs a household has, based on its size. The essential costs are set for a two-person household. We scale this amount linearly by the size of the household. Using these costs and the current housing costs of a household, we calculate how much money a household has left over. We add the rent allowance a household receives to this amount. We subtract a non-linear amount of the leftover money if the leftover amount is above zero.

```
def saved_money(household, rent_allowance, housing_costs):
 # Households only pay tax if their income is positive
    tax = 0
    if income > 0:
        tax = income * TAX_RATE

    essential_costs_by_household_size = (0.9 + 0.05 * household.size) * ESSENTIAL_COSTS
    left_over = household.income + rent_allowance -
                essential_costs_by_household_size - housing_costs - tax

    # If households have money left after paying obligatory costs, they spend some
    # of this money on non-essential "things".
    if left_over > 0:
```

```
        spend = round(2 * left_over ** 0.75)
        return left_over - spend

    return left_over
```

Gain wealth

Based on its saved money a household gains wealth. If the amount of saved money is positive, we add it to the household's current wealth. If the amount of saved money is less than zero, the household still owes money. If it has enough wealth to pay the owed amount, it does so. Otherwise, the household becomes homeless.

```
def gain_wealth(saved_money, household):
    if saved_money >= 0:
        household.wealth += left_over_money
    elif saved_money < 0 and household.wealth > abs(saved_money):
        household.wealth -= abs(left_over_money)
    else:
        household.become_homeless()
```

Sell

When a house owner sells their house, they receive the money. Using this money, they pay off their current mortgage. If the received amount is not enough to do this, the household uses its wealth to pay off the remainder of the mortgage.

```
def sell(household, sell_value):
    winnings_from_selling = value - household.mortgage_remaining
    if winnings_from_selling > 0:
        household.wealth += winnings_from_selling
    else:
        household.wealth -= abs(winnings_from_selling)
```

Act

A household selects the best option from all brochures it receives. It then acts on the absolute best option. If the utility of this best option is not better than the utility of its current situation, the household stays in its current house. Otherwise, the household decides it wants to move.

```
def act(household, best_buying_option, best_rental_option,
        best_social_rental_option):
    best_option = max(best_buying_option, best_rental_option,
                      best_social_rental_option)

    if best_option.utility > household.current_utility:
        move(household, best_option)
```

Changing houses

When a household decides to move, the actions it takes depend on its current house and the new house.

If the household is currently buying, it first has to sell its current house. The household uses the fact that there is a better option available as a trigger to change its housing situation. Therefore, the household lists its house for sale. When the house is sold, the household becomes homeless and starts looking for a new house.

If the best option house is a social house, we add the household to the list of interested households. We resolve these lists of interested households at the end of the month. Then, we move the interested household with the longest waitlist time into the house.

Otherwise, the household moves immediately into the best option. First, its current house gets listed on the rental market above market price. Next, the household moves into the new house. For both the buying and rental market auctions, we use a first-come first-serve approach as described in Section 5.6.2.

For rental houses, a household gets a contract for an undetermined period at the listed rent price. We fix the rent for the entire duration of the contract. So, the rent does not change for as long as the household rents the house.

When buying a house, a household first gets a mortgage from the bank. This mortgage determines how much a household will pay for housing each month. The duration and interest rate of this mortgage is the same for each household. Households can pay a part of the buying price from their wealth. This lowers their mortgage and reduces the monthly payment.

In case a buying house has an owner, this owner gets paid the selling price. The owner uses this money to pay off their current mortgage. All leftover money, if any, is added to their wealth. It can be the case that the paid money is not enough to pay off the mortgage. A household now uses its wealth to help pay off the mortgage.

```
def move(household, new_house):
    if household.contract.house is BuyingHouse:
        list_buying_house(household.contract.house)
    elif new_house is SocialHouse:
        new_house.show_interest(household)
    elif household.contract.house is RentalHouse or SocialHouse:
        list_rental_house(household.contract.house)
        move_in(household, new_house)
    elif household is homeless:
        move_in(household, new_house)

def move_in(household, house):
    if house is BuyingHouse:
        sell(house.owner, house.buying_price)
        house.owner = household
        mortgage = house_value_to_monthly_payment(house.buying_price, household.wealth)
        household.contract = Contract(mortgage, house)
    elif house is RentalHouse:
        household.contract = Contract(house.rent, house)
```

Construction

As discussed in Section 6.3.2, we fix the number of constructed houses per year for all models. This means we know how many houses to construct in any given year. We draw the size of a new house from a distribution. This distribution is a parameter of the model. The rest of the process is the same as for the initialization of the model. We draw the quality from a uniform distribution. We also assign the same percentage of houses to the rental and buying markets.

```
def construct(year):
    to_construct = TO_CONSTRUCT_PER_YEAR[year] / SCALE_FACTOR

    for i in 0..to_construct:
        size = draw_from_size_distribution()
        quality = random(1, 10)

        if random() < PERCENTAGE_FOR_RENT:
            house = RentalHouse(size, quality)
            rental_market.list(house)
        else:
            house = BuyingHouse(size, quality)
            buying_market.list(house)

        houses.append(house)
```

Balance Population

We balance the population of households at end of each year. We do this to match the population with the expected distribution. Similar to the creation of households, we have slots with an age range, a household size, and a count. The age range has a minimum and a maximum age. The count states how many households we need for the slot. For all slots with the same age range, we run a balancing procedure. This procedure moves households from slots with too many households to slots with not enough households. It does this by changing a household's size.

```
def balance_households_for_age_range(slots_with_same_age_range):
    to_remove = []
    to_create = []
    for age_range, size, expected_count in slots_with_same_age_range:
        scaled_count = expected_count / SCALE_FACTOR
        households_currently_in_slot= filter(
                lambda h: age_range.min <= h.age < age_range.max and h.size == size, households)

        difference = scaled_count - len(households_currently_in_slot)

        # We need to remove households from this group
        if difference < 0:
            to_remove_from_slot = random_sample(households_currently_in_slot, abs(difference))
            to_remove.append(to_remove_from_slot)
        elif difference > 0:
            # Store how many households we need to create for this slot; And which size they have
            to_create.append((size, difference))

    for new_size, amount in to_create:
        for i in 0..amount:
            if len(to_remove) > 0:
                household = to_remove.pop()

                before_utility = household.current_utility()
                household.size = new_size
                after_utility = household.current_utility()

                # If the utility of the new situation is worse than before,
                # we mark that the household wants to move
                if after_utility < before_utility:
                    household.want_to_move = True
            else:
                # There are no existing households that we can modify,
                # so we need to create a new household
                households.add(self._create_household(min_age, max_age, new_size, model.year))

    # Remove any remaining households that we did not use above
    households.remove_all(to_remove)
```

Appendix B

Model parameters

Table B.1: Model parameters

Parameter	Description	Default value
Calibration length	Number of months the calibration of the model takes.	100
Scale factor	Factor by which we scale down the number of households and houses.	12,500
Start year	First year of a model run.	2012
End year	last year of a model run.	2060
Number of runs	Amount of times a model is run.	50
Initial rental market model	Initial rent market price prediction model.	$\text{price} = 250 \cdot \text{house size} + 1500 \cdot \text{house quality} + 180,000$
Initial buying market model	Initial buying market price prediction model.	$\text{price} = 4 \cdot \text{house size} + 8 \cdot \text{house quality} + 150$
Transactions for fitting	The number of most recent transactions used to fit the new parameters of a market price prediction model.	100
Brochure size	Maximum number of houses in a brochure.	10 listings
List price percentage	Percentage of the market price that is the value of a newly listed home. For both the rental and buying market.	104%
List price decrease	Percentage that a listed house decreases in value every month it is not rented out.	3%
Initial liberalization threshold	-	€750
Utility function	-	As shown in Section 6.4.1
Homelessness penalty	Penalty for being homeless. Subtracted from the utility.	5
Rental percentage	Percentage of all houses that is part of the rental market.	45%
New house size distribution	Distribution from which we draw the size and quality of new houses.	Skewed distribution with a mean of 80
New house quality distribution	Distribution from which we draw the size and quality of new houses.	Uniform distribution of integers 1 to 10
Cost of essential consumption	How much essential consumption costs for a two-person household per month. Value is scaled for other-sized households.	€400

Parameter	Description	Default value
Mortgage length	Number of years in which mortgage has to be repaid.	25 years
Mortgage interest rate	Interest rate of a mortgage.	5%
Minimum age	Minimum age of a household.	15 years
Maximum age	Maximum age of a household.	100 years
Minimum house size	-	10 m2
Maximum house size	-	1,000 m2
Minimum house quality	-	1
Maximum house quality	-	10
Minimal rental price	Minimal price for which house can get listed on the rental market.	€100
Minimal buying price	Minimal price for which a house can get listed on buying market.	€10,000
Wealth buying portion	Percentage of its wealth a household uses when buying a house.	30%
Waitlist time assignment percentage	Percentage of households we assign a waitlist time based on their age. All other households are assigned a waitlist time of zero.	75%

Appendix C

All results

Taxing m^2 per person

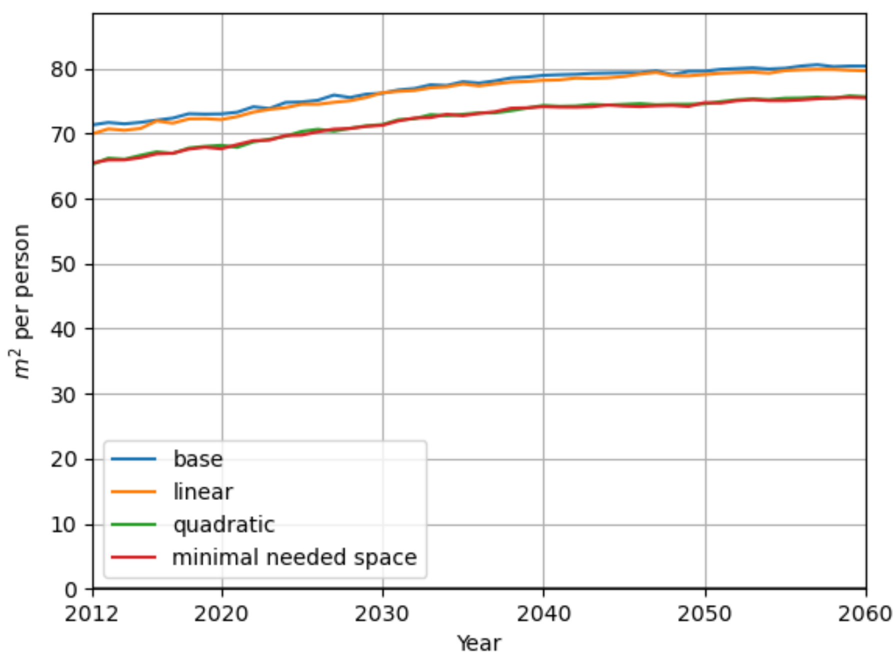

Figure C.1: Average space per person for taxation policy

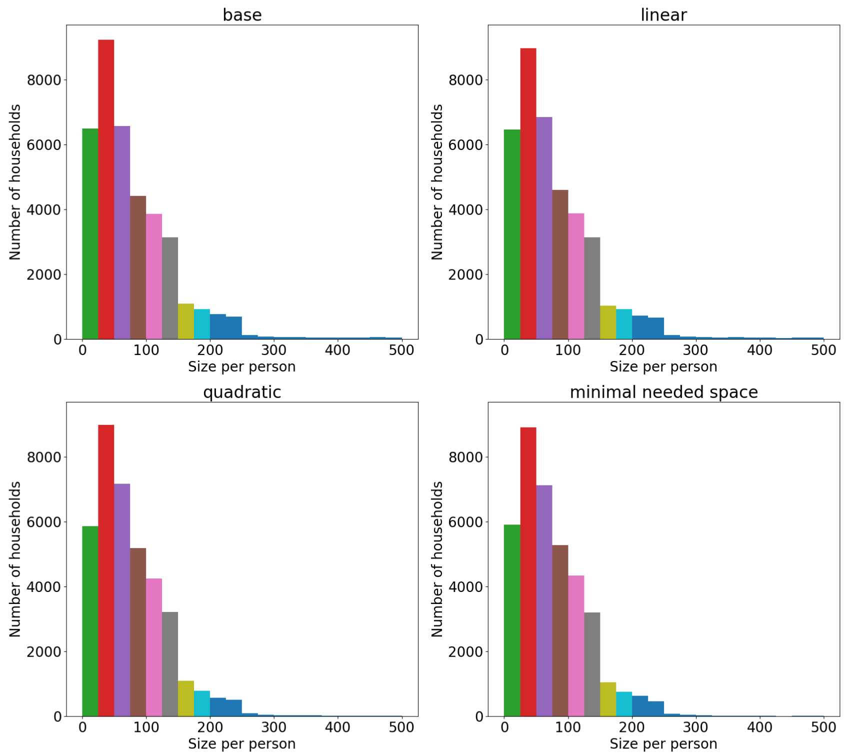

Figure C.2: Space per person histogram for taxation policy

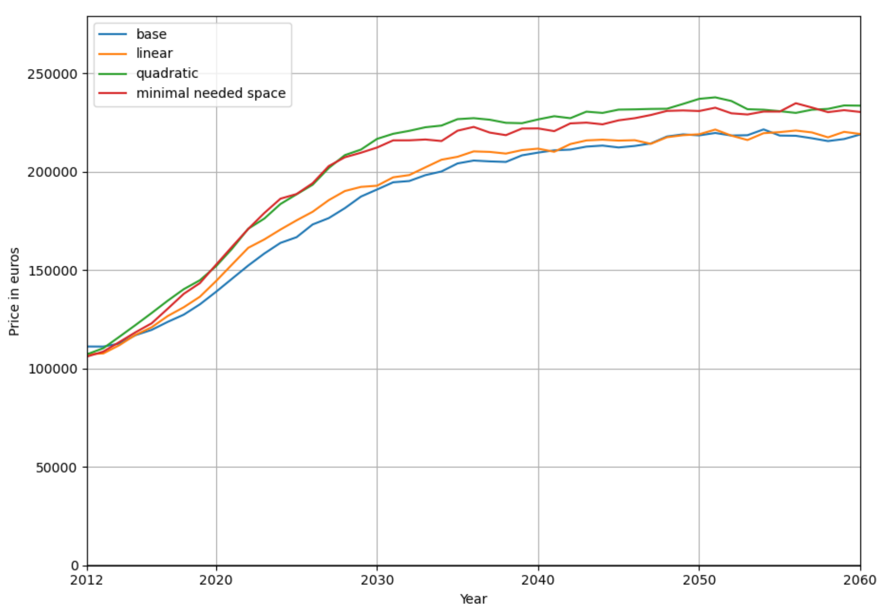

Figure C.3: Buying price for taxation policy

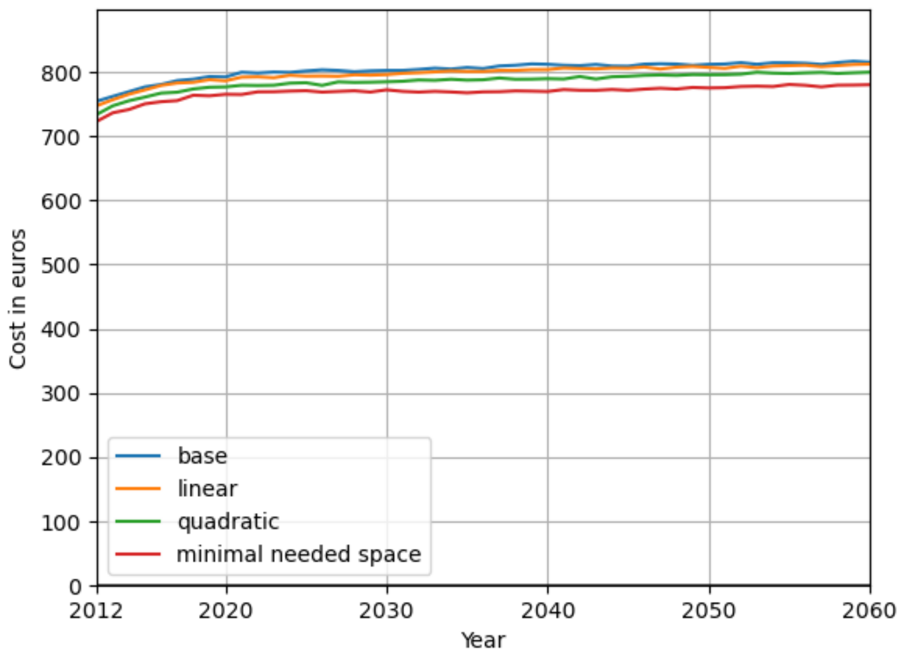

Figure C.4: Average rent for taxation policy

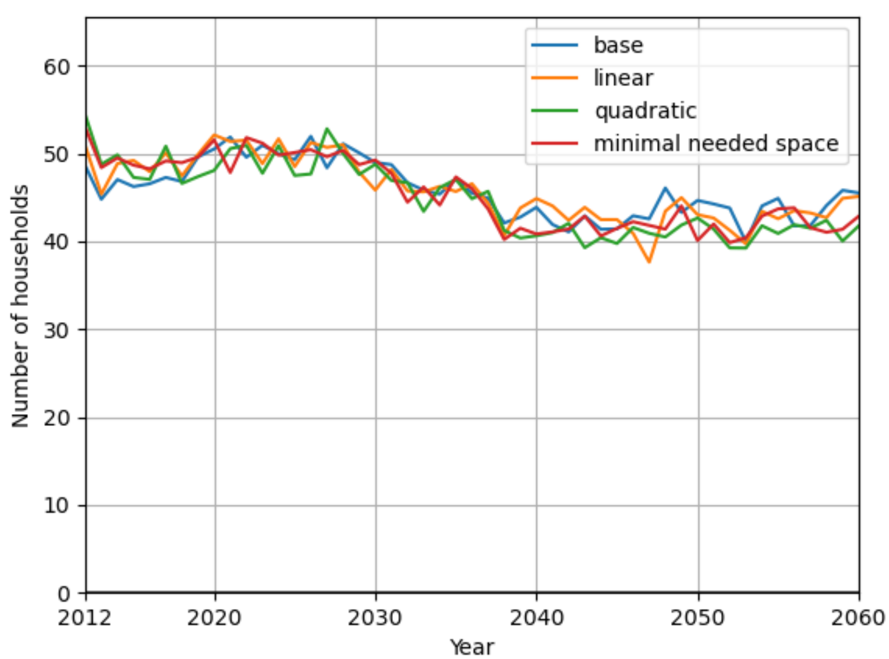

Figure C.5: Homelessness for taxation policy

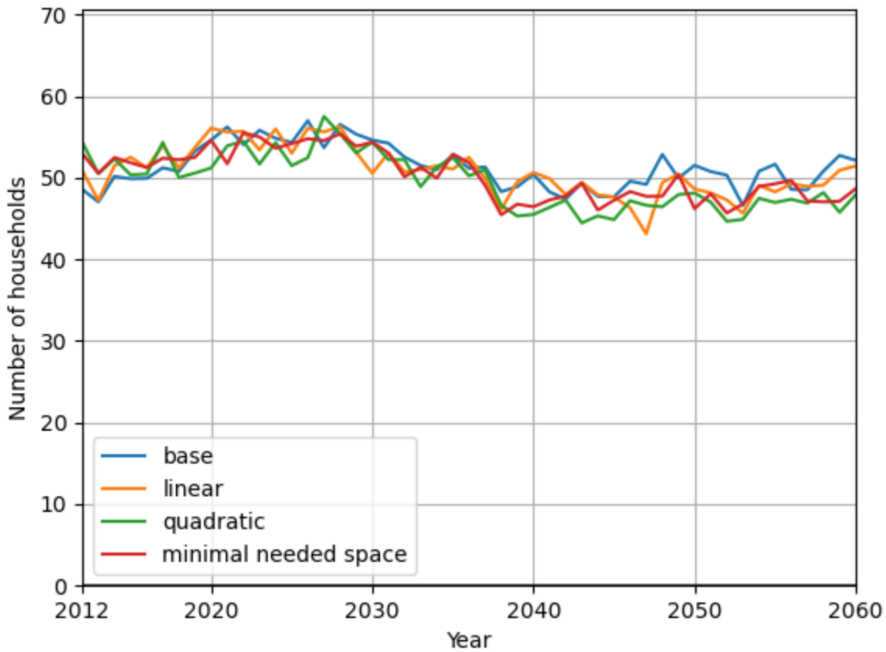

Figure C.6: Want to move count for taxation policy

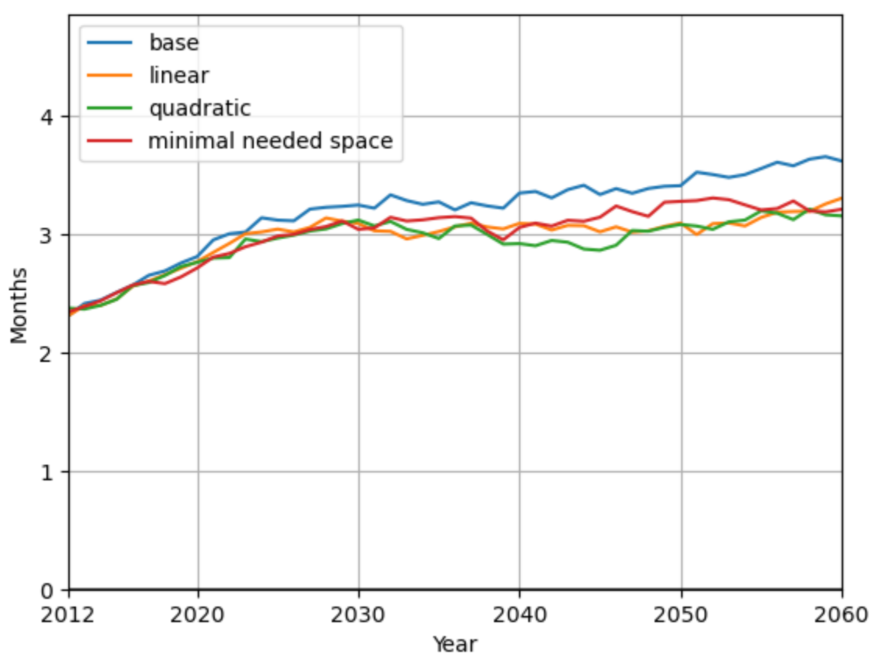

Figure C.7: Want to move time for taxation policy

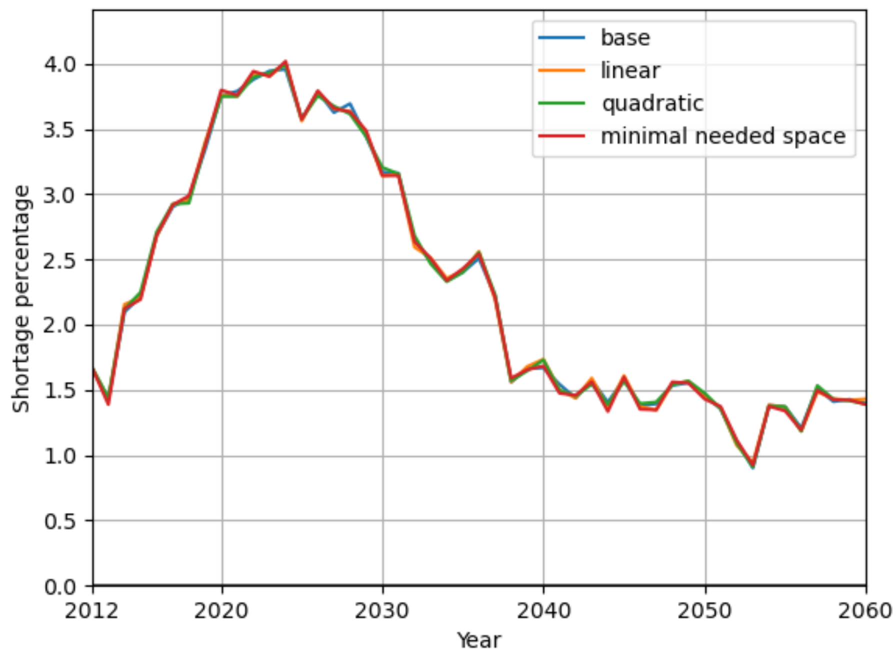

Figure C.8: Housing shortage for taxation policy

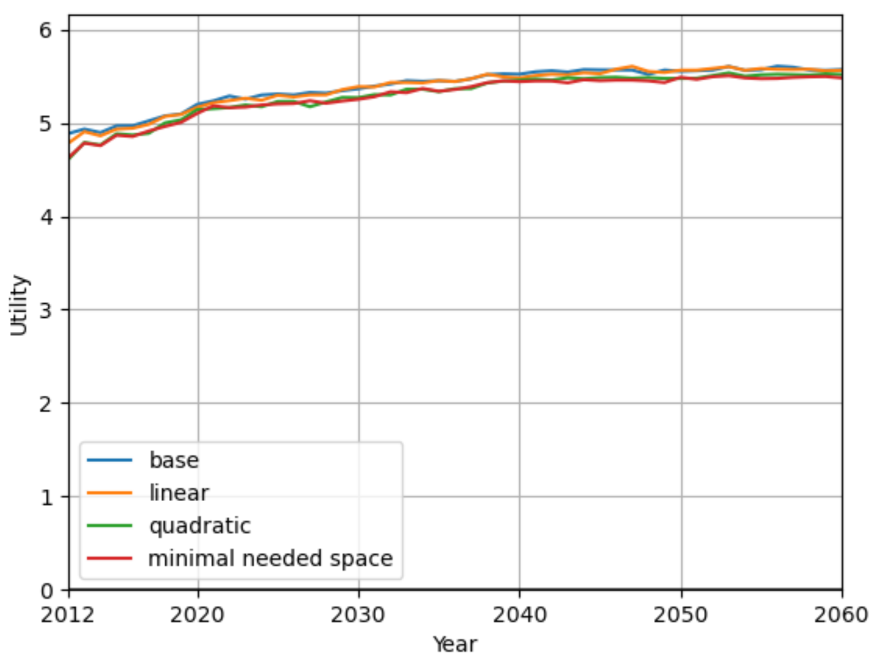

Figure C.9: Average utility for taxation policy

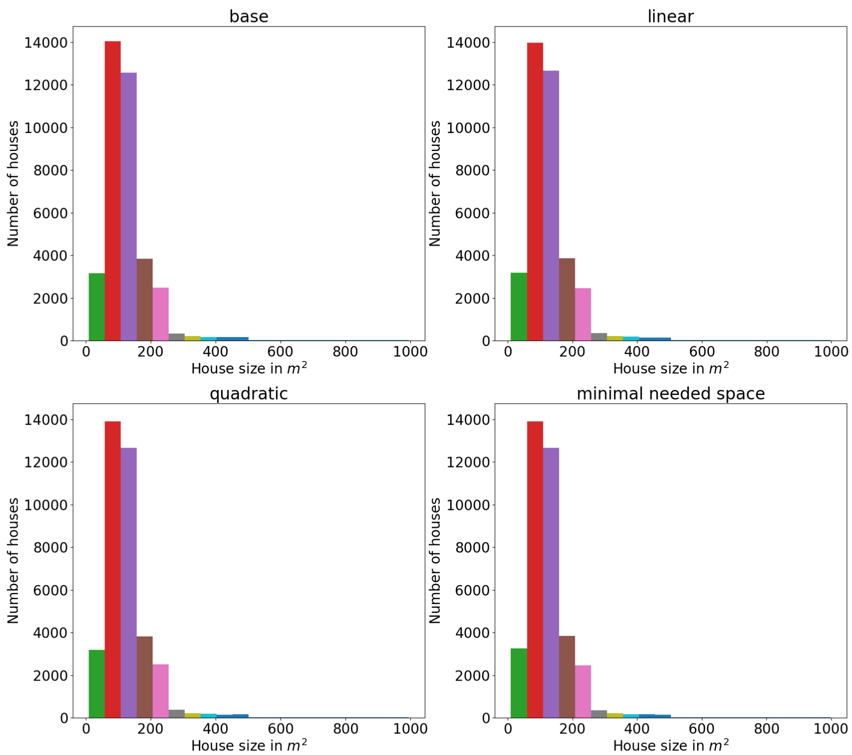

Figure C.10: House size histogram for taxation policy

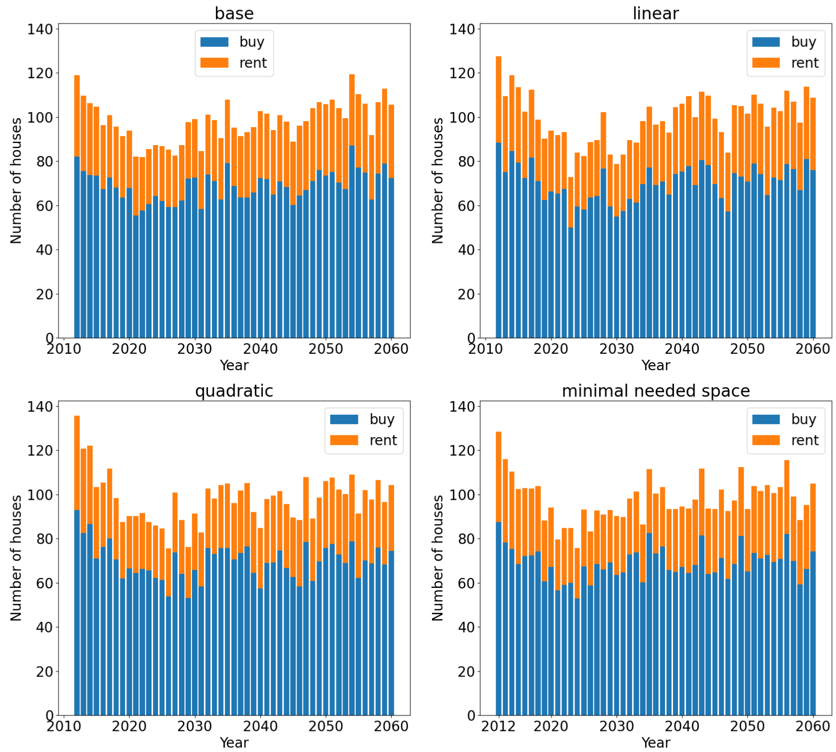

Figure C.11: Number of listed houses for taxation policy

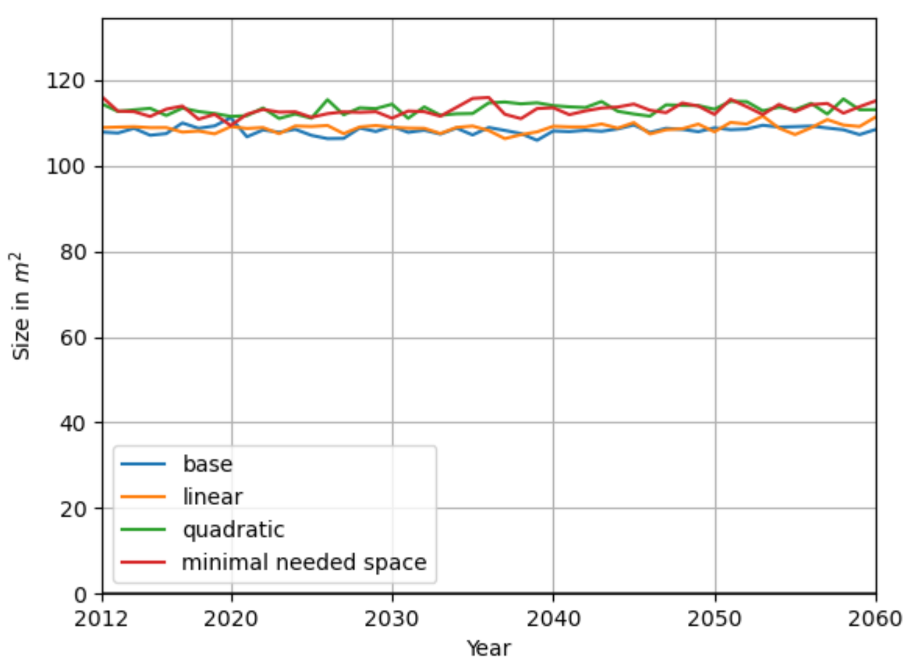

Figure C.12: Average size of listed houses for taxation policy

Limiting space per person

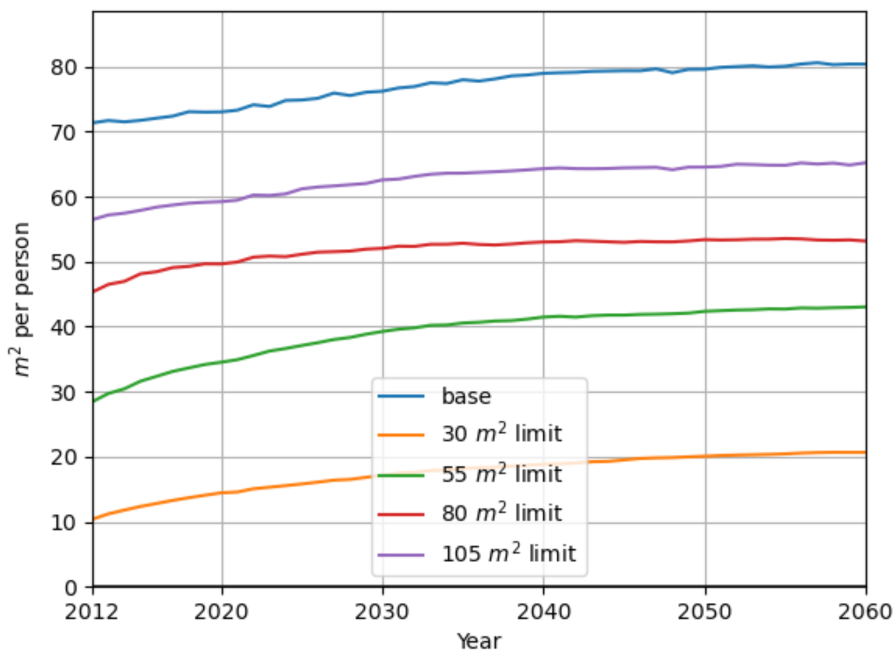

Figure C.13: Average space per person for limit policies

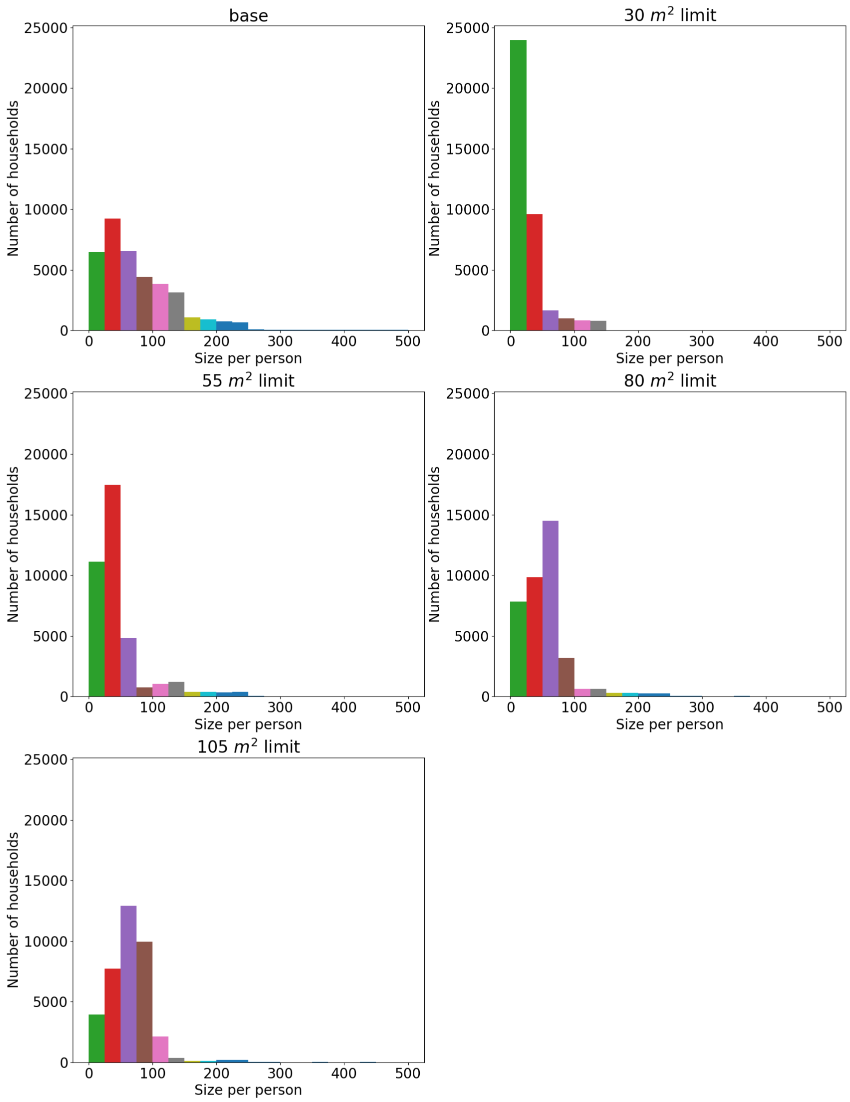

Figure C.14: Space per person histogram for limit policies

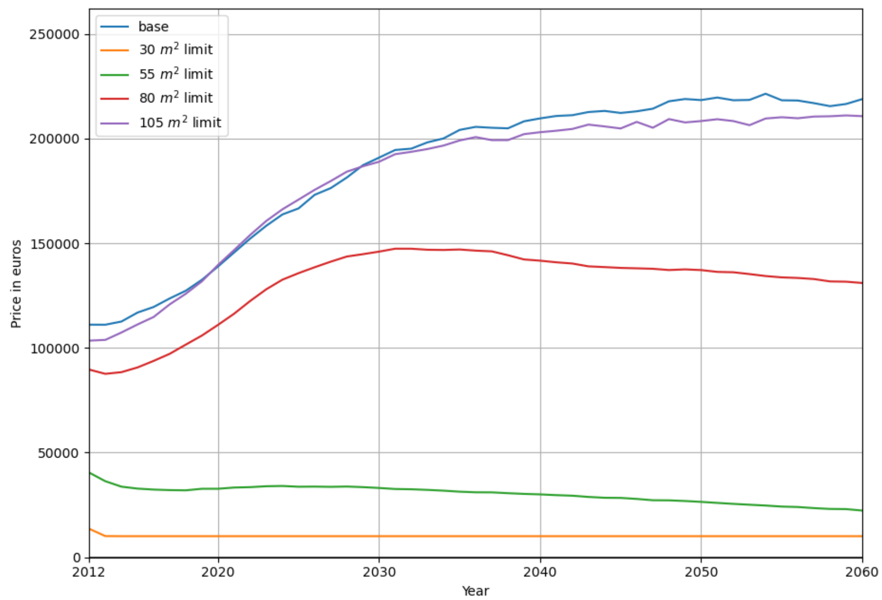

Figure C.15: Buying price for limit policies

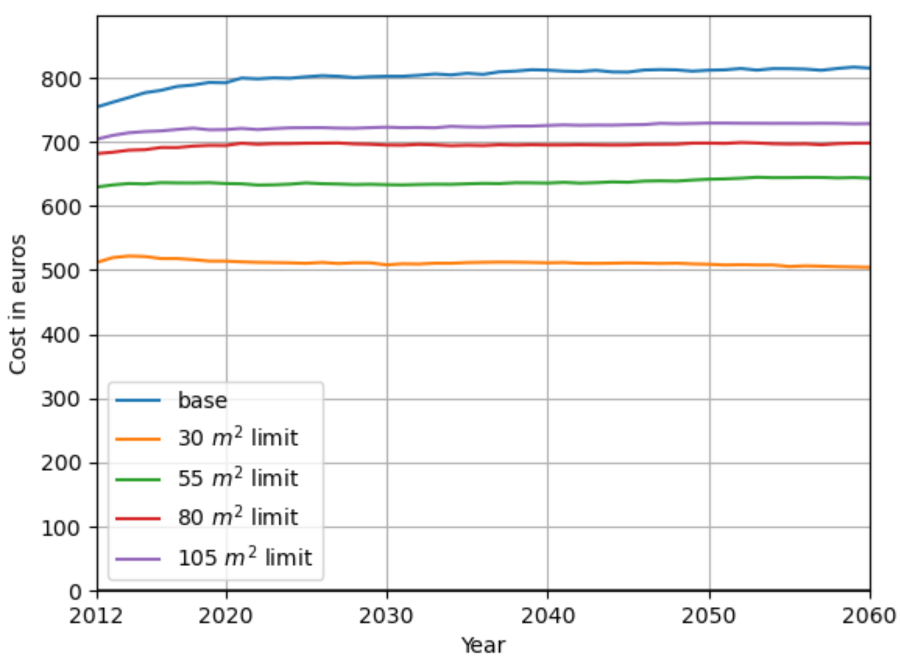

Figure C.16: Average rent for limit policies

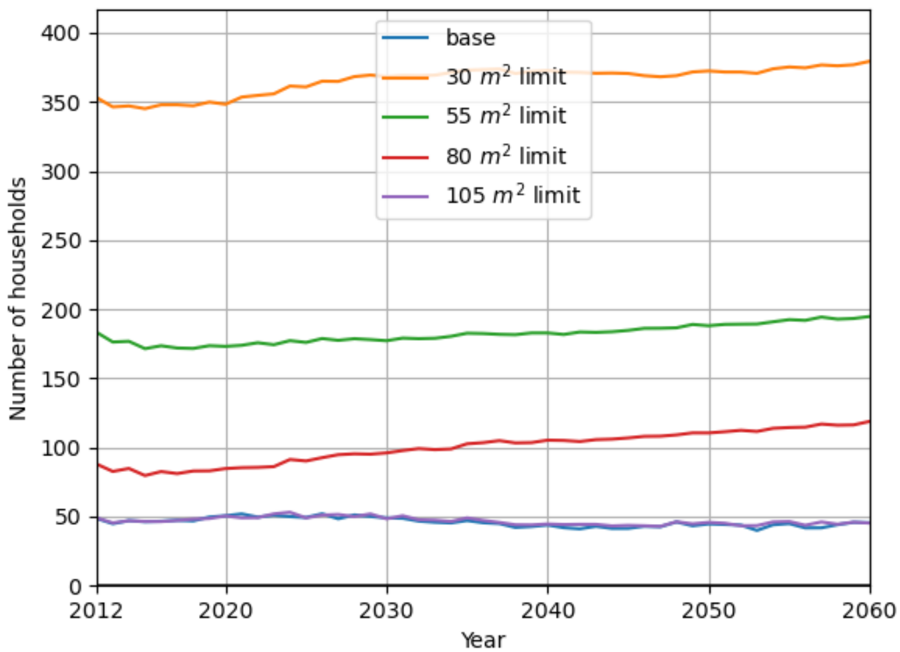

Figure C.17: Homelessness for limit policies

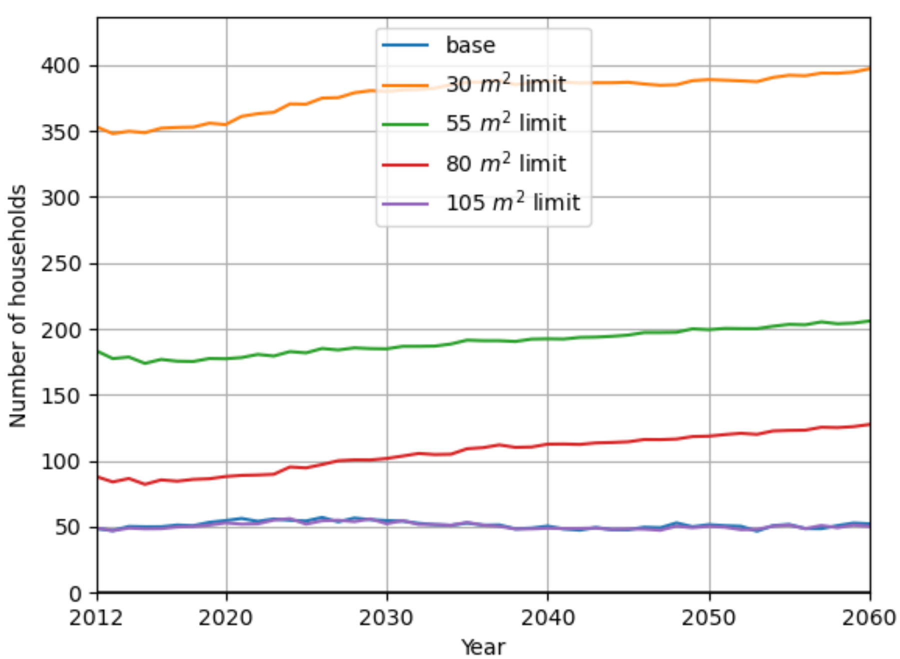

Figure C.18: Want to move count for limit policies

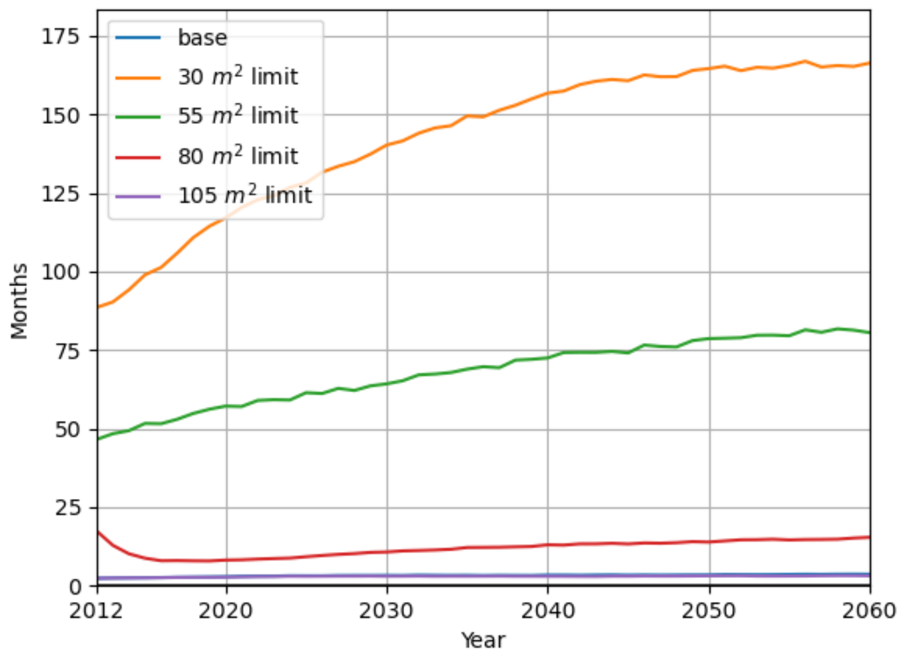

Figure C.19: Want to move time for limit policies

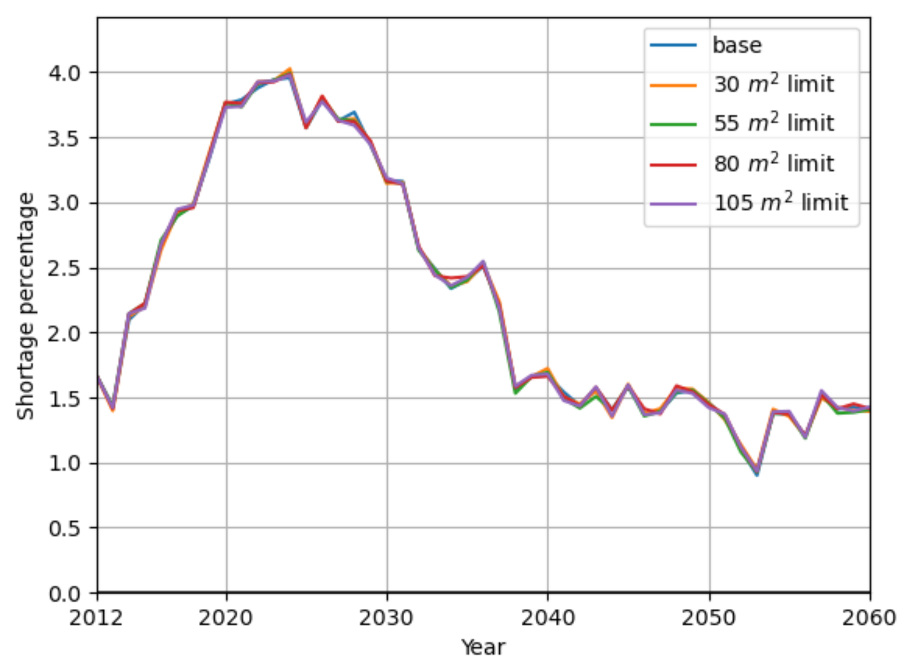

Figure C.20: Housing shortage for limit policies

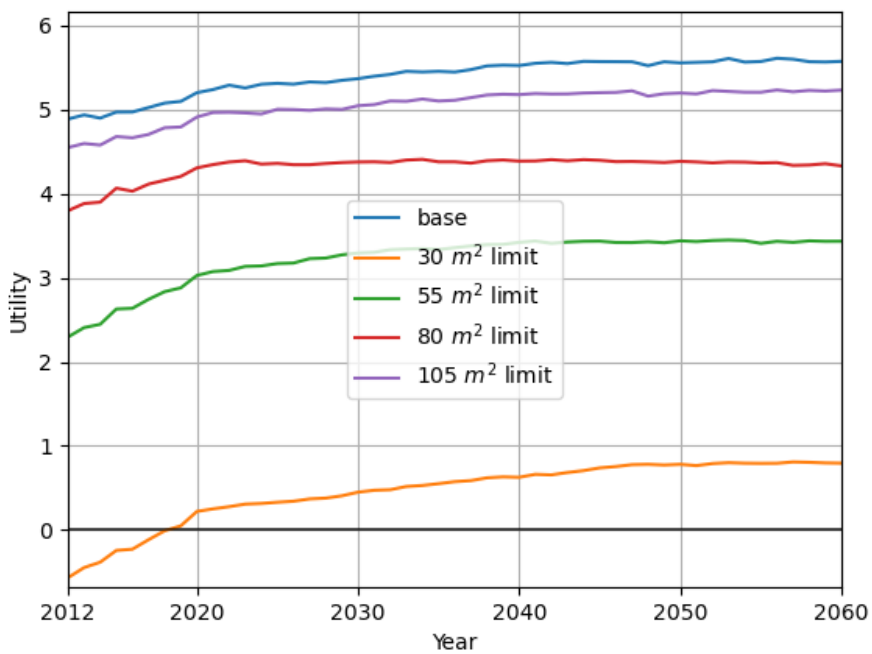

Figure C.21: Average utility for limit policies

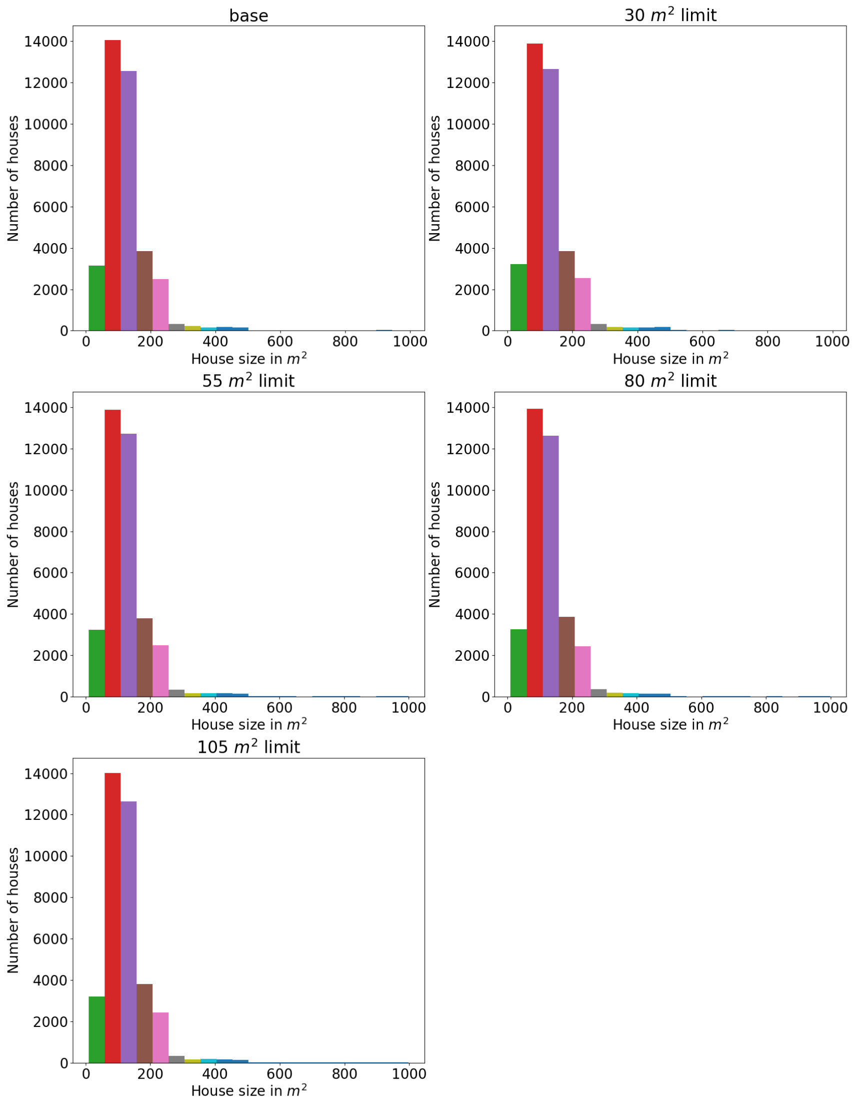

Figure C.22: House size histogram for limit policies

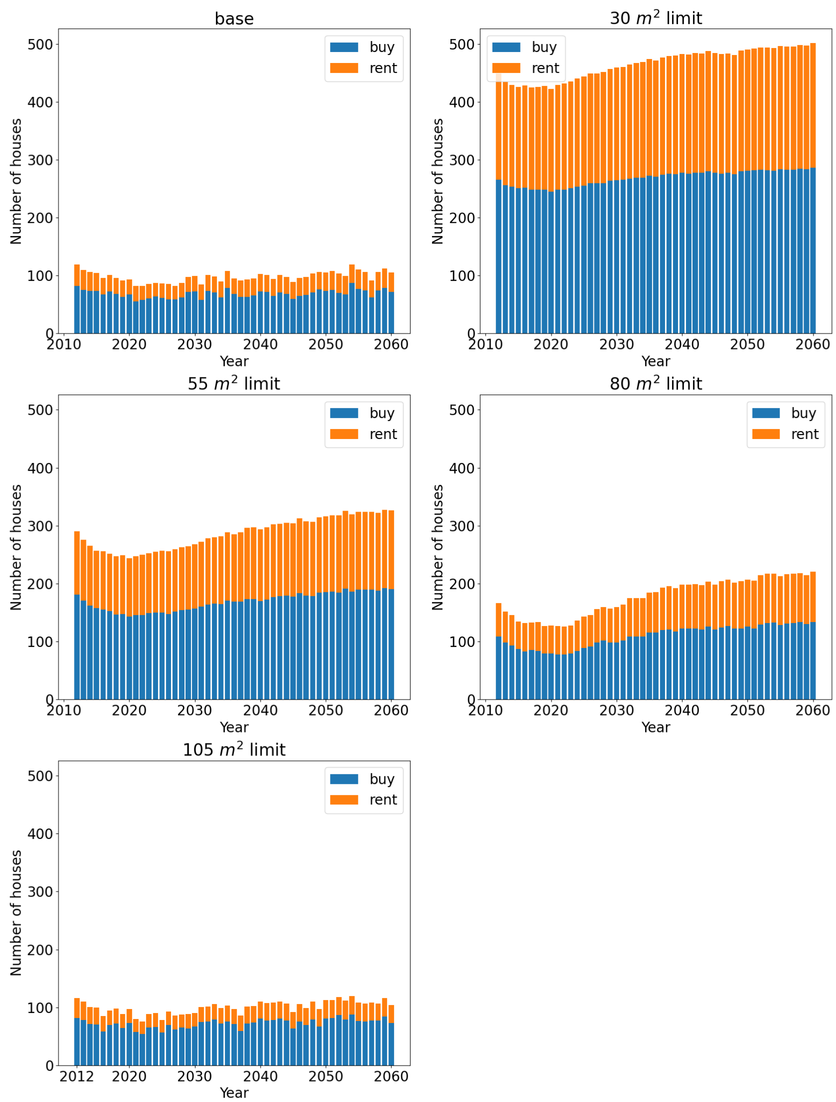

Figure C.23: Number of listed houses for limit policies

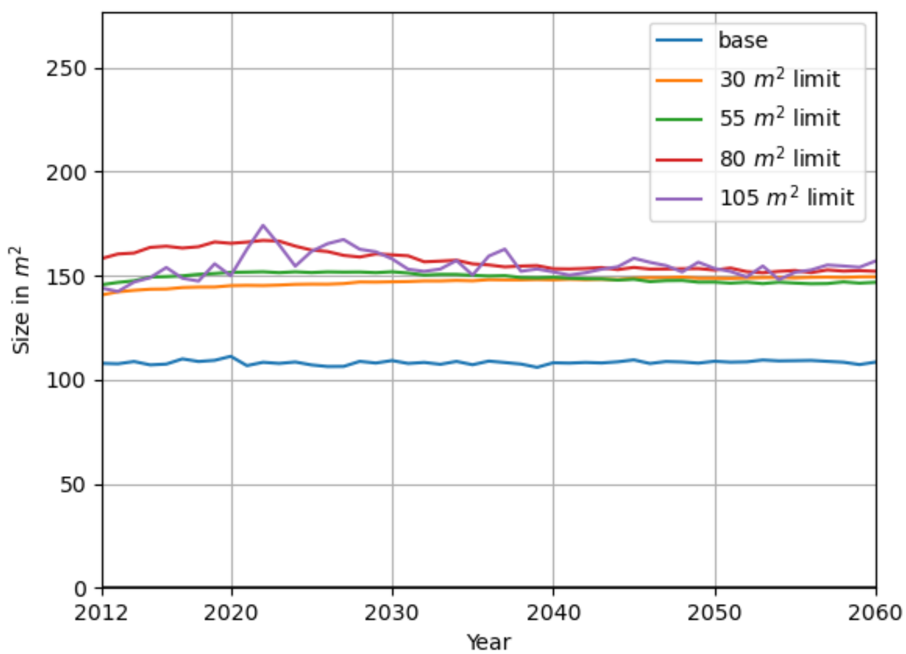

Figure C.24: Average size of listed houses for limit policies

Sharing rental houses

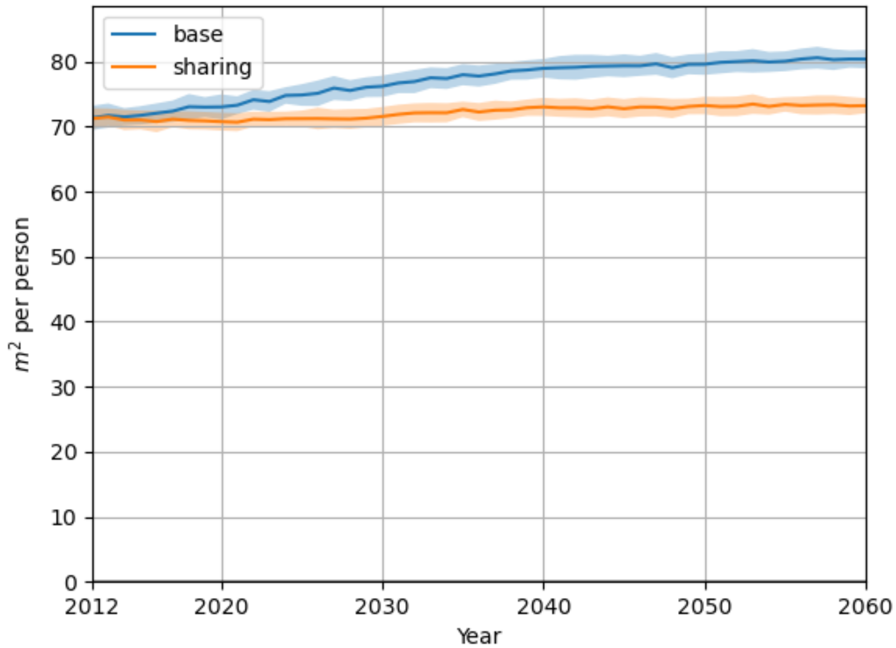

Figure C.25: Average space per person for sharing policy

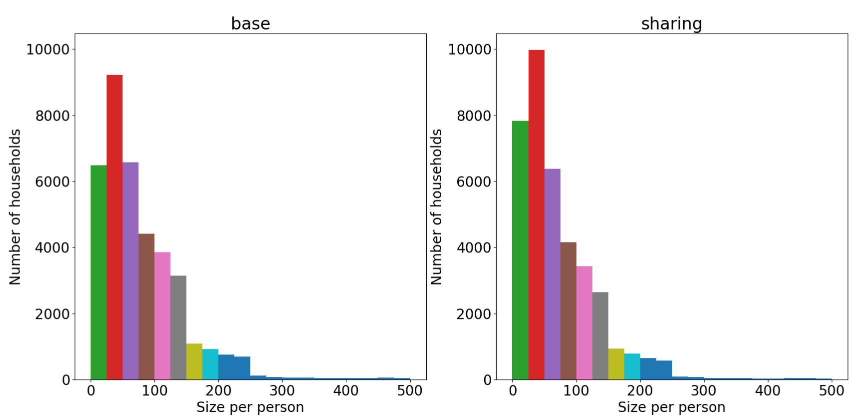

Figure C.26: Space per person histogram for sharing policy

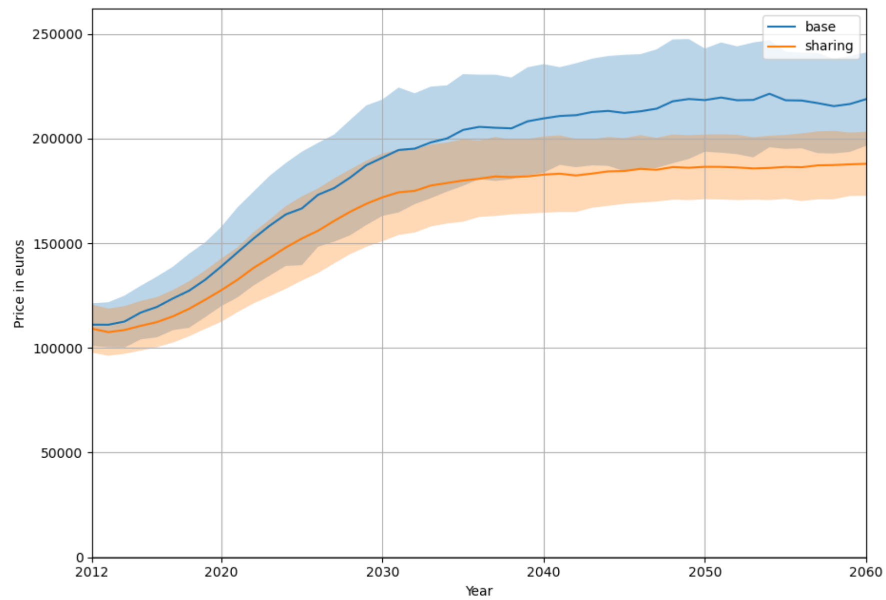

Figure C.27: Buying price for sharing policy

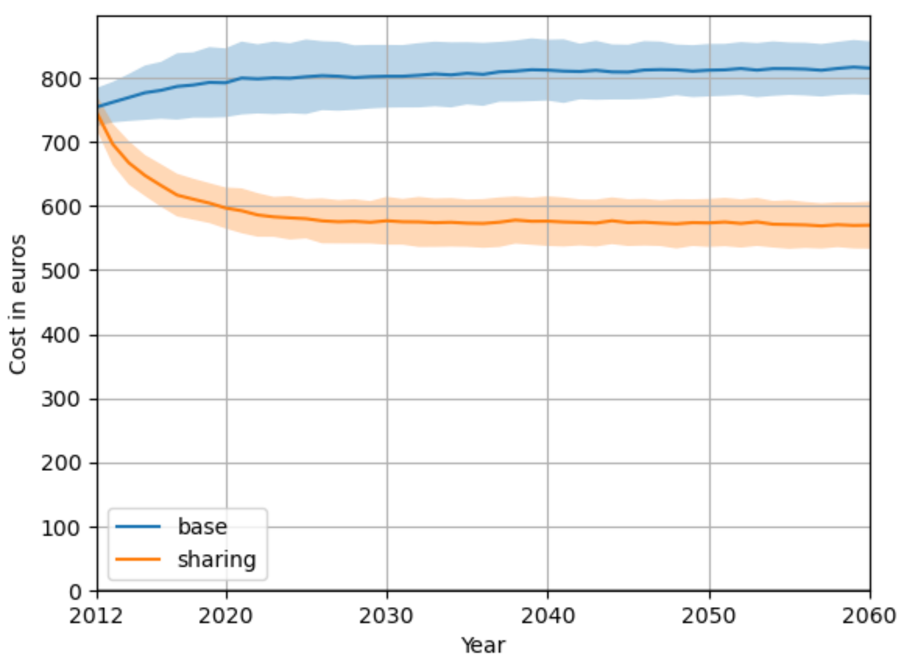

Figure C.28: Average rent for sharing policy

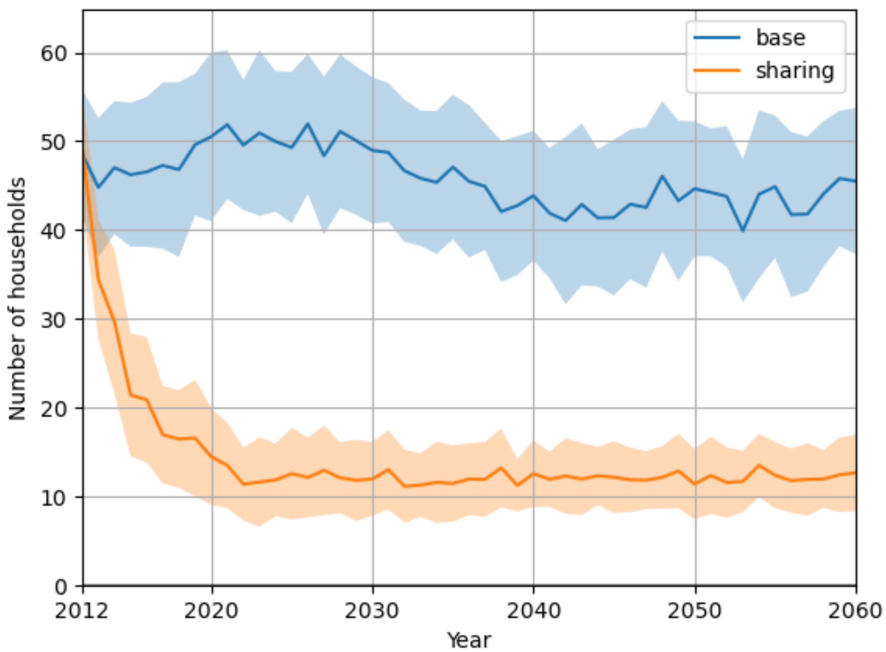

Figure C.29: Homelessness for sharing policy

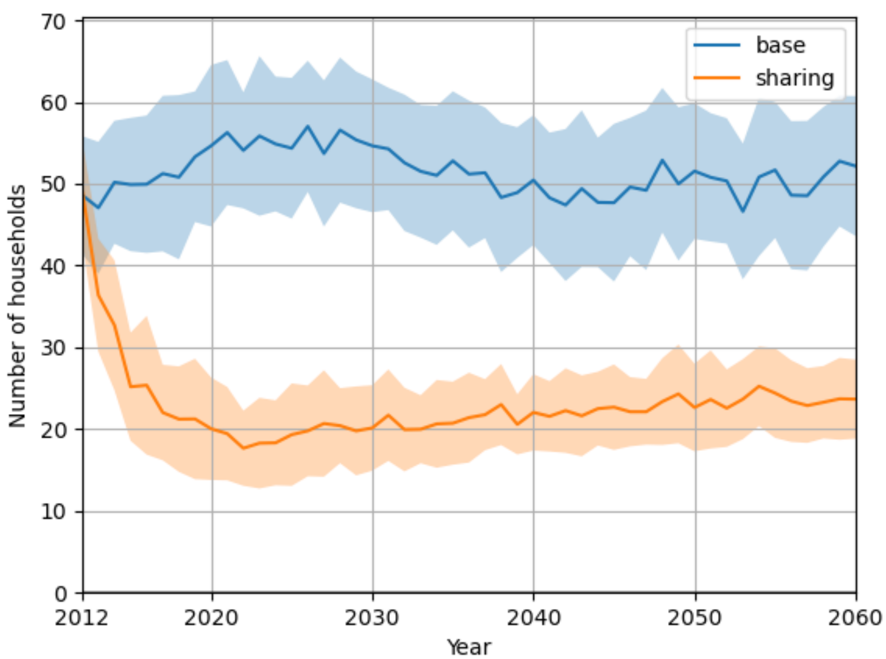

Figure C.30: Want to move count for sharing policy

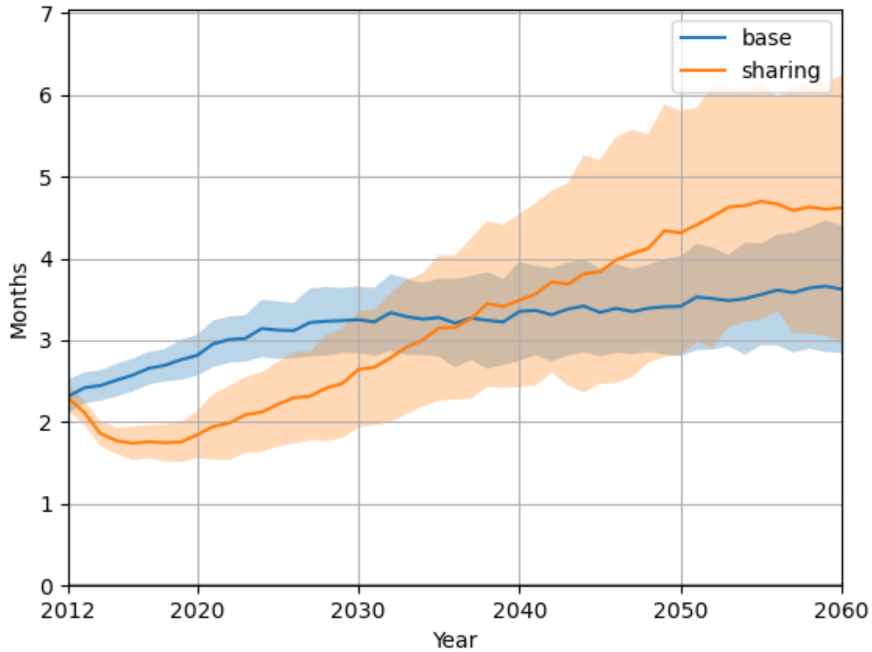

Figure C.31: Want to move time for sharing policy

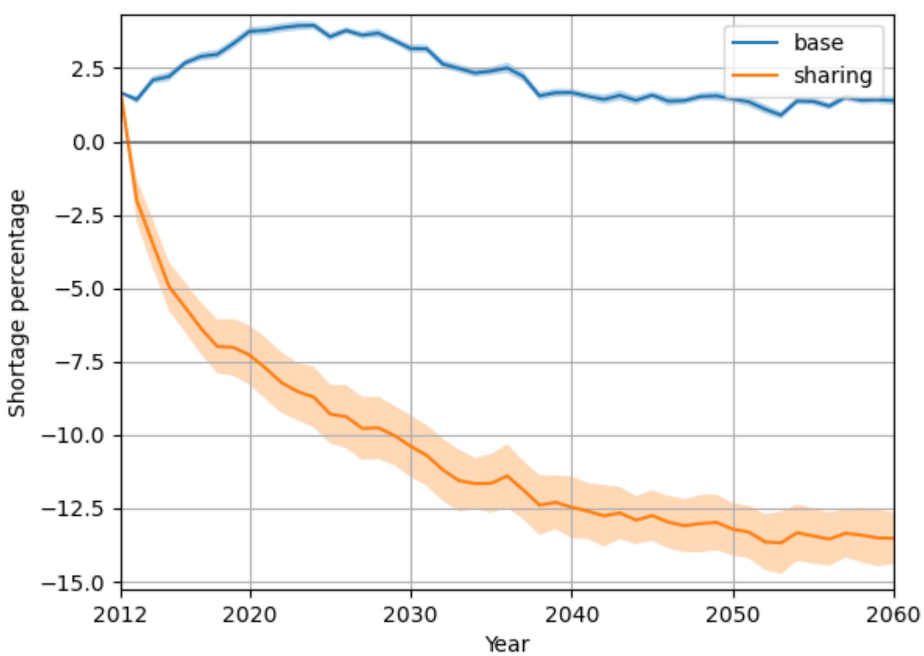

Figure C.32: Housing shortage for sharing policy

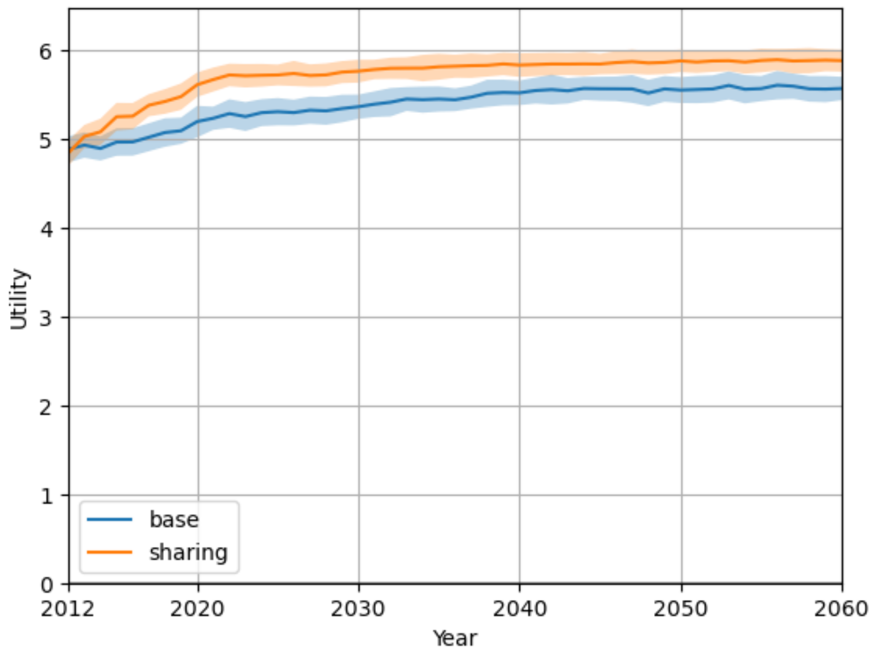

Figure C.33: Average utility for sharing policy

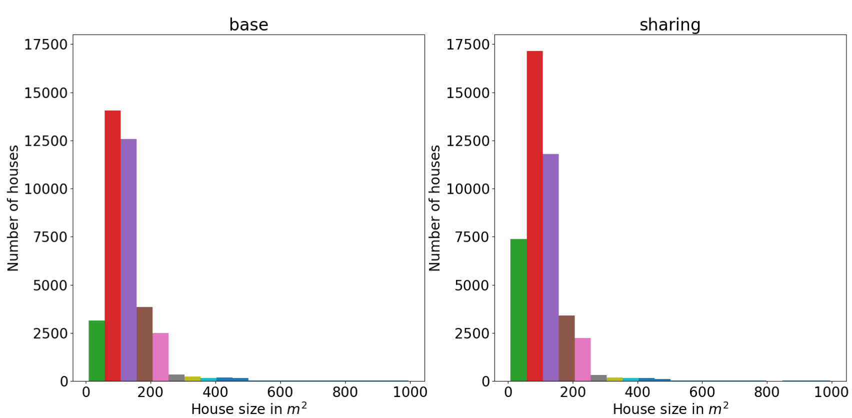

Figure C.34: House size histogram for sharing policy

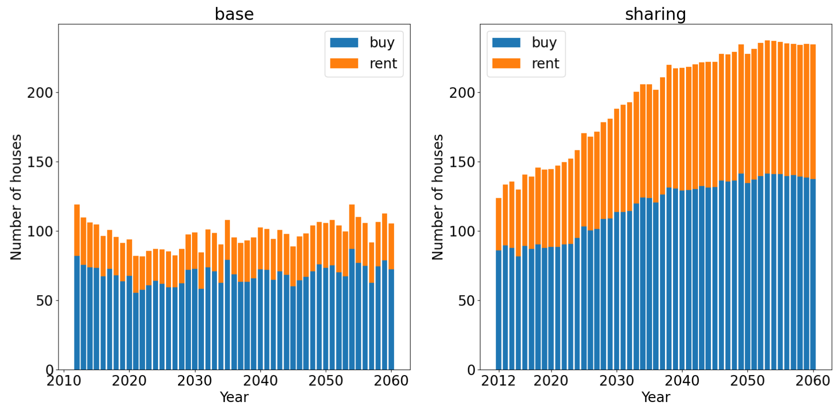

Figure C.35: Number of listed houses for sharing policy

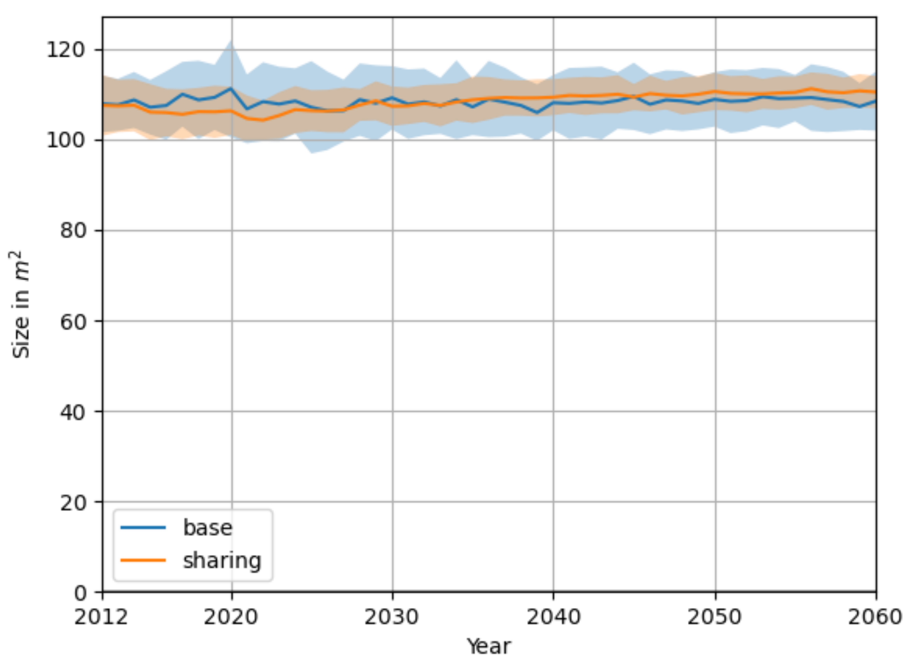

Figure C.36: Average size of listed houses for sharing policy

Splitting buy houses

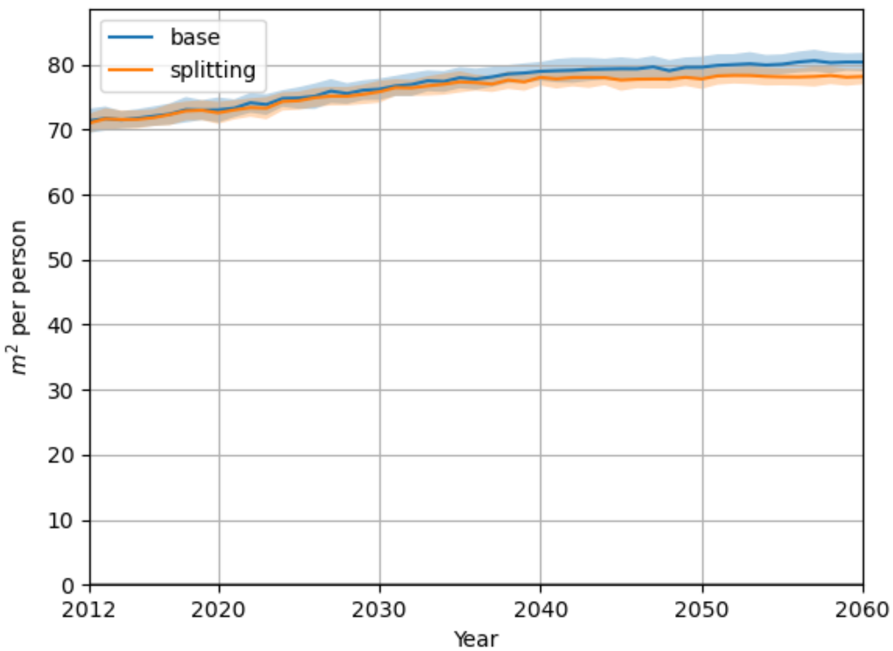

Figure C.37: Average space per person for splitting policy

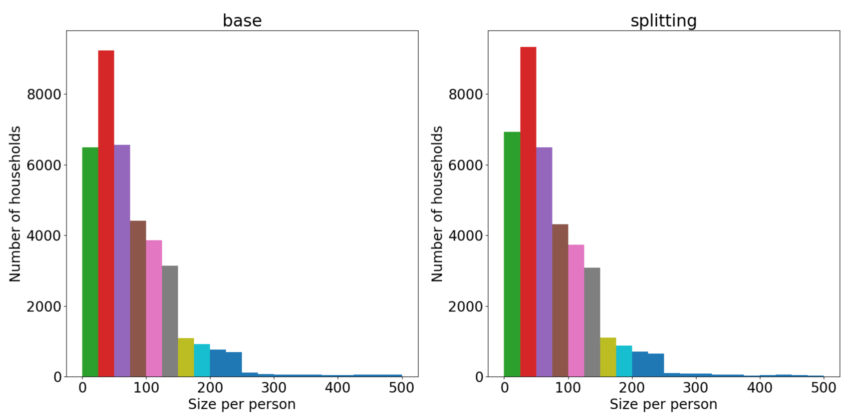

Figure C.38: Space per person histogram for splitting policy

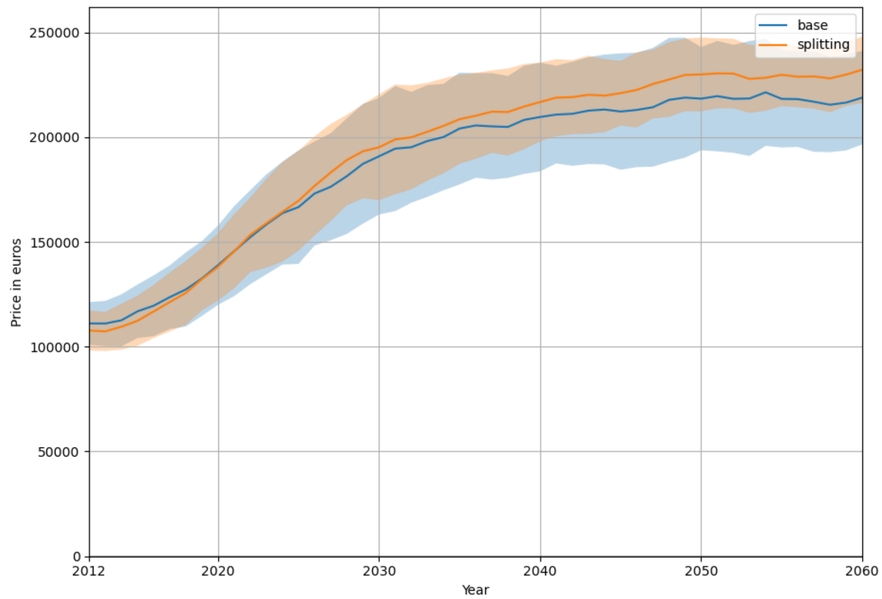

Figure C.39: Buying price for splitting policy

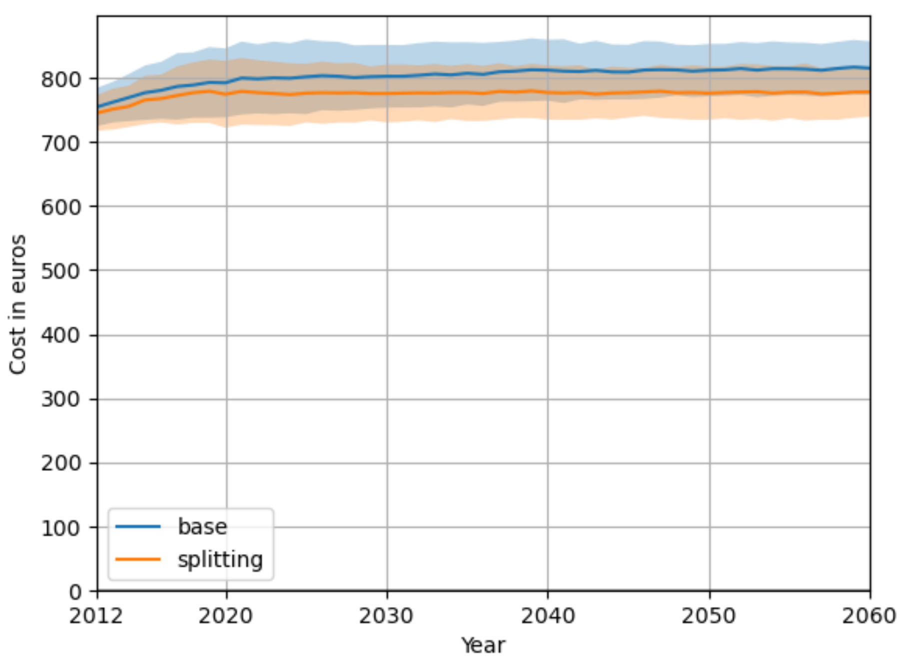

Figure C.40: Average rent for splitting policy

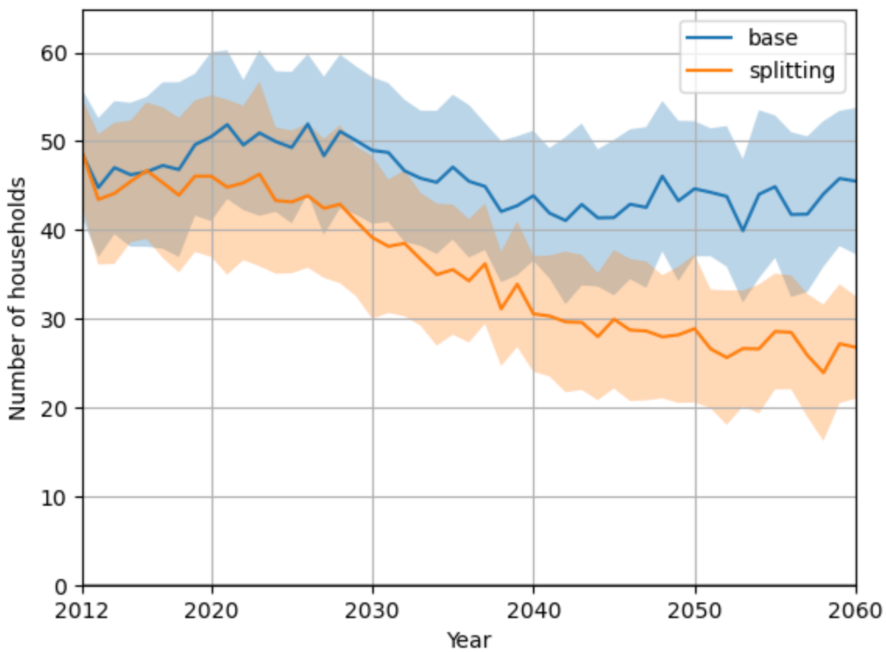

Figure C.41: Homelessness for splitting policy

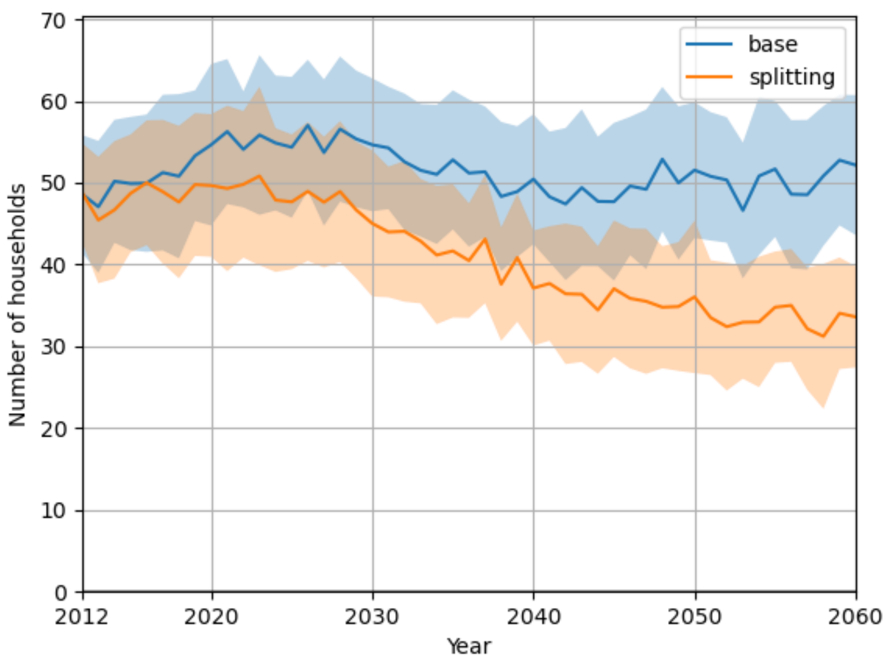

Figure C.42: Want to move count for splitting policy

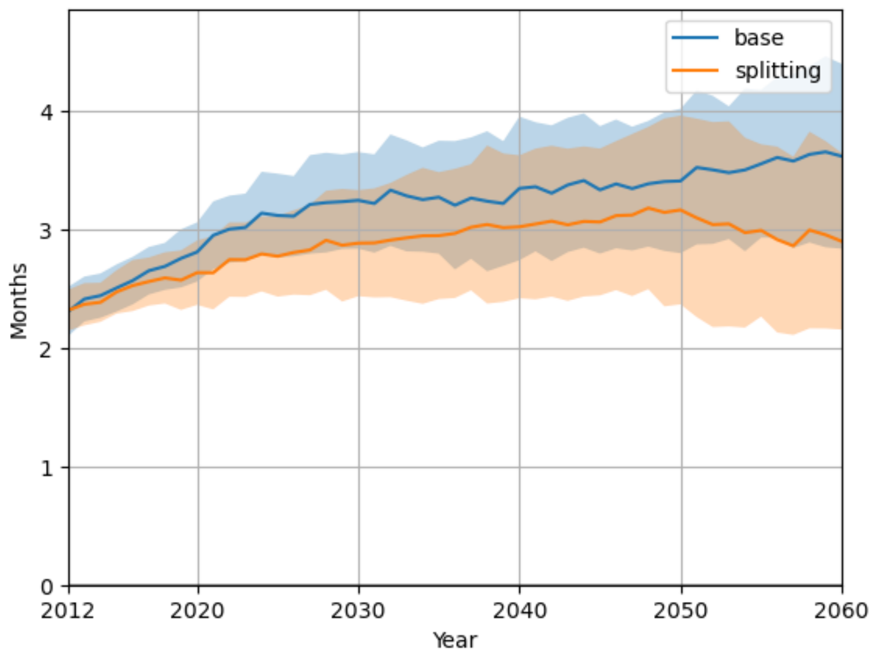

Figure C.43: Want to move time for splitting policy

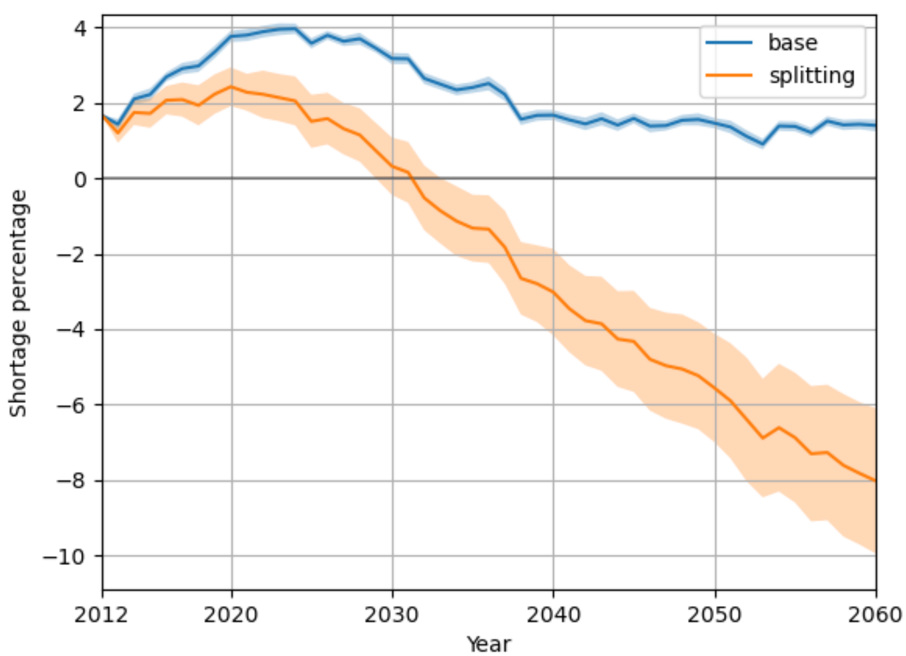

Figure C.44: Housing shortage for splitting policy

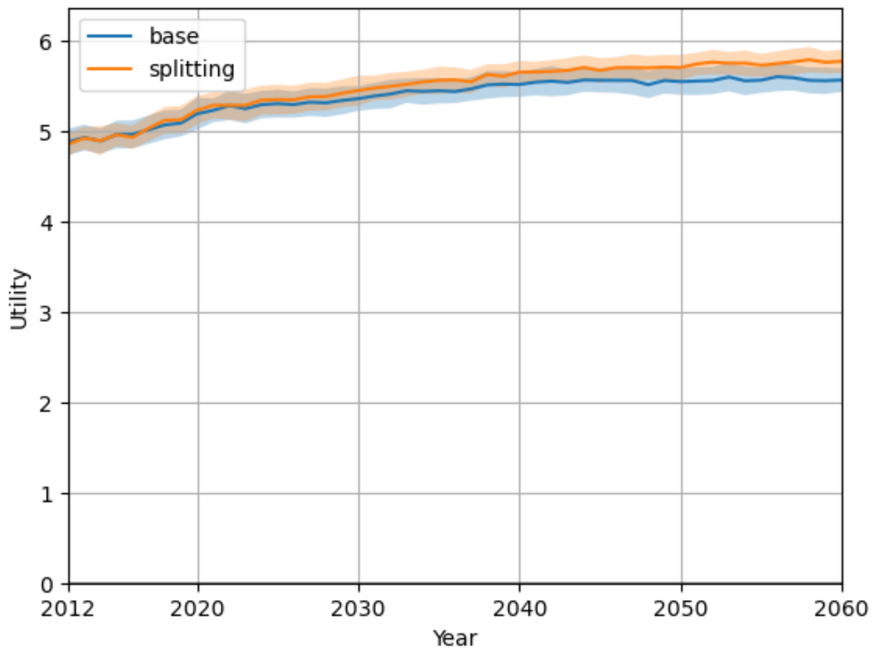

Figure C.45: Average utility for splitting policy

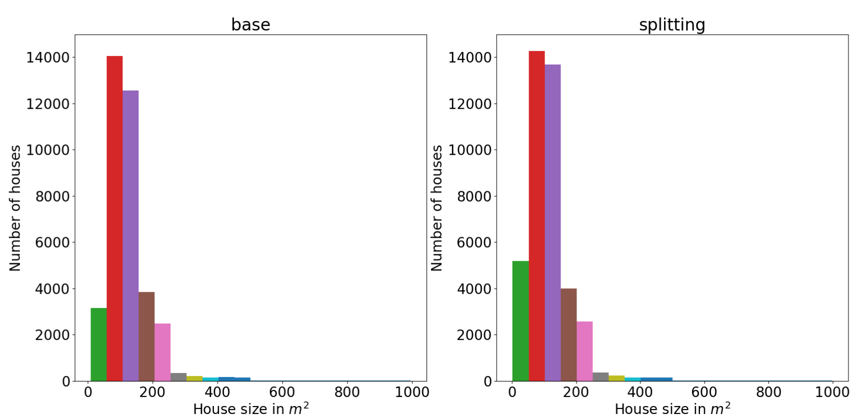

Figure C.46: House size histogram for splitting policy

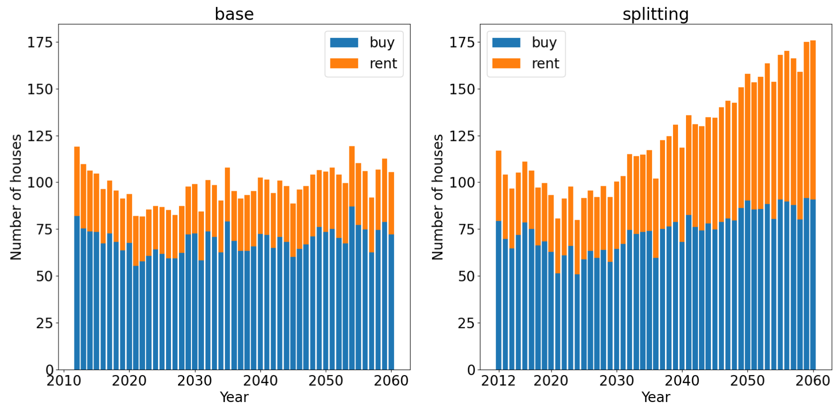

Figure C.47: Number of listed houses for splitting policy

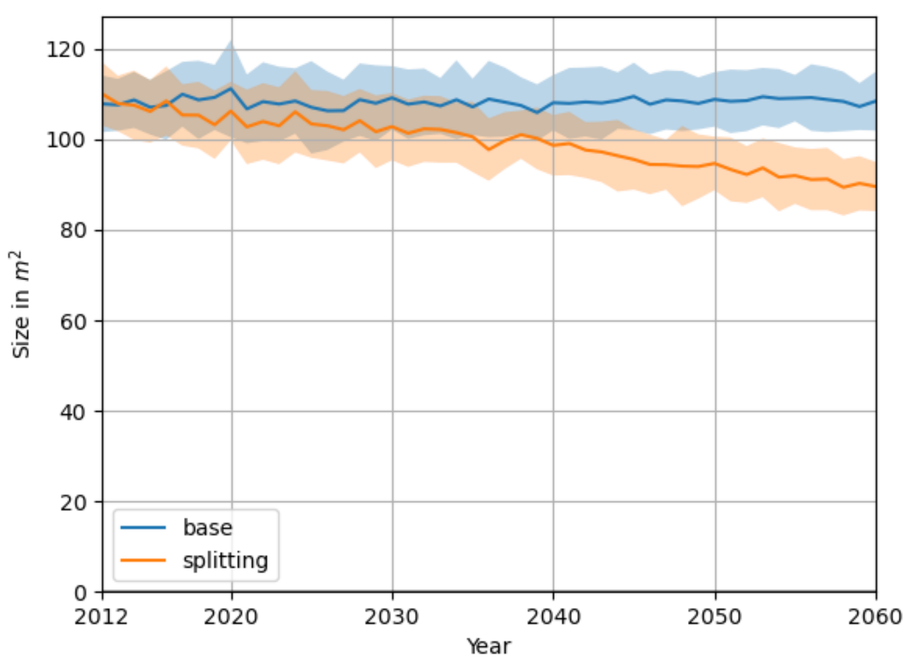

Figure C.48: Average size of listed houses for splitting policy

Rewarding splitting

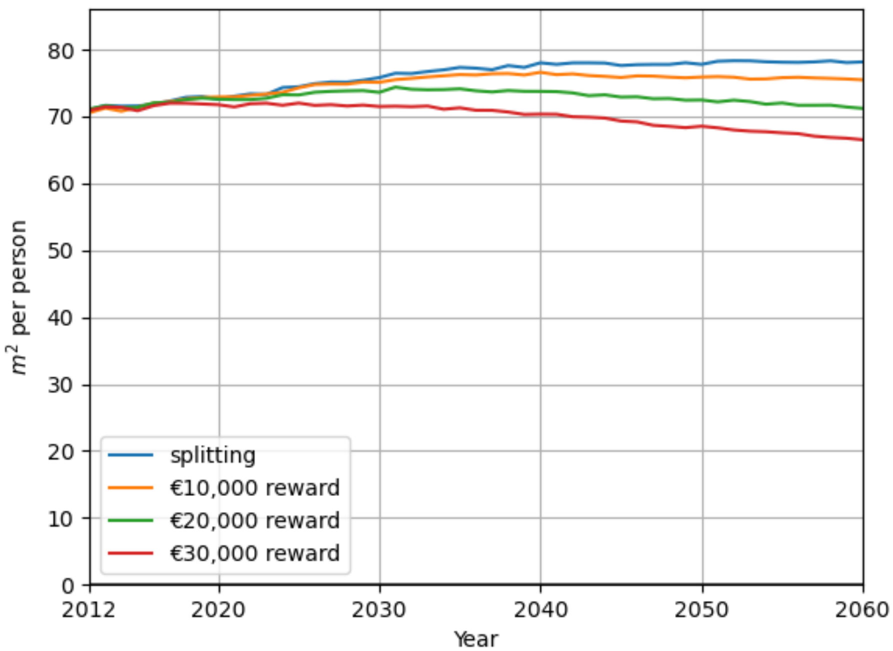

Figure C.49: Average space per person for rewarding splitting policy

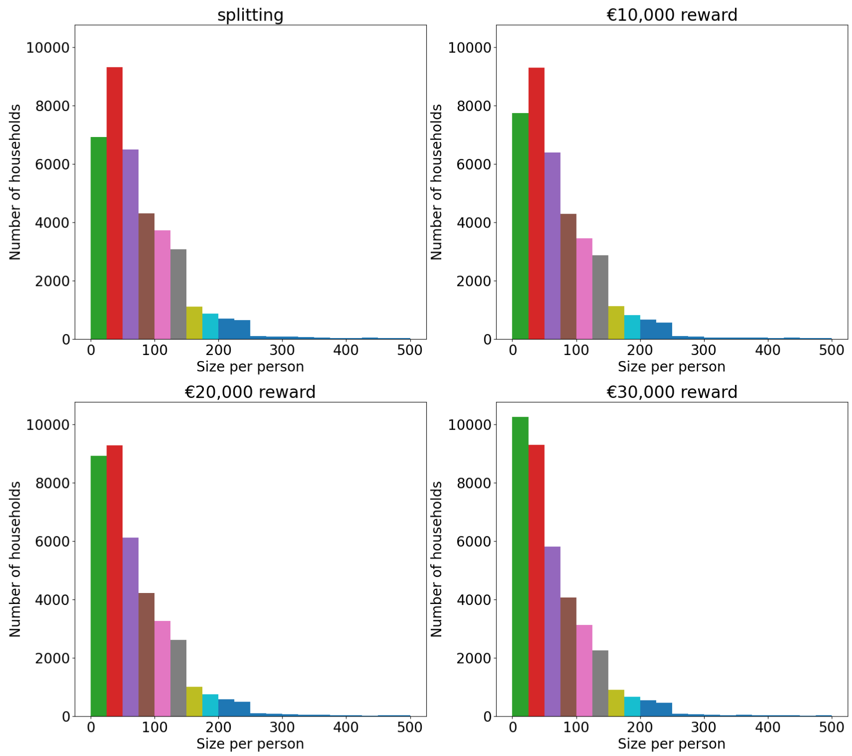

Figure C.50: Space per person histogram for rewarding splitting policy

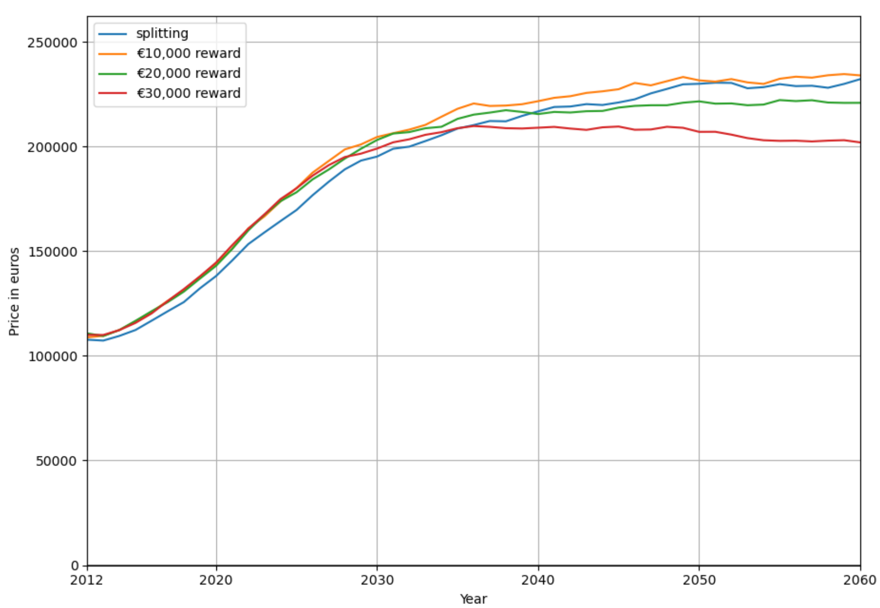

Figure C.51: Buying price for rewarding splitting policy

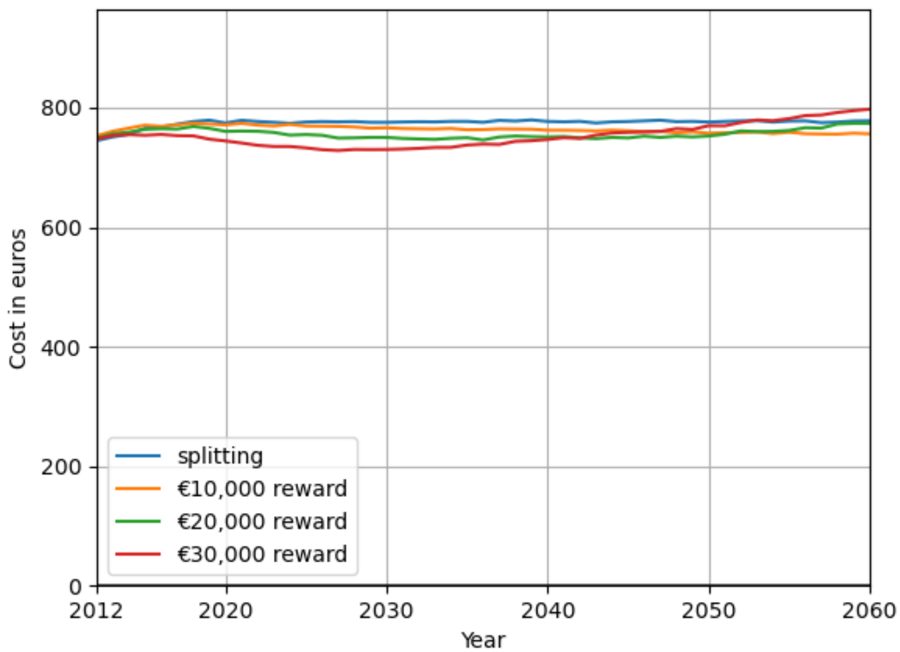

Figure C.52: Average rent for rewarding splitting policy

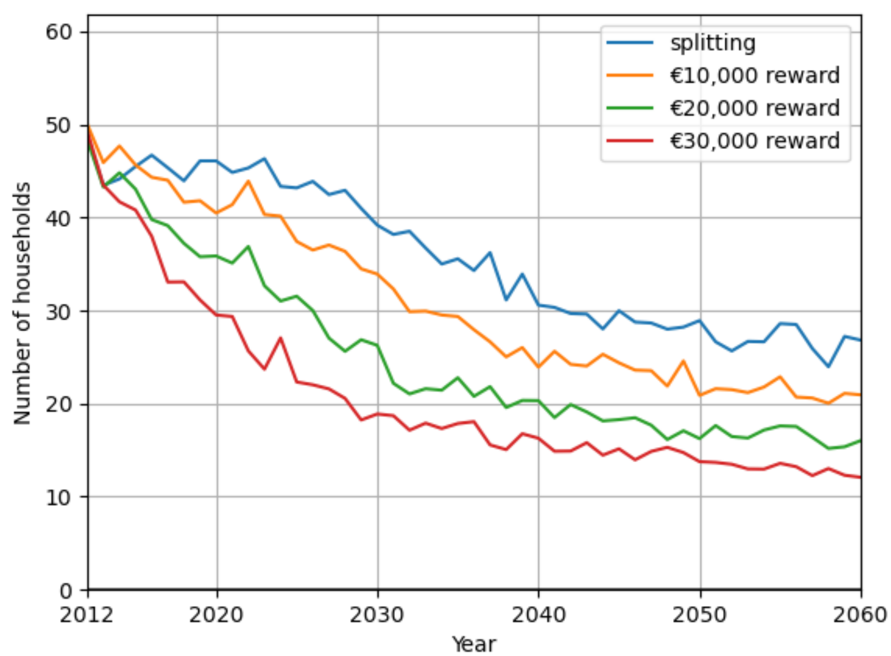

Figure C.53: Homelessness for rewarding splitting policy

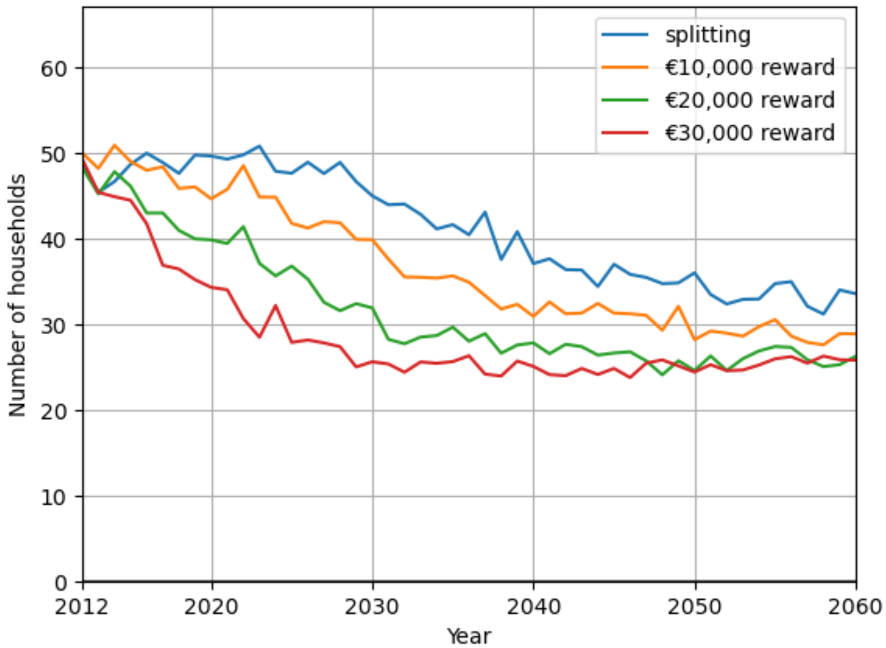

Figure C.54: Want to move count for rewarding splitting policy

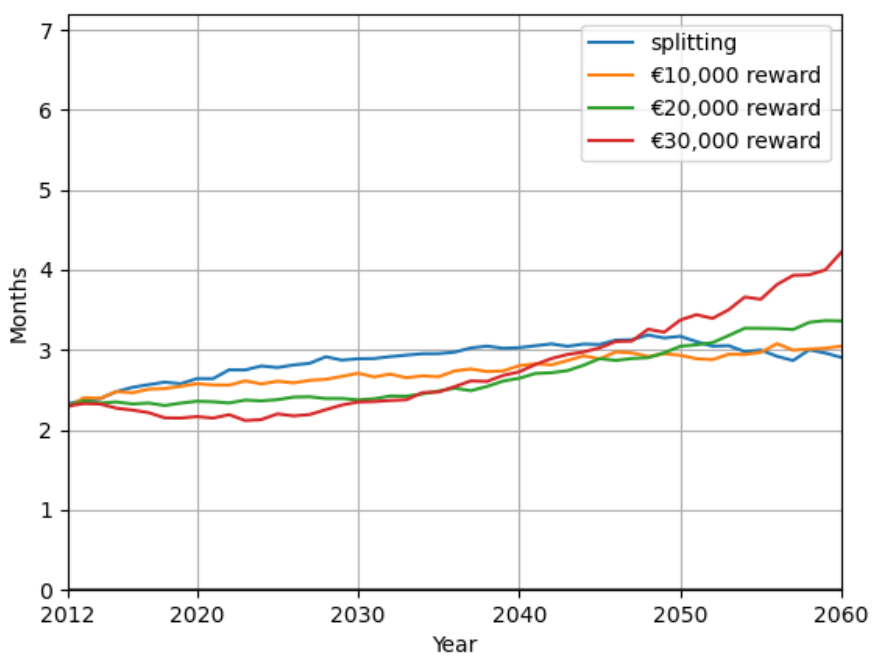

Figure C.55: Want to move time for rewarding splitting policy

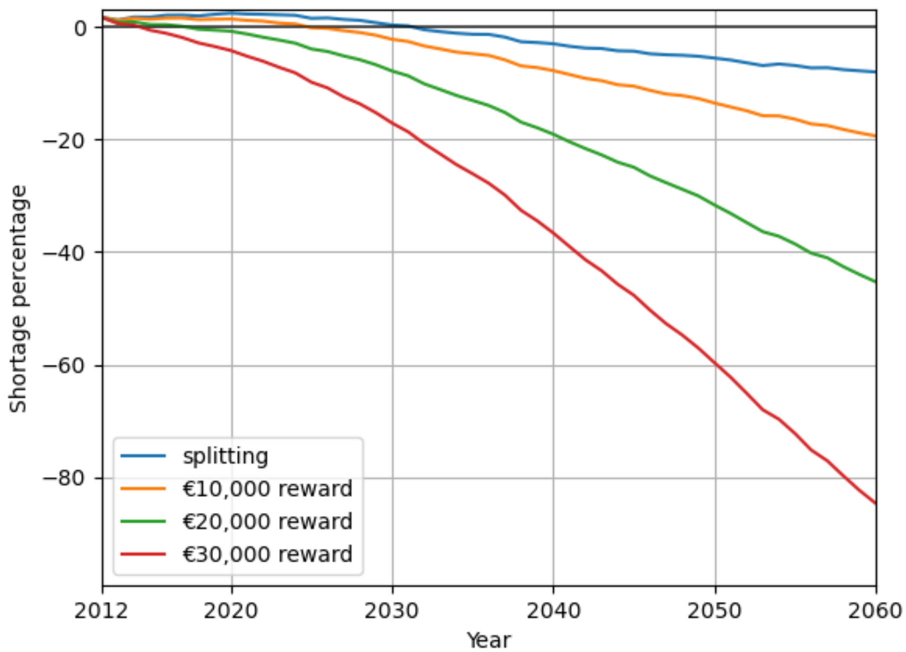

Figure C.56: Housing shortage for rewarding splitting policy

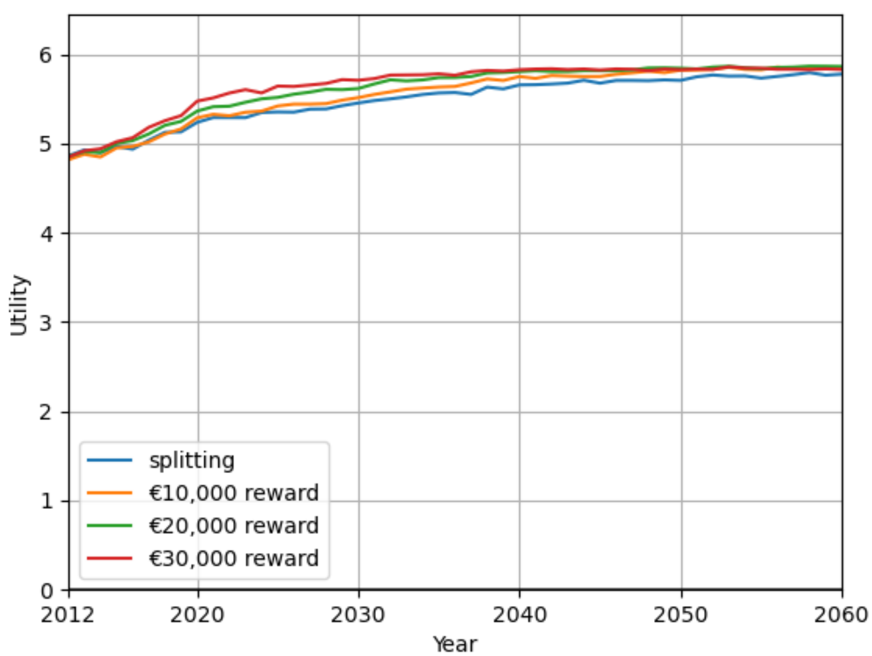

Figure C.57: Average utility for rewarding splitting policy

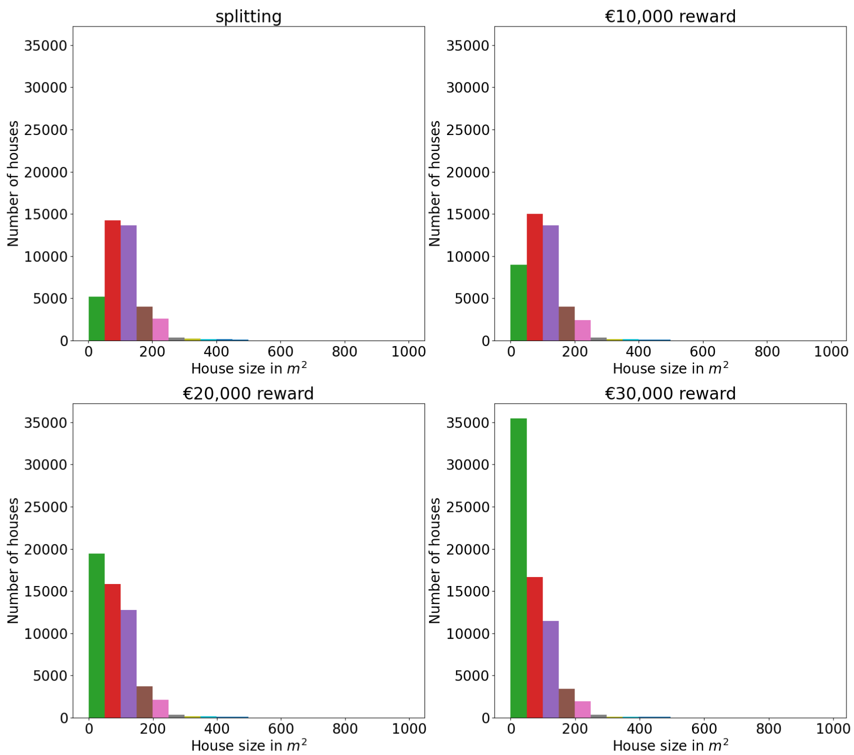

Figure C.58: House size histogram for rewarding splitting policy

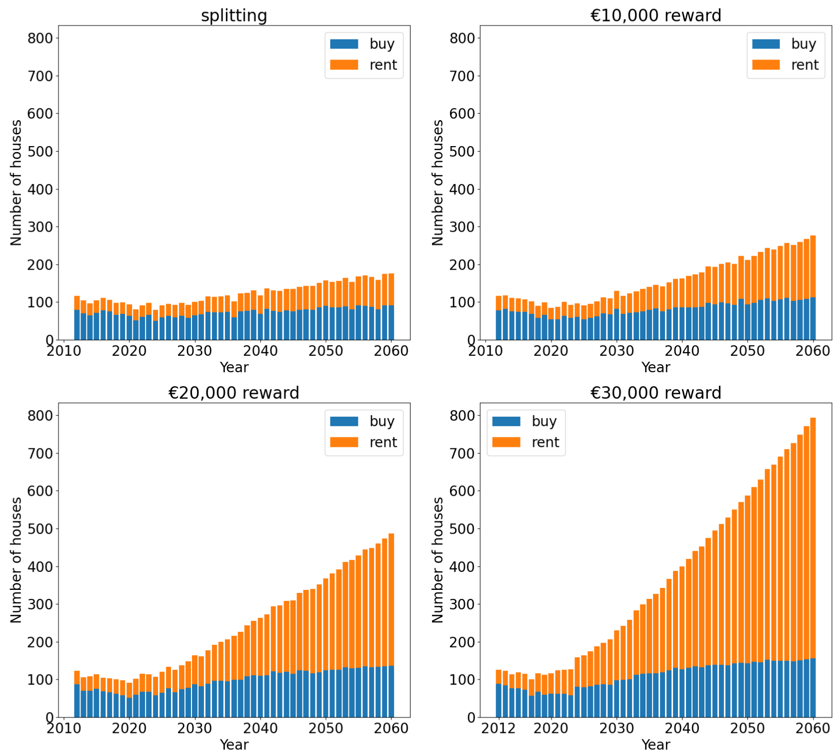

Figure C.59: Number of listed houses for rewarding splitting policy

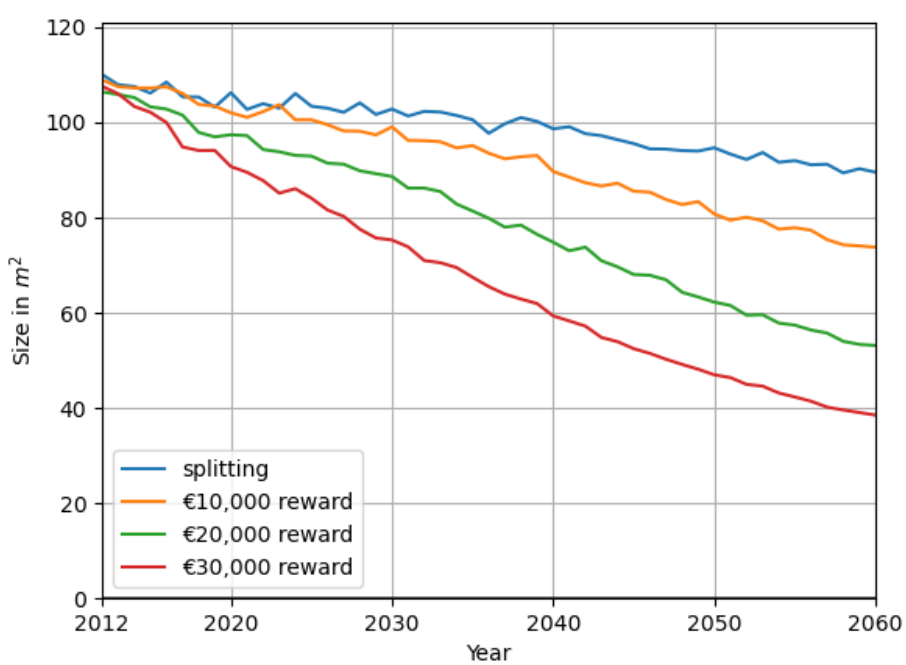

Figure C.60: Average size of listed houses for rewarding splitting policy

Split big listed houses

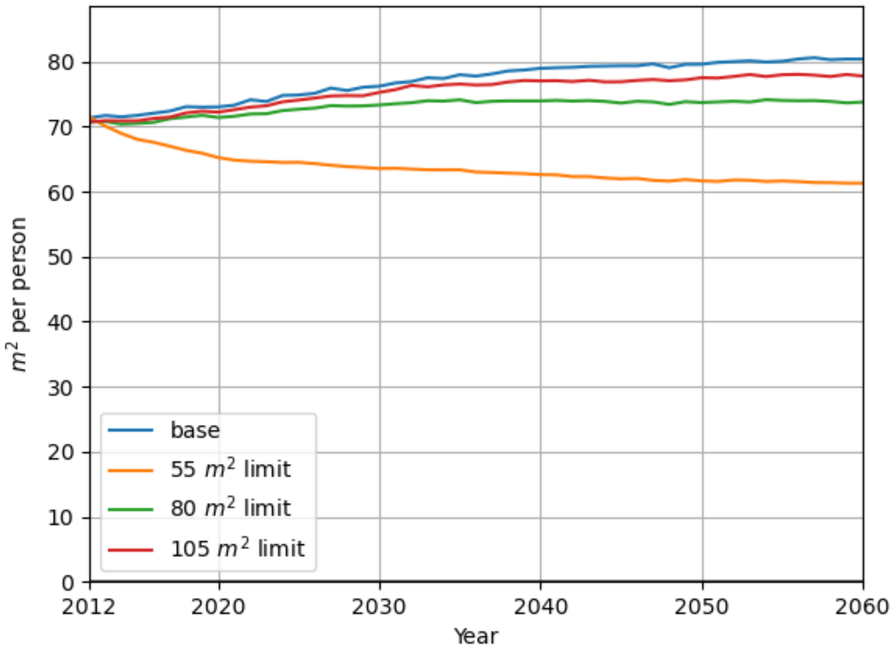

Figure C.61: Average space per person for splitting listed houses policy

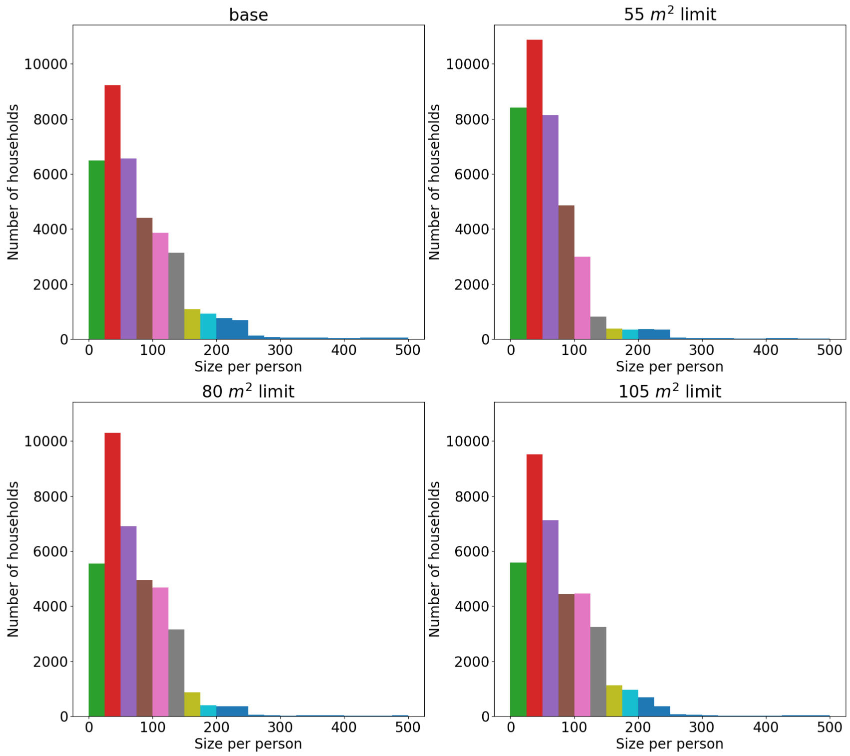

Figure C.62: Space per person histogram for splitting listed houses policy

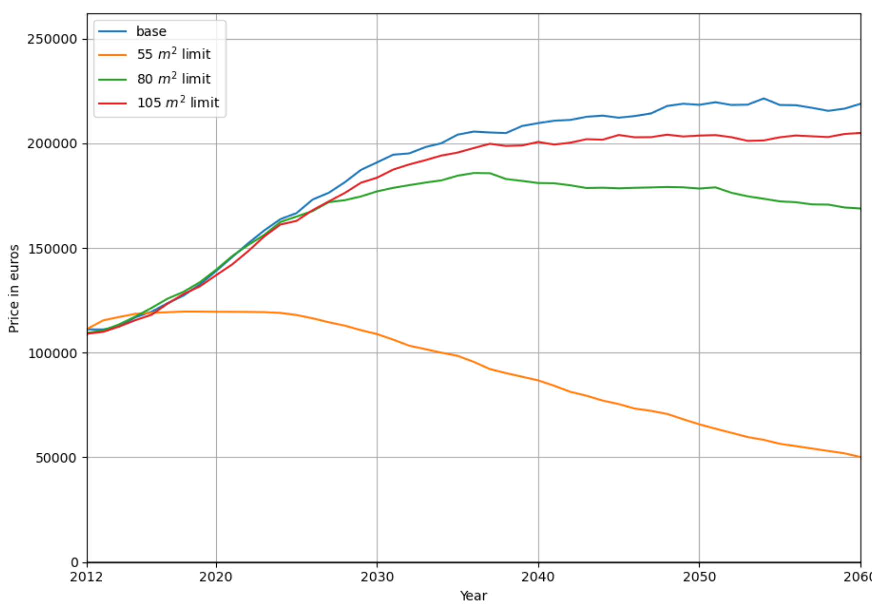

Figure C.63: Buying price for splitting listed houses policy

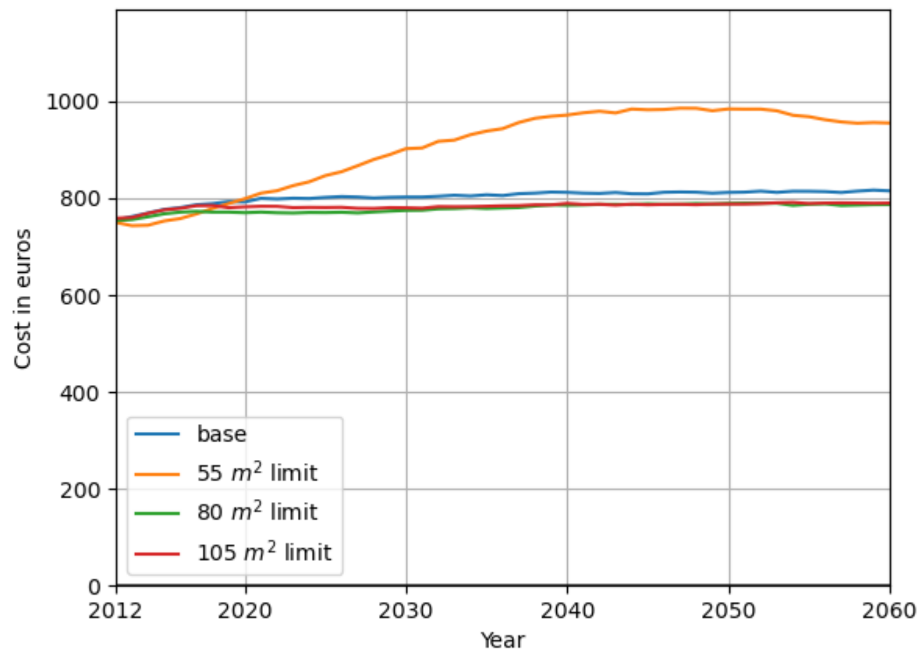

Figure C.64: Average rent for splitting listed houses policy

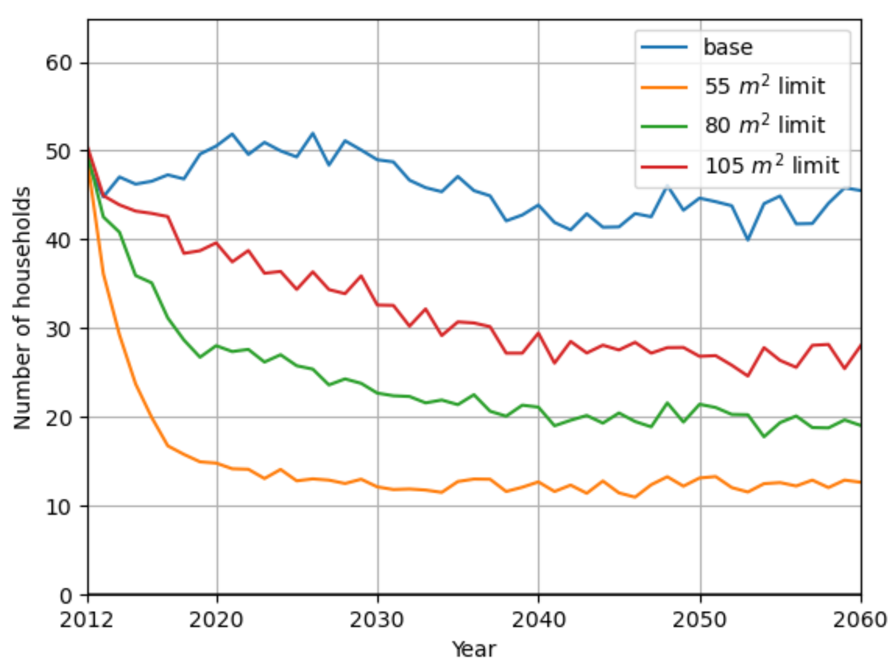

Figure C.65: Homelessness for splitting listed houses policy

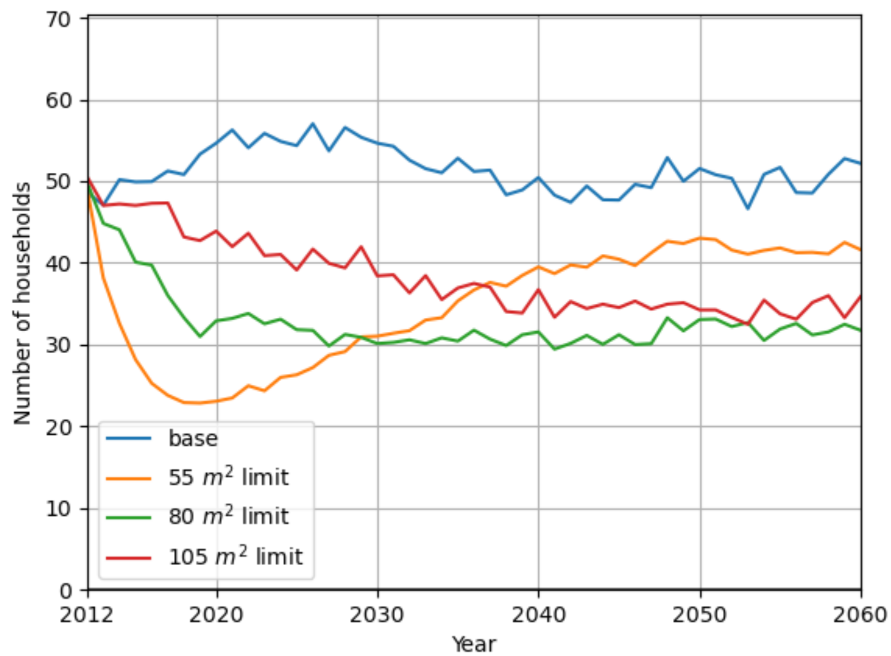

Figure C.66: Want to move count for splitting listed houses policy

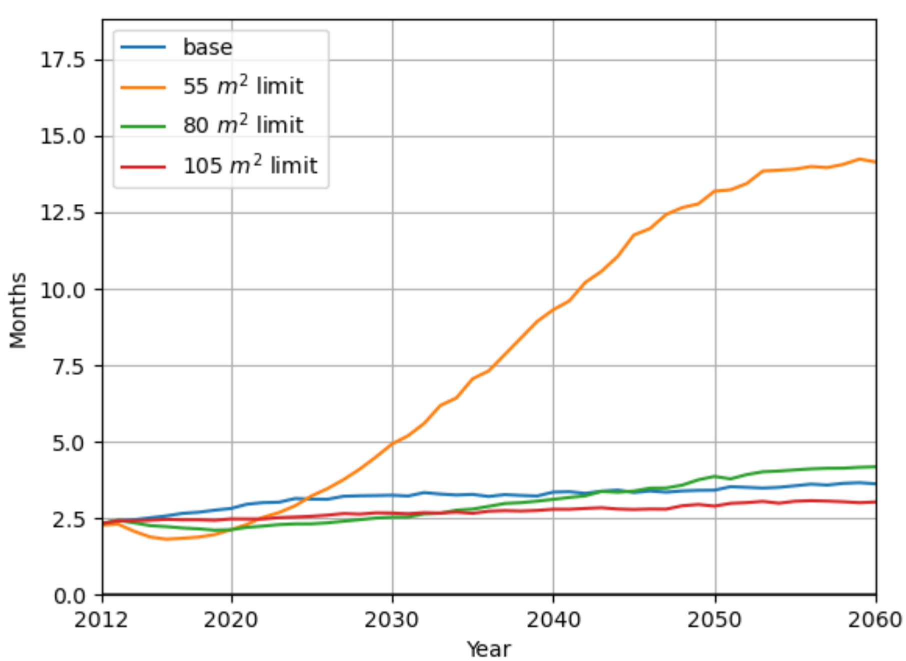

Figure C.67: Want to move time for splitting listed houses policy

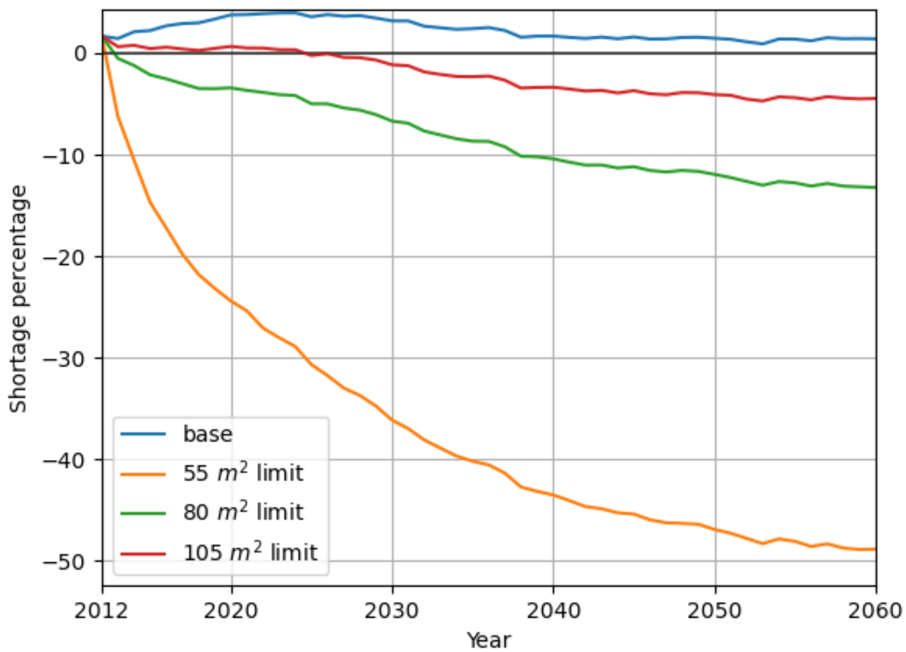

Figure C.68: Housing shortage for splitting listed houses policy

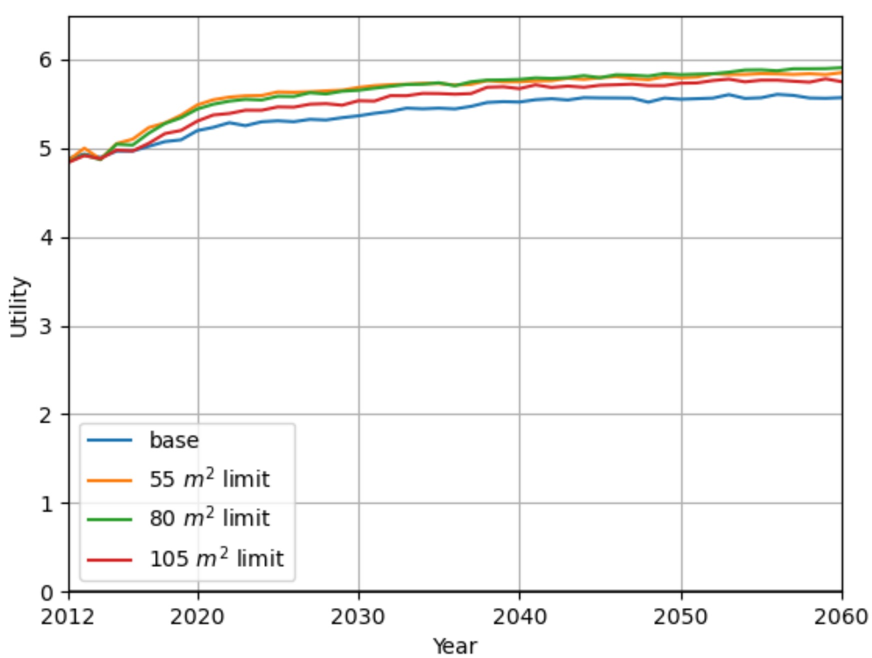

Figure C.69: Average utility for splitting listed houses policy

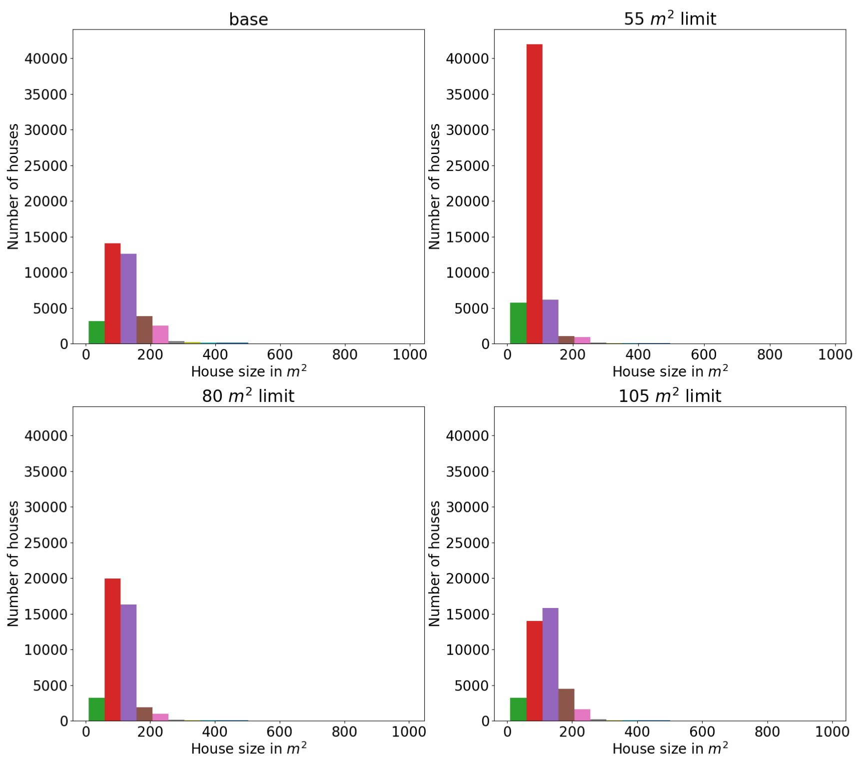

Figure C.70: House size histogram for splitting listed houses policy

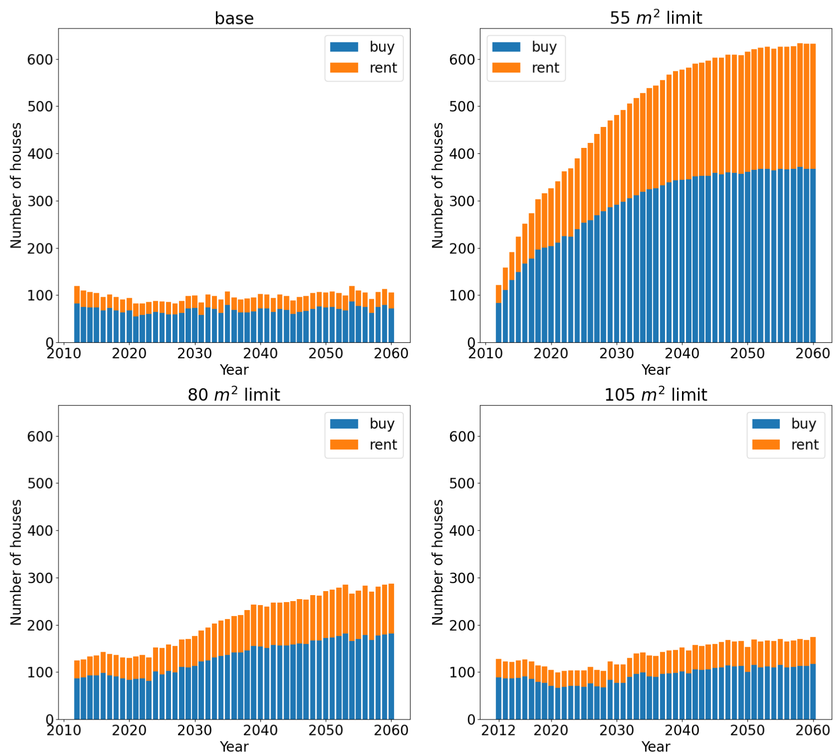

Figure C.71: Number of listed houses for splitting listed houses policy

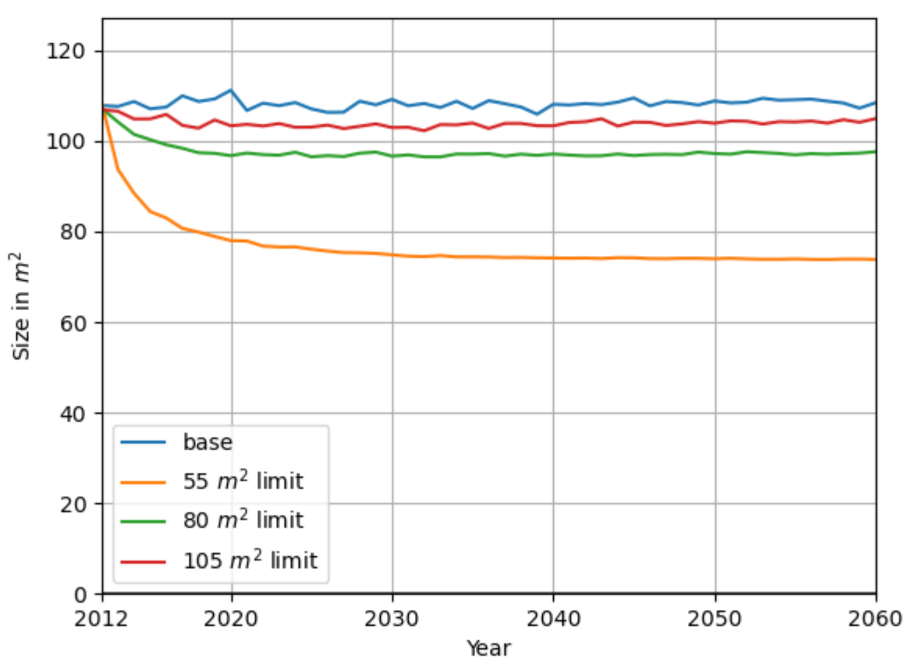

Figure C.72: Average size of listed houses for splitting listed houses policy