An individualized Tontine pension on the blockchain

- A blockchain enabled way of sharing longevity risk -

- PUBLIC VERSION -

UNIVERSITY OF TWENTE.

JULY 4, 2018

Author
Joost Muis
Student number
s1620878

Supervisors
A.C.M. de Bakker (UT)
B. Roorda (UT)
H. Terpoorten (APG)

Abstract

This research project is carried out on behalf of the GroeiFabriek which is part of APG Group N.V., from now to be called APG. APG is the largest pension provider in the Netherlands. The pension of one in five households in the Netherlands is managed by APG. The GroeiFabriek focuses mainly on innovation by providing an environment of continuous innovation. This research is part of a GroeiFabriek experiment. The experiment is a collaboration between APG, PGGM and Ortec Finance.

The research question of this research is: *"How can longevity risk be shared on the blockchain among pension fund participants using the Tontine principle?"*. In order to answer this question a blockchain infrastructure that enables participants to share longevity risk is developed. The way of sharing longevity risk is inspired by the Tontine principle, because it is transparent and intuitive. This way of sharing longevity risk gives the possibility to gain insights in managing an individualized pension system, which is relevant given the current discussions about the "doorsneesystematiek" used in the Dutch pension system. Sharing longevity risk on the blockchain in a peer-to-peer network can offer an old age pension to everyone with an internet connection. The benefit of using something as transparent and immutable as the blockchain is that it enables an old age pension to be managed in a trust less environment. This trust less environment enables reliable pension products not only in the Netherlands but also in countries where there is a lack of trust in the government and/or pension providers.

This research consists of two parts, an actuarial part in which the logic of the Tontine pension managed on the blockchain is explored and developed and a blockchain part in which the logic defined in the actuarial part is translated to a blockchain application that manages pension fund participants.

In the actuarial part a Tontine pension fund is modeled in which pension fund participants can save for their old age pension, where the premium is invested into assets based on their life-cycle, where the investments are traceable to the individual level. In this pension product, pension fund participants receive mortality gains when other participants decease, as the remaining balance of the deceased is allocated to the living participants. The pension fund participants start receiving variable periodical payments at the age of retirement based on an interest term-structure and their survival rate. The mortality gains are allocated according to the risk premium method which is logically deduced later in this paper. To minimize risk subsidization, this Tontine pension can manage different solidarity groups with different life expectancies by adjusting the death probabilities used in the computations for the annuity payments and the allocation of mortality gains. It is important to have a minimum number of pension fund participants in order to receive a stable benefit from mortality gains.

The actuarial logic found in the first part of this study is used to create a blockchain application on the Ethereum blockchain. It is possible to manage pension fund participants in a Tontine pension on the blockchain. If the Tontine pension product is managed on a public blockchain it is important to cut certain calculations into smaller pieces, otherwise the calculations would exceed the maximum allowed computational power in a single transaction. If one chooses to deploy a Tontine pension product on the public Ethereum blockchain, it is important to consider that the costs of managing such a pension product would be highly correlated with the Ether price, making the costs very volatile. Managing a Tontine pension on the blockchain enables everyone with an internet connection to have access to a reliable, transparent and fair pension product.

Keywords: Individualized pension, Tontine pension, longevity risk, solidarity, fair mortality gains, variable annuity, blockchain, Ethereum.

Preface

This thesis is the result of my research project conducted at APG on an individualized Tontine pension managed on the blockchain to conclude the master Industrial Engineering and Management with the specialization Financial Engineering and Management at the University of Twente.

This document gives a basic explanation on pensions and blockchain, making it readable for a broad audience however a financial and/or computer science background does help.

A special thanks goes out to APG for giving me the opportunity to work on such an interesting research project, also I would like to thank my boss for supporting me and facilitating the pleasant working conditions during this research, also a special thanks to my colleagues in Heerlen for giving me such a warm welcome and letting me be part of their team.

I would like to thank my friends and family for their love and support, my girlfriend for reminding me that evenings and weekends are not meant for working. Finally, a special thanks goes out to my grandfather whose personality, knowledge and experience has been a continuous source of inspiration even till long after his passing.

I hope you will enjoy reading this document as much as I enjoyed writing it.

Joost Muis

Amsterdam, July 4, 2018.

Table of Contents

List of Tables

Table **Page**

List of Figures

Chapter 1

Introduction

This research project is carried out on behalf of the GroeiFabriek which is part of APG Group N.V., from now to be called APG. This chapter describes the project, the project's background, the problem statement, the methodology and elaborates on the research project.

1.1 Background

APG is a financial service provider that facilitates executive consultancy, asset management, pension administration, pension communication and employer services. The pension of one in five households in the Netherlands is managed by APG. APG manages approximately 467 billion euros (October 2017) in pension assets [20].
The GroeiFabriek is part of APG and mainly focuses on innovation. The GroeiFabriek has four main activities [26]:

1. Stimulate the growth of new ideas, from within the APG organization, through knowledge sessions with partners and by following (technological) developments.

2. Experiment without limits, following a structured approach by performing research and applying solutions.

3. Building an ecosystem, an environment of continuous innovation.

4. Bringing ideas to market by launching them or by handing them over to other business units.

Currently the GroeiFabriek is working on many projects in the fields of artificial intelligence, sustainability and blockchain technology. In collaboration with PGGM, APG is experimenting with building a pension infrastructure using blockchain technology [1]. This research is part of an APG experiment (in collaboration with PGGM and Ortec Finance). In this experiment a team consisting of actuaries, developers, consultants and students will try to develop a blockchain infrastructure that enables participants to share longevity risk. In this experiment, it was decided that the way of sharing longevity risk on this blockchain infrastructure will be done by a so called Tontine principle, because it is transparent and intuitive. Studying a Tontine pension also provides the possibility to gain insights in managing an individualized pension system which is relevant given the current discussions about the "doorsneesystematiek" used in the Dutch pension system [11]. The principle of a Tontine pension will be further elaborated in section 2.1.8.

1.2 Problem statement

Managing a pension fund is something complex and it is still unclear how to translate this to a blockchain application. Since managing a pension fund is so diverse, it is beyond the scope of this research to elaborate how all aspects of managing a pension fund can be managed using blockchain technology. This paper only focuses on how the dynamics of sharing longevity risk within a pension pool can be managed using blockchain technology. More precisely, how longevity risk can be shared using the programmable logic of a smart contract running on an Ethereum Virtual Machine on an Ethereum blockchain, where the way of sharing longevity risk is inspired by the Tontine principle.

1.3 Research objectives

This study has the objective of providing insights on how to share longevity risk among pension fund participants using blockchain technology. More specifically, how longevity risk can be shared using the programmable logic of a smart contract on a Ethereum blockchain where the way of sharing longevity risk is similar to the way of longevity risk sharing in a Tontine. This study should give insights in the technical feasibility of sharing longevity risk on the blockchain given the limitations of the current blockchain technology, give insights in the required logic of sharing longevity risk in a Tontine inspired pension product, provide insights in the advantages and disadvantages of using blockchain technology given its decentralized nature, provide insights in the benefits of using the concept of a smart contract for longevity risk sharing and should provide insights in the benefit payments retired pension fund participants will receive given such a way of longevity risk sharing. The focus of this study will lie on the Dutch pension sector. This study will mainly focus on the technical and actuarial aspects and will keep legislation/legal perspective out of scope.

1.4 Research questions

Main question

How can longevity risk be shared on the blockchain among pension fund participants using the Tontine principle?

Subquestions

1. How can longevity risk be shared in a fair way in a pension fund using the Tontine principle?
2. How does the number of pension fund participants in the Tontine pension fund influence the benefit payments of its participants?
3. How volatile are the benefit payments in the Tontine pension fund given different economic scenarios?
4. How does the life expectancy influence benefit payments of its participants, in both homogeneous and non-homogeneous populations?
5. Is it possible to share longevity risk in a Tontine pension fund managed on an Ethereum blockchain?
6. What are the advantages and disadvantages of sharing longevity risk in a Tontine pension fund managed on an Ethereum blockchain given blockchains decentralized nature?

1.5 Research methodology

This study starts with a literature review on the pension dynamics and blockchain dynamics that are relevant to this study. There isn't much literature available on pension applications using blockchain technology, therefore no section will be dedicated in the literature review on blockchain technology in combination with pension management. In order to find out if it is possible to share longevity risk among pension fund participants in a pension fund using the Tontine principle, a multidisciplinary team of actuarial field experts, programmers, consultants and students work together in an experiment. The research is performed as in the build-measure-learn feedback loop as shown in figure 1.1.
First, the idea of a pension fund in which pension fund participants share longevity risk according to the Tontine principle on the blockchain will be further thought out. A functional design of the Tontine pension will be made, the scope will be determined of what should be handled on the blockchain and what aspects should be handled outside the blockchain. Then a simulation model will be build taken the previous obtained insights into account in which the behavior of a Tontine pension fund is modeled. This simulation model can provide data on how such a Tontine pension

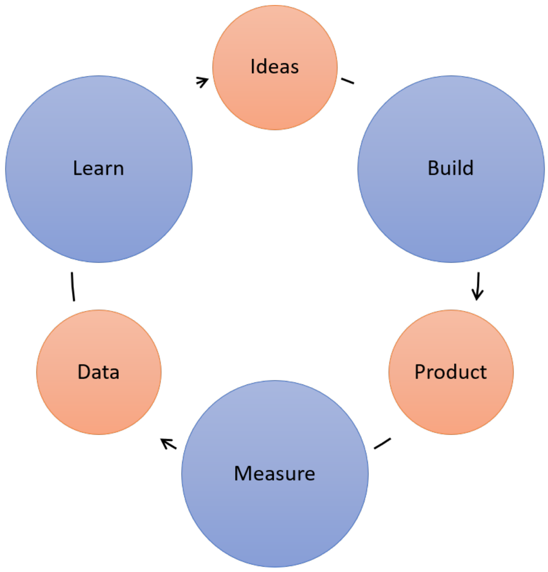

Figure 1.1: Lean startup build-measure-learn feedback loop [35].

fund would behave. This data can be used for actuarial analysis, so that new insights can be obtained on the proposed model so that the learnings can be used to further develop the Tontine pension product. Taken the learnings gained from the actuarial analysis based on the simulated data into account, a Tontine pension product will be developed on the Ethereum blockchain, which in turn will be used to generate data that can be analyzed to further develop the Tontine pension product and to test the feasibility of sharing longevity risk in a Tontine pension fund on the blockchain.

1.6 Outline

This research consists of multiple parts. At first a literature review is done in which relevant information about the Dutch pension system is given and information about blockchain technology and some of its underlying concepts. The insights gained from the literature study are then used to scope and define a simulation model that contains the logic required for a Tontine pension managed on the blockchain. The simulation model is in turn used to do a feasibility study to simulate how the Tontine pension behaves under different circumstances so that these behaviors can be analyzed. The logic as defined in the simulation model will be translated into a blockchain application, which in turn will be tested and evaluated. The insights gained from the simulation model, the feasibility study and the blockchain application will be used to answer the subquestions and the main question in the conclusion. Finally the study and the subjects for future research will be discussed.

Chapter 2

Literature review

This chapter briefly describes the Dutch pension system, the most commonly used pension arrangements in the Netherlands, the role of the asset manager, how important mortality rates are for a pension fund and elaborates on how longevity risk can be shared. It will give a brief introduction to blockchain technology and will explain why it is a strong concept. There is still very little literature available on pension applications using blockchain technology, therefore no section will be dedicated to pension management using blockchain technology in this chapter.

2.1 Pension management

2.1.1 Structure Dutch pension system

The Dutch pension system consists of three pillars [8]. The "algemene ouderdomswet" (AOW) which is a Dutch Law that supplies retirees with a basis income once they retire, the height of this is adjusted yearly based on the development of the minimum wages. By working or living in the Netherlands you build this right automatically.

The second pillar, which is the main focus of this study, is the pension built via the employer. In the Netherlands about 90 percent of the employers use this pension arrangement in addition to the AOW. This pension arrangement is usually built by paying a pension fund, which will in turn invest the paid premiums so that it can payout the pension of the pension fund participant.

The third pillar consists of individual additional pension arrangements e.g. life insurance, annuities or extra premiums, which can be used to for example close a pension gap or to retire early.

2.1.2 Solidarity

The key word in the Dutch pension system is solidarity [36]. In the Netherlands a so called "doorsneepremie" is mostly used for supplementary old-age pensions [22], this means that everyone gets the same amount of pension rights for their paid premium, regardless of difference in life expectancy. For example a woman gets the same pension rights as a man for the same amount of paid premium whilst on average women live longer than men. This means the system knows both risk solidarity, by sharing different risks within a pension pool, such as longevity risk, as subsidizing solidarity meaning that in some cases no effort is done to match the participant's risk to the paid premium.

2.1.3 Premium arrangements

There are certain pension arrangements that have an influence on the amount of premium a pension fund participant has to pay and how much retirement benefit the pension fund participant will receive. The Dutch market comprises both defined benefit arrangements and defined contribution arrangements, later to be called DB and DC arrangements. It is also possible to have hybrid forms [17].

DC arrangement

The notion DC arrangement is pretty self explanatory. It means that the pension fund participant pays premium, but has no guarantees on the height of the retirement benefit. The height of the retirement benefit is determined at the moment of retirement, where actuarial methods are used to calculate the height of the benefit payments based on the paid premiums and the investment returns the pension fund has made investing the pension fund participant's money [28]. This means that the risks mainly lies at the individual pension fund participant, however this pension fund participant can also benefit more from having better investment returns. Traditionally a pension fund participant would receive a constant benefit at the age of retirement, however since the introduction of the improved premium arrangement [18], the pension fund participant can choose at the time of retirement to keep money invested, this in turns means that the received retirement benefit will be variable based on investment returns. It is often chosen to combine a DC arrangement with insurances against biometric risk e.g. passing away or incapacity for work.

DB arrangement

In a DB arrangement, the pension fund participants get a certain pension claim in return for their premium payments. In contrast to the DC arrangement, the premium payments in the DB arrangement are variable. The aim is to build a certain percentage of pension each year so that the pension fund participant can enjoy its pension at the age of retirement. In a DB arrangement the height of the benefit payments is secured as well as possible. Meaning that the risks of investing these paid premiums are not carried by the individual. So in contrast with the DC arrangement, the retirement benefits are not dependent on the investment returns of the individual, but are dependent on the investment returns of the solidarity group as a whole. Making the height of the benefit payments dependent on the investment returns of the solidarity group is done with the use of a coverage ratio. When the coverage ratio of the pension fund gets below a certain threshold, the benefit payments of the entire group of pension fund participants can be reduced or the pension fund can be obliged to not to apply indexation for inflation.

2.1.4 Asset management

The asset manager of a pension fund invests the money received from the pension fund participants. In a DC arrangement the amount of money invested in for example equity and bonds is dependent on a pension fund participant's life-cycle and sometimes investment preference. When a pension fund participant gets older, the amount of money invested in equity usually reduces whilst the amount of money invested in bonds increases in order to carry lower risks. In case of a DC arrangement the investment returns reflect in a change in pension payment at the time of retirement whilst pension members having a DB arrangement don't carry this risk, except for when the pension fund is below a certain coverage ratio where the benefit payments of the entire solidarity group are reduced.

2.1.5 Mortalities

Different people will die at different ages. Pension fund participants are no exception to this. Whilst some people die very young and haven't had time to enjoy their pension, some people become very old and enjoy more benefit than they actually put into their pension account. Pension funds typically have a positive result on people with a short life span and a negative result on people with a long life span. Data on the life span of pension fund participants plays a key roll for a pension fund because it helps a pension fund to determine the required premium payments and it helps the pension funds to determine the periodic payments to retirees. Coping with these mortalities is one of the key roles of a pension provider, but doing this efficiently is dependent on a wide range of factors [23].

AG table

A party that models the Dutch mortality rates based on the Dutch historical mortalities and mortalities in countries that have wealth which is similar to the Netherlands, is the Koninklijk Actuarieel Genootschap [16], later to be called AG. The AG takes it upon themselves to supply the financial industry with insights on the development of mortality rates. An example of data which is used by pension funds in the Netherlands is the AG table. This table holds the mortality rates for males and females for different ages for different years. These rates are used in for example annuity computations.

Select mortality

The AG table treats the population as a homogeneous group. The mortality rate is only obtained for a certain year by gender and age. This may be a good assumption for the totality of the group, however this doesn't have to hold for subsets of the group [33]. As a pension provider, the mortality rates of these subsets can be interesting, since pension benefits are dependent on mortality rates and pension providers try to allocate the pension benefits as fair as possible and prevent having too many positive or negative results on people that decease.

In Tontine based products, select mortality becomes important, since if one would open a Tontine product up for everyone, there is an incentive for people with a higher life expectancy to join with people in a Tontine that have a lower life expectancy, because in this situation the people with a higher life expectancy expect to receive more money. These differences in life expectancy can be handled in a Tontine pension fund by using for example the fair mortality gains method or the risk premium method described in section 2.1.8 and section 4.1.1 respectively, so that in general people with a higher life expectancy don't benefit more from their invested money, than people with a lower life expectancy.

2.1.6 Asset liability management

Asset Liability Management, later to be called ALM, is an important tool for determining investment strategies, indexation and contribution structure for a pension fund. ALM is an important instrument for determining the measure of indexation for DB pension rights. The coverage ratio is the relationship between the net present value of assets and liabilities. When the coverage ratio is below a certain threshold, the pension fund can be obliged not to apply indexation [27].

2.1.7 Longevity risk

Longevity risk is the risk of pension fund participants living longer than mortality tables would predict [33]. Because of this risk, it is important that annuities are priced the right way. One of the benefits of having a group of pension fund participants is that the longevity risks can be shared among the pension fund participants. In this case, by longevity risk the increase in the age of individuals is meant rather than that of the whole population of the solidarity group. In a solidarity group, pension fund participants that live a long time gain benefits from pension fund participants that have a shorter life span. This principle is called risk solidarity [36]. Basically when people decease, the surplus of that account is used to finance the pension fund participants that live longer than average. There are different ways how people that are contributing to a pension plan can handle this longevity risk. There are different ways of managing longevity risk, also if an individual doesn't participate in a pension plan at a pension provider.

Systematic withdrawals

One of the simplest way of managing retirement savings as an individual, is by using a systematic withdrawal plan (SWP) [25]. What someone does in this case, is invest in a diversified portfolio of for example 50 percent stocks and 50 percent bonds that is designed with a high probability that the retirement savings will last till 20 to 30 years after retirement. When retired, the retiree can withdraw money from this fund periodically.

Lifetime annuities

As an individual, one can invest in lifetime annuity bonds that can be bought for a certain amount and yield a percentage annuity for the remainder of someones life span. An alternative for this lifetime annuity bond is an inflation-adjusted lifetime annuity bond.

Longevity insurance

A way to protect an individual against longevity risk is by taking an insurance. A way of doing this is to buy a deferred annuity, which starts to payout at a later age. For example if someone would use systematic withdrawals to pay for his or her retirement, it is likely that if the retiree lives long enough, at a certain point in time the invested portfolio is reduced to zero. If the individual has invested in a lifetime deferred annuity, he or she will receive payments from a predetermined point in time till passing away.

Tontine

The Tontine principle is a principle where investors pay to an investment fund, where the payments are invested in for example a treasury bond which yields annual coupons. These coupon payments are paid and divided among the investors that are alive proportional to their invested amount. This principle combines the features of an annuity with a gamble. Each investor receives a lifetime annuity and when someone deceases, the coupon of that investor is divided among the living investors, resulting in an increase of coupon for the remaining living investors. This is repeated until all but one of the investors are dead, where the survivor receives all. This means there is no inflow of new participants to the Tontine and that the number of participants is finite. In 1868 Tontines were introduced as a combination of a life insurance and an old-age saving plan [34]. In 1905 about two-thirds of all life insurance in force were of this type, however in 1906 the sale of Tontines was prohibited. One could argue if such a Tontine insurance would incentify helping nature to become the last surviving investor. However, from an actuarial point of view a Tontine is a very sound and attractive life-cycle investment.

2.1.8 Tontine pensions

The Tontine principle is well suited to create a Tontine pension [25]. In a Tontine pension, participants join by paying an amount of money to the investment fund. This payment is then added to this person's account, the person in question can pay multiple premiums over time, which are in turn added to the person's account. The money on the account is invested by an investment manager based on the person's life-cycle and investment preference. Once this person reaches the age of retirement he or she will start to receive periodical payments based on their account balance and an annuity factor. When the person dies, the money that is still in his or her balance is allocated to the living pension fund participants proportional to their wealth, increasing their periodical payments.
This can be illustrated by a quantitative example:

	Balance	Alive	Mortality gains
Person A	100	1	0
Person B	150	1	0
Person C	300	1	0

Table 2.1: Tontine mortality gains example: before mortality

	Alive	Mortality gains	Balance after mortality gains
Person A	1	120	220
Person B	1	180	330
Person C	0	-	-

Table 2.2: Tontine mortality gains example: after mortality

This is however not a fair way of allocating the mortality gains to the living pension fund participants, since young pension fund participants are favored in this situation, since these young pension fund participants are more likely to receive a higher number of mortality gains than the elderly. Therefore it is more fair to use a different way of allocating mortality gains.

Fair mortality gains

As described in [25], there is a way to allocate mortality gains fairly in a Tontine pension to pension fund participants of all ages and genders. Namely, by computing the fair mortality gains. In a transfer plan where pension fund participants receive fair mortality gains, the expected return on the mortality gains of each pension fund participant equals zero. This can be broken down to the equation 2.1.

$$0 = -f_i s_i + \sum_{j \neq i} \frac{f_j s_j w_i}{(1 - w_j)}, \qquad \text{for each pension fund participant } i \tag{2.1}$$

In equation 2.1, f is the force of mortality probability[1], s is the pension balance and w is the fair transfer weight, i and j are referring to pension fund participant i and pension fund participant j. This yields a set of m non-linear equations equal to the number of pension fund participants that can be solved yielding an unique solution for weights w_i and w_j. For some quantitative examples see [25], on page 778-782.
These set of non-linear equations cannot be solved analytically, there are numerical algorithms to find a solution for weights w_i and w_j, however these algorithms require computing power. An alternative method for allocating mortality gains is the risk premium method, which is described in section 4.1.1.

2.1.9 Replacement rate

The replacement rate is a way of comparing benefit payments of pension fund participants in relation to their income. The replacement rate is the benefit payment paid to a retiree divided by their last earned salary. Even though there are some points of critique on using the replacement rate to measure pension adequacy, for example in cross-country analyses [21], these points of critique are not relevant for this research as the analyses are only done for Dutch pension fund participants and those in similar living conditions. In practice it may occur that a pension fund participant starts working less in the years before retirement, which can cause a decrease in salary. In the simulation model, pension fund participants do not start to work less in the years before retirement, making this factor irrelevant for this research. However when using the replacement rate to compare benefit payments of real pension fund participants, it might be interesting to look into how large the influence of the pension fund participants working less in the years before retirement is on the replacement rate and if this should be handled in a certain way.

2.2 Blockchain

In October 2008, the paper Bitcoin: A Peer-to-Peer Electronic Cash System was published under pseudonym Satoshi Nakamoto, describing a way of securely transferring electronic cash without having to rely on financial institutions [31]. Only a few months later the first Bitcoin software was launched. The technology that makes Bitcoin possible is commonly referred to as the blockchain. In its essence, a blockchain is a decentralized transparent ledger with transaction records, shared by all network nodes, updated by miners, monitored by everyone and controlled by no one [38] and where no funds can be double spend. This section briefly describes the fundamentals that underly blockchain technology. This description will not go into technical details because that is beyond the scope of this research.

[1] The force of mortality probability is the instantaneous probability of mortality given a certain age measured on an annualized basis. The force of mortality probability is identical in concept to a hazard function.

2.2.1 Cryptocurrency

Bitcoin is still the most known cryptocurrency and has the largest market capitalization [2]. Since the release of Bitcoin, cryptocurrencies and the technology behind them has improved. One example of this is Ethereum, which is the second largest coin in market capitalization [2]. Ethereum was first introduced in 2014 and was launched in 2015. One of the aspects that makes Ethereum so revolutionary is that it allows users to not only send transactions, but also enables them to send complex code structures with their transactions, making the behavior of the digital money programmable. This concept will be elaborated further in section 2.2.9. Today many more coins have been introduced, but Bitcoin and Ethereum are still the most commonly used currencies.

2.2.2 Decentralized ledger

The technology behind Bitcoin is commonly referred to as the blockchain. The blockchain is basically a chain of blocks, containing the hash value of the previous block, a time-stamp, some data, transactions and a nonce [32]. The concept of a hash in this context is explained in section 2.2.4. A nonce is an arbitrary integer which is added to the block to find a hash that meets certain conditions. New blocks are periodically added to this chain of blocks. All blocks added to the blockchain are traceable to the genesis block, which is the first block of the blockchain. This chain of blocks is basically a ledger containing all transactions done from the genesis block up until the last block. All participants on the blockchain have a copy of this ledger, have at all times access to the information in the ledger and every participant on the blockchain can check whether or not the information in it is correct [39]. There is no single owner of the blockchain and anyone who wants to participate can participate. Since every transaction is logged in the blockchain it is possible for every participant on the blockchain to compute all transactions done to or from a single account[2] since the genesis block and so validating whether or not that account has enough funds to do a new transaction and so prevent double spending. Before new transactions are added to the blockchain, the transactions are broadcasted to the network and validated, where the majority of the blockchain participants have to agree upon what the new ledger including the new transactions will be.

2.2.3 Double spending

Preventing double spending is a fundamental element of blockchain technology. This can be handled in different ways. For example Bitcoin works with unspent transaction output later to be called UTXO. Ethereum uses account balances rather than UTXOs. In Bitcoin only UTXOs can be spend in a transaction, meaning that spend transaction outputs cannot be spend again.

[2] It is important to note that transactions can be linked to ones public account, however that account is not directly linkable to a person.

These UTXOs can be determined by tracking all transactions done to and by a certain wallet[3]. If one wants to do a transaction, the sum of UTXOs of the entire wallet has to be spend, where the UTXOs can be allocated to different wallets. If Bitcoin wallet A for example has 10 UTXOs and wants to send 1 UTXO to Bitcoin wallet B, the owner of Bitcoin wallet A would send 1 UTXO to Bitcoin wallet B and 9 UTXOs to Bitcoin wallet A (in this case the change address) in a single transaction. One could compare this with spending a 10 euro bill at the supermarket, purchasing a product for 1 euro, one has to give the 10 euro bill, but receives 9 euro back in change while it is not possible to spend more money from ones wallet than that is put into it. The concept of UTXOs makes it possible to track all transactions back to the genesis block. The Ethereum blockchain works with an account-based system, which relies on a global state storage of accounts, balances, code and storage [3], where an account balance can do transactions as long as it has sufficient funds.

2.2.4 Hashing

One of the concepts that makes a cryptocurrency like Bitcoin so strong is hashing. Hashing as done in the blockchain ensures that no one can tamper with the transaction history, which makes the blockchain immutable[4]. For example Bitcoin uses a Secure Hashing Algorithm later to be called SHA, in order to confirm data integrity. This hashing algorithm has a few important properties [19]:

1. **Deterministic:** If the hash function is used on the same data multiple times, it will always yield the same result.

2. **Quick Computation:** The hash function is able to compute a hash based on an input very quickly.

3. **Pre-Image Resistance:** This means that the hash is one way. Based on an input it is very easy to compute a hash, however based on a hash it is difficult to determine an input.

This means that whenever the same data is used as input for the SHA function, this will always lead to the same output, however when the input is slightly changed, it is very difficult to predict what the new hash will be without computing it, the computation of a new hash is very quick, also it is very difficult to, from a SHA output, determine the original input. The SHA256 algorithm produces a hash that is 256 bits long, meaning that there are 2^{256} possible outputs. SHA256 doesn't have a one to one mapping, allowing for many more possible inputs than outputs. The

[3]A wallet, like for example a Bitcoin wallet is an address which is used to send coins to and from. One has a private key which can be used to sign transactions and which should only be known by the owner of the wallet and a public key generated from the private key which is publicly available so that everyone can see what transactions have been done to and from a certain address to verify if the address has sufficient coins for a certain transaction.

[4]This assumption only holds when more than half of the computational power in the network is used for just computations.

SHA256 function may be used to hash a message having a length of l bits, where $0 \le l \le 2^{64}$ [5]. This means that multiple input values can lead to the same output hash. It is called a hash collision when the same inputs to a hashing function result in the same hash. Trying to find a specific hash that corresponds to an output created by the SHA256 function using a brute force algorithm, would mean that at most all input messages of length l bits in the search space $0 \le l \le 2^{64}$ have to be checked, which are too many possibilities to be checked for being feasible using current days technology. Finding an input that corresponds to a specific output hash, does not guarantee that the input of the SHA256 function was the same as that of the found input, or that by coincidence different inputs of the SHA256 function led to a hash collision. If one would use birthday attacks, that exploits the mathematics behind the birthday problem in probability theory[5], the computational complexity of finding a hash collision can be reduced to $O(2^{n/2})$, where n is the length of the hash output in bits. This would mean that a hash collision is expected to be found checking 2^{128} hashes [37].

How different the output hashes are with just a small change in its input can be easily demonstrated with an example. If one would use the input *Hello world* or *Hello world!* for the SHA256 [6] function it yields completely different hexadecimals. For example:

$$SHA256(\text{Hello world}) = 64ec88ca00b268e5ba1a35678a1b5316d212f4f366b2477232534a8aeca37f3c$$
$$SHA256(\text{Hello world!}) = c0535e4be2b79ffd93291305436bf889314e4a3faec05ecffcbb7df31ad9e51a$$

If one would give the the hexadecimal c0535e4be2b79ffd93291305436bf889314e4a3faec05ecffcbb7df31ad9e51a it is very difficult to guess its original input, without knowing that its input was *Hello world!*. Now lets translate this example to a blockchain example. As explained in 2.2.2, a block in the blockchain contains a link to the hash of the previous block, the transactions from certain accounts to certain accounts, the transfered amounts, a time-stamp, some data and a nonce. For this example, a block will only contain one transaction and no time-stamp[7].

[5]The birthday problem can easily be explained through an example. If one has a room with 23 people. If one tries to find a specific date on which at least one person has a birthday, the probability of guessing such a date correct is $1-(\frac{364}{365})^{23} \approx 0.0612$ (ignoring leap years), while the probability that one or more people share the same birthday in this room is $1-\frac{365!}{(365-23)! \cdot 365^{23}} \approx 0.5073$, which is significantly higher. This can be translated to finding hashing collisions, where the birthdays of the people in the room reflect the generated output hashes and the days in a year reflect the finite number of possible output hashes [6].

[6]SHA256 means that the hash value consists of 256 bits

[7]From hereon forward only the first 10 characters of the hexadecimals are shown as for the example in image 2.1.

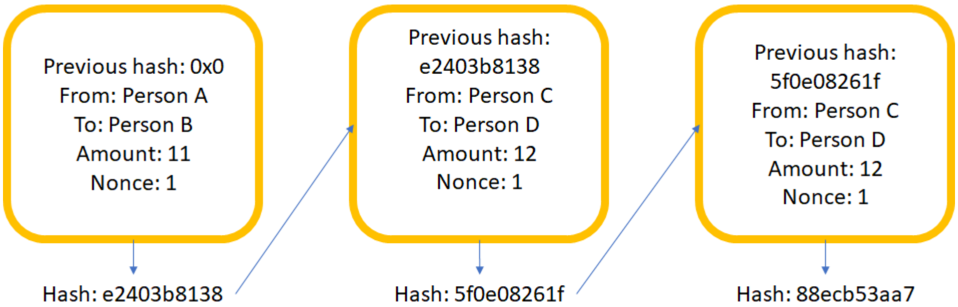

Figure 2.1: Blockchain example 1.

As figure 2.1 illustrates, the block data is translated into a hash, this hash is used in the following block, the data in this following block is in turn hashed and used in the following block and so on. Now if someone would try to change the transaction in the second block, to instead of transferring 12 from person C to person D to transfer 120, the hash of this block will change to $e0979d7fe7$, as a result, the hash code of the following block will change to $5d0932b248$ and all blocks that follow will change as well, making it very easy to detect when a value is changed even slightly. This means that all hash codes of the continuing blocks have to be recalculated.
In this example, it was possible to compute the hash codes almost instantaneously, meaning that if one would tamper with previous conducted transactions, it is still feasible to compute the hash codes of all the following blocks. However in Bitcoin certain conditions are given to the hash code, making it computational intensive to find the hash code that matches a block, this is where the nonce starts to play a role. This will be explained in section 2.2.5.

2.2.5 Consensus

There are multiple consensus algorithms, this chapter only discusses the Proof of Work and the Proof of Stake algorithm, later to be called PoW and PoS algorithm because these are the most commonly used consensus algorithms for cryptocurrencies. Before understanding the PoW and PoS algorithm, it is important to understand the concept of miners and the role miners play on the blockchain network. Mining and consensus algorithms are mainly important on a public blockchain. The difference between private and public blockchain will be explained in section 2.2.7 and 2.2.6 respectively.

Miners

Miners on a blockchain can function in different ways depending on the type of blockchain. In this explanation, miners as in the Bitcoin network are explained. A blockchain network as Bitcoin knows miners. Miners are competing with each other to find new valid hashes. This is incentified by rewarding miners that find a new valid hash with a reward in Bitcoin, the process of finding valid hashes takes computational power and will be explained in the next subsection. In Bitcoin

this reward is halved every time period and eventually disappears, in this situation miners might start to require higher transaction fees.
Whenever a valid hash is found, the finder broadcasts this to the network, after which all miners on the network will validate if this hash was indeed valid, if the hash is valid the broadcaster is rewarded with the mining fee. Whenever a new longest chain of blocks is found that meets the hashing conditions, this chain is agreed upon by the miners in the network as the new single source of truth.

PoW algorithm

As stated in section 2.2.4, hashes are easy to compute, however to make it more computational intensive to find the right hash, certain conditions are set to a hash code. If it would be required for a hash code to start with for example a 0, then nonces are added to the data in a block until the hash code is found that start with a 0. If the condition would be that the hash code needs to start with 5 zeros, nonces are tried until a hash code is found that start with 5 zeros, which takes significantly more computing power. Finding the hash code that meets a certain condition by changing the nonce is known as the PoW algorithm. Because the only way of finding a hash code that meets this condition is by trying different nonces until the hash is found that satisfied the conditions. Taking the previous example as in figure 2.1, if the condition is added such that the first six signs of the hash function must equal zero, it becomes as shown in example 2.2.

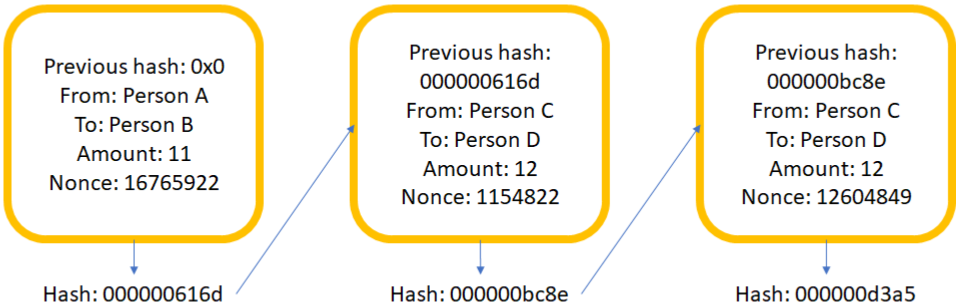

Figure 2.2: Blockchain example 2 PoW.

Now if someone tries to tamper with a block, for example changing the transaction in the second block to 120, the hashes of the second block and the third block would become *38d8dd9b1f* and *90bc2991b6* respectively and these hashes clearly don't meet the condition for a valid hash, meaning that in order to make these transactions valid the PoW needs to be redone. As the computing power required to find a valid hash increases, it becomes more difficult for an individual to tamper with the transaction record. The hash code in example 2.2 is still fairly easy to compute, however in the Bitcoin network, the condition of finding a valid hash is changed every certain number of blocks, changing the computational power required to find a valid hash so that on average the network of miners will find a valid hash every 10 minutes [10]. Making it in the current situation infeasible for a single miner to tamper with historical blocks and then recomputing all the hashes of the following blocks, because a single miner simply doesn't have enough computing power. This keeps the network trustworthy as long as more than 50% of the miners are just and not trying to tamper with (historical) transactions.

PoS algorithm

A different consensus algorithm is the PoS. The purpose of the PoS algorithm is the same as with the PoW, however the process to reach the goal is quite different [13]. In the PoS algorithm, validators are given the opportunity to place bets on which blocks are going to be finalized. For example if a validator bets on block A to be mined, if the block is actually mined the validator receives a reward, if the block is not mined the validator will be penalized, incentifying validators to bet only on blocks with a high probability of being mined. Penalizing validators for betting on blocks that are not likely to be mined is necessary because of the nothing-at-stake problem [12]. If a validator shows bad behavior by for example betting on blocks that are not likely or should not be mined, the system has to be able to punish this misbehavior, otherwise validators that misbehave have nothing to lose and may just as well try to process unlikely transactions on the blockchain. In this situation, validators will only bet on a block if there is a strong believe the block will be processed in order to avoid the penalty. In this situation the validators are incentified to bet on the blocks which they think are most likely to be processed in the future, driving the process toward convergence [15]. The validators will vote on what block should be processed, where voting right will be proportional to the validators coins in a proof-of-stake chain. If one validator would buy up more than 51% of the voting right in such a proof-of-stake chain, the community simply has to apply a patch where the clients ignore the attacker's fork [15]. A few benefits of the PoS algorithm is that it encourages community involvement, because holders of coins have the incentive to mine, it is less computationally intensive since the mathematical puzzle to solve is a lot easier in PoS and the miners have the incentive to keep the ledger of transactions correct, since the miners that are mining actually have a stake which loses value if the ledger would be corrupted, plus validators showing fraudulent behavior can be penalized, losing a stake in the network.

2.2.6 Public chain

Cryptocurrencies like Bitcoin and Ethereum where everyone can participate in are examples of cryptocurrencies that use a public blockchain. In a public blockchain anyone can become a miner, can own and manage the shared ledger and execute the consensus algorithm. These networks typically have mechanisms that incentify participants to join the network [29]. The PoW algorithm as explained in section 2.2.5 is a consensus algorithm which requires a relatively large amount of computing power and is fairly slow to execute, however in a public network it is a feasible algorithm to keep the right transactions in the ledger as long as more than half of the computing power in the network is dedicated to legitimate transactions. There are cases where parties want to use a blockchain, but don't want it to be publicly available, where only limited access is given to whom joins the network, where transactions need to be processed quick and the consensus mechanism should be less computationally intensive. In this case, people can use a private blockchain. One of the strong features of a public blockchain is that it is censorship resistant, this means that whenever someone has access to the internet, it is possible to access the blockchain, even when governments try to forbid or discourage the usage of a blockchain.

2.2.7 Private chain

There are cases where people or companies do want to have the benefits of a blockchain infrastructure, but don't want everyone to be able to access it. In this case private blockchains can be used. Anyone added to this network has to get an invitation by the initiator of the blockchain network, or must be listed in the rules put in place by the network initiator. Private blockchains such as for example Hyperledger have a greater magnitude of transaction throughput and have greater scalability than most public blockchains [29]. In a private blockchain network only permissioned participants will have access to the transactions, which allows for advanced governance schemes.

2.2.8 Ethereum

The largest blockchain platform supporting advanced programmable logic currently available is Ethereum [4]. Ethereum is currently the second largest cryptocurrency in terms of market capitalization [2]. In Ethereum one can not only transfer money, but it also allows for transferring programmable code in which a.o. it is possible to program the behavior of the money. So enabling the concept of programmable money. Bitcoin also allows for sending data in transactions and some programmable logic, however Ethereum allows more complex code structures making Ethereum much more suitable for, for example programming pension schemes. The behavior of this programmable money can be specified in a so called smart contract. One of the key aspects of the Ethereum blockchain is the Ethereum Virtual Machine. The Ethereum Virtual Machine is able to understand the compiled programmable logic specified in the smart contracts and execute this code. The programming language in which such smart contracts are written is called Solidity which is a contract-oriented programming language. On the Ethereum blockchain there is an unit that represents computational power called gas. When doing a transaction on the Ethereum blockchain it is possible to specify the maximum amount of gas which can be spend in a transaction and specify the gas price (offering a higher gas price makes it more likely for your transaction to be executed). So if one for example would invoke an infinite loop, the computational power required to do that transaction will reach the maximum amount of computational power for that transaction and so ending the computation, restoring the state of the blockchain to before that transaction, except for the gas required to do the computation. Since Ethereum is open source, it is possible to make a copy of this blockchain. If one does so, it allows for specifying your own rules in terms of transaction costs, gas limits, consensus algorithm, etc. One example of such a copy is Quorum [14]. Since the consensus algorithm is different in this copy, it allows for a much larger volume of transactions. It also allows for setting up rules on who may or may not join the blockchain, or what parts of the blockchain are accessibly by what party.

2.2.9 Smart contracts

When doing a transaction on the blockchain, it is possible to not only transfer some value, but also to transfer some data. This data can be a certain script in which one can express a certain logic. In this logic one can program a certain set of rules. This enables one to not only transfer money, but also to program how this money will behave using this logic which is included in the transaction in a so called "smart contract" [4]. The data that can be send in a transaction is limited, as is the maximum computational power one transaction can execute. Just like the transactions in the blockchain these smart contracts are immutable. The coded logic in the smart contract is always enforced, meaning that code is law [38]. This means that one doesn't require trust in the counter-party in order to make an agreement, because the agreements are enforced by code[8]. The concept of a smart contract isn't new, in the 1990s it already became clear that algorithmic enforced agreements could become a significant force in human cooperation [40], the Ethereum blockchain is an example of an environment that integrates this concept into the transactions of money.
This concept of smart contracts enables a wide range of applications, from programming a vending machine to place and pay refilling orders when it is below a certain inventory level, to setting up automatic payments for options on the stock market. These smart contracts do not enable users to do anything which was previously impossible, however it solves a problem by minimizing the need for trust [38].

2.2.10 Decentralized applications

Decentralized applications, later to be called Dapps, take the concept of a smart contract a step further. A Dapp is basically an application that is running on the blockchain. With the use of one or multiple smart contracts, one programs a certain logic that specifies the behavior of money in that application. Some examples of Dapps are: OpenBazaar, a market place for buying and selling items in the physical world managed by a Dapp, LaZooz, which is a ride sharing platform,

[8]A car auction managed in a smart contract can be an intuitive example of how agreements are enforced by code. The owner of a car offers a car on an auction in a smart contract. People are able to bid on the car via the smart contract, temporarily storing the money they have bid in the contract. At the closing time of the action, ownership of the car is transferred to the highest bidder, the money of the highest bidder is transferred to the now former car owner and the money that was stored in the contract of the losing bids are returned to the losing bidders. If the described logic as in this example is specified in the smart contract, there is no way to deviate from this agreement. People that bid are required to have sufficient funds in their balance, bidding does oblige the bidder to buying the car if it is the winning bid and the owner of the car is forced to accept the winning bid.

Twister which is a social peer-to-peer micro blogging platform, Bitmessage which is a secure way of messaging or Storj which is a file storing platform [38].

2.2.11 Decentralized autonomous organizations

A decentralized autonomous organization, later to be called DAO, is similar to a Dapp, but more complex. A DAO is a decentralized organization or cooperation ran on the blockchain that can function fully autonomously without any human involvement. This is a concept derived from artificial intelligence where autonomous agents perform the tasks involved with running a cooperation that where otherwise performed by humans. This concept is particularly interesting for financial companies. This could for example enable fully automated banks or even pension funds on the blockchain. Imagine a pension fund DAO where anyone in the world can participate in, that is automatically investing according to the investment preference and life-cycle of the individual pension fund participants, where longevity risk is shared in a predefined and transparent way and that starts doing periodical payments at the age of retirement.
This research is the first step in making this dream on what pensions will entail in the future a reality. This project may lay the foundation for a pension product that may very well set the standard in pensions in the future.

Chapter 3

Simulation model

This chapter describes a simulation model that simulates the behavior of a Tontine based pension fund as designed during this research. This chapter will give insights on the requirements of this simulation model, the model's scope, the model's input variables, how the data is processed in the simulation model, the output variables the simulation model gives and how the model is validated. The simulation model is a key element of this research, because the simulation model can generate data for actuarial analysis, so that the behavior of a Tontine based pension can be studied under different conditions and so that the pension logic can be further developed to make the Tontine based pension as fair as possible and feasible for deployment on the blockchain. The simulation model has the possibility to use different methods for allocating the mortality gains, which is important for determining what way of sharing longevity risk using the Tontine principle is fair (subquestion 1) and is of key importance in a Tontine based pension.

The simulation model can start with a pension fund where the pension fund is already filled with pension fund participants (possibly from different salary groups), or where the pension fund starts with zero participants where there is a varying influx of pension fund participants, or with a pension fund which is already filled with pension fund participants but drains over time. These dynamics are relevant for studying how the number of pension fund participants in the Tontine pension fund influence the benefit payments of its participants (subquestion 2).

It is possible to do simulations with different economic scenarios which is very important for studying the volatility of the benefit payments of the pension fund participants (subquestion 3), so that insights can be obtained on the volatility of a Tontine based pension fund.

The life expectancy and the homogeneity of the pension fund participants can be changed in the simulation model so that insights can be obtained in how to deal with an increase in life expectancy and how to deal with groups of pension fund participants that have life expectancies that differ from each other (subquestion 4).

3.1 Requirements

Discussing the simulation model with an interdisciplinary team of developers and actuaries led to the model's scope and requirements. The simulation model should provide insights in:

- How mortality gains can be allocated in a Tontine pension.
- What the fairness of sharing longevity risk in a Tontine pension using different methods of allocating mortality gains is.
- How different subsets of the pension fund participants with different life expectancies influence the process of longevity risk sharing.
- How many pension fund participants should participate in a Tontine pension for it to be fair and feasible.
- How much pension benefits pension fund participants that participate in this Tontine pension will receive.
- The dynamics of a starting Tontine pension managed on a blockchain infrastructure.
- What happens if pension fund participants start leaving the Tontine pension fund.
- How asset returns can be allocated to individual pension fund participants.

3.2 Scope

A simulation model is always a trade-off between development time and complexity. Therefore certain aspects were let out of scope of the simulation model.

In scope

- Death probabilities are based on AG2016 tables [16], as it is important in the simulation model to generate death events that are realistic.
- Deterministic returns, in order to compare scenarios with for example different numbers of pension fund participants without experiencing the variation from different economical scenarios.

- Stochastic returns based on the model used for the "haalbaarheidstoets" from De Nederlandse Bank later to be called DNB conform the advice of the commission of parameters [7], in order to determine pension benefits of different pension fund participants in different economic scenarios.
- Discount factors based on different term structures [24], so that the term structures are sampled with a correlation with for example return on equity and return on bonds to give a more realistic reflection of the truth.
- Different life-cycles based on age and investment preference, to better reflect realistic investment preferences.
- Difference in death probability, pension balance and income given an age, so that analysis can be done on how fair longevity sharing is when pension fund participant from different life expectancies are joined in one solidarity group, with and without accounting for the difference in life expectancy when computing annuity.
- Annuity computations based on death probabilities and interest rates, so that on every year the annuity payment of a pension fund participant can be computed based on the survivor rate and discount rate.
- Variable growth of the number of pension fund participants to model how a starting Tontine pension on the blockchain would behave, given that it starts with a certain amount of pension fund members and would grow over the years.
- A pension fund where pension fund participants leave the pension fund over time to model what happens to the benefit payments of the remaining pension fund participants when pension fund participants start leaving the pension fund at different rates.
- Variable pension ages are taken into account, to see what influence different retirement ages have on the fairness of the benefit payments and to see how to cope with a population that has a higher life expectancy.

Out of scope

- This pension product covers only an individual's old age pension. Making factors like marriage, children and a partner irrelevant for the simulation model.
- Changing employers is not taken into account as the pension product isn't necessarily bound to a specific employer.
- To simplify the simulation model, unemployment and disabilities are not taken into account. This does not undermine the simulation model's usability as no analysis on these factors are desired for this research.
- Event driven simulation is not implemented, since the simulation model is mainly used for actuarial analysis, it would suffice to do the simulation steps with a yearly time interval, so that all computations are done on a yearly base.
- Difference in premium payment, the assumption is made that every pension fund participant yearly pays a certain percentage of its salary to the pension fund, not missing payments and not being late with payments.
- The simulation model accounts for salary growth over time, but only following a fixed pattern.
- The simulation model doesn't take inflation into account.
- The simulation doesn't take coverage ratios into account, but models the balance of the individual pension fund participant and pays no more than the money an individual has paid in premium, gained from investment returns and mortality gains.

3.3 Model description

The information in this section is confidential and therefore unavailable in the public version.

3.4 Model startup

The Tontine pension product is an ongoing product without an end, therefore the simulation model that mimics the behavior of the Tontine pension product is a non-terminating simulation model. This means that in order to do analysis on the generated output data, it is important to first let the simulation model reach a steady-state. The model can reach a steady state with the use of a warm-up period. The length of the warm-up period is dependent on the time required for the observed variable to reach a steady-state mean. The observations from the warm-up period are not used for the analysis. This warm-up period does however not apply when the behavior of a starting Tontine pension fund is modeled. The required warm-up period for the simulation model can be determined using the Welch's Graphical Procedure [30].

3.4.1 Determining warm-up period

In order to determine the warm-up period of the simulations, the Welch's graphical procedure is used. The Welch's procedure consists of four steps [30].

1. Make n replications of length m where $n \geq 5$, where Y_{ji} is the ith observation of the jth replication.

2. Compute the average of the observations $\bar{Y}_i = \frac{\sum_{j=1}^{n} Y_{ji}}{n}$ for $i = 1, 2, ..., m$.

3. In order to smoothen out the oscillations in the computed averages, the moving averages are computed as follows:

$$\bar{Y}_i(w) = \frac{\sum_{s=-w}^{w} \bar{Y}_{i+s}}{2w+1} \qquad \text{if } i = w+1, ..., m-w$$

$$\bar{Y}_i(w) = \frac{\sum_{s=-(i-1)}^{i-1} \bar{Y}_{i+s}}{2i-1} \qquad \text{if } i = 1, ..., w$$

   Where w is the window and is a positive integer such that $w \leq \frac{m}{4}$, meaning that the window cannot exceed a fourth of the simulation length. These are called moving average because i is the ith observation point moving through time.

4. Visualize the moving averages $\bar{Y}_i(w)$ in a graph for $i = 1, 2, ..., m-w$ and chose warm-up period l beyond the point where $\bar{Y}_1(w), \bar{Y}_2(w), ...$ appears to have converged.

The Welch's procedure should initially be done with $n = 5$ or 10 replications, preferably with a m as large as practical considering the speed of the simulation model. If the values in the visualization don't seem to converge, more replications of length m should be added, so that $\bar{Y}_i(w)$ will get smoother.

Chapter 4

Feasibility study

This chapter describes a feasibility study of the Tontine pension product, modeled as described in chapter 3. Some aspects of the Tontine pension model were analyzed without the use of the simulation model. This chapter answers the first four subquestions of this research. It is important to note that the replacement rates in this chapter are excluding AOW, which is the first pillar of the Dutch pension system, the general old age pension.

4.1 Allocation of mortality gains

The allocation of mortality gains is at the core of the Tontine pension model and in this context is key in sharing longevity risk. The rules for allocating mortality gains can have a great impact on the benefit payments that pension fund participants will receive. So far two ways of distributing mortality gains are discussed; proportional to capital and fair mortality gains. [25] explains why to use the fair mortality gains method over allocating mortality gains according to capital, because this favors younger people to join a pension pool, so that they can benefit of more instances of mortality gains than older people. In order to use the fair mortality gains method, one has to solve a system of non-linear equations which gets computationally intensive as more people join the Tontine pension fund. In section 4.1.1 the risk premium method is logically deduced as a method for allocating mortality gains. The risk premium method is less computationally intensive as the fair mortality gains method. In this section the different ways of allocating mortality gains will be evaluated in terms of distribution and replacement rate. In determining the method of allocating mortality gains, both the fairness as the computational complexity are important as it should be managed on the blockchain in which one has to cope with some technical limitations and computational constrains.

4.1.1 Risk premium

The information in this section is confidential and therefore unavailable in the public version.

4.1.2 Distribution

The difference in mortality gains allocation methods can be demonstrated using a fictional example. In this example there is a Tontine pension fund of 160 people, 80 women and 80 men, of which one man dies and his pension result is allocated to the surviving participants. The balances are based on the cumulative premium payments of each individual to that point assuming that all pension fund participants joined at age 25 and have had the same salary developments. Figure 4.1 and figure 4.2 show the distribution of the different types of mortality gains for men and women respectively.

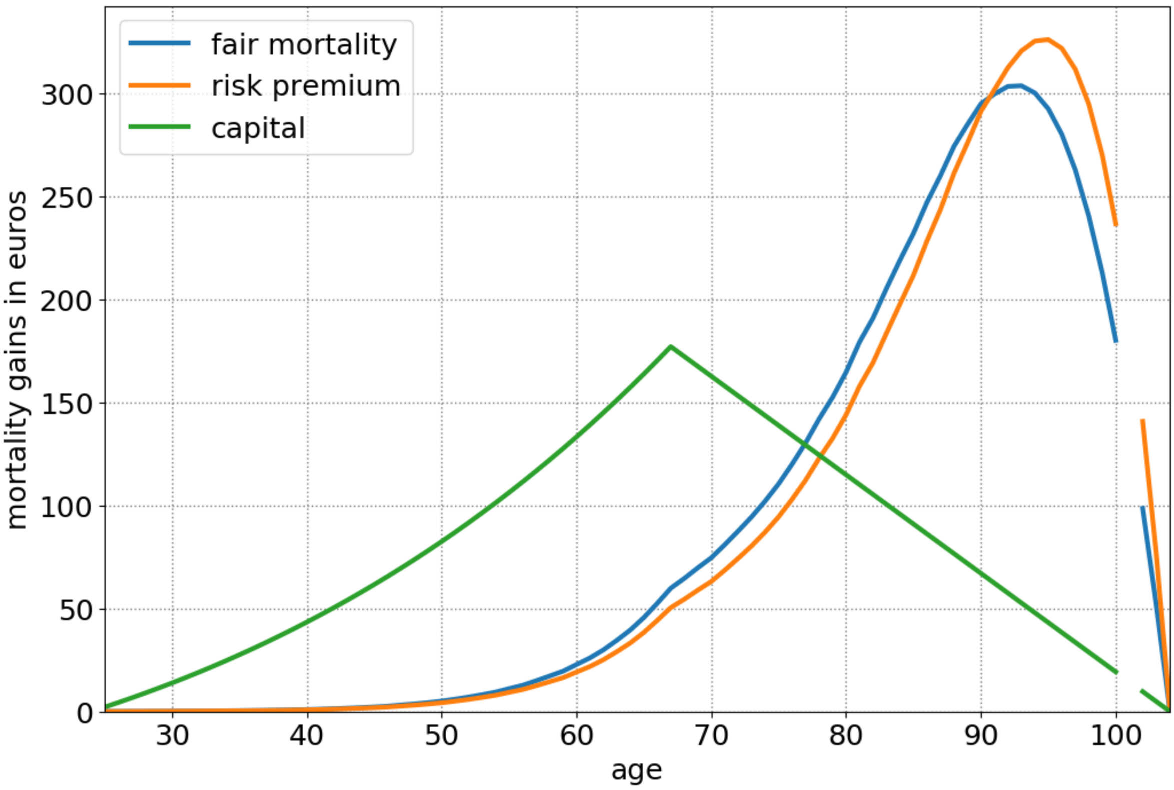

Figure 4.1: Comparison different mortality gain distributions for men.
Reading example: The graph shows the distribution of mortality gains when a single person dies. Consider age 80 in the graph. In this case a man dies of age 101 and the capital remaining in his balance is allocated to all living participants, in the case of allocating mortality gains according to capital the living male pension fund participant of age 80 would receive approximately 115, when allocating mortality gains using the risk premium method the living pension fund participant of 80 would receive approximately 144 and in case of using the fair mortality gains method approximately 165.

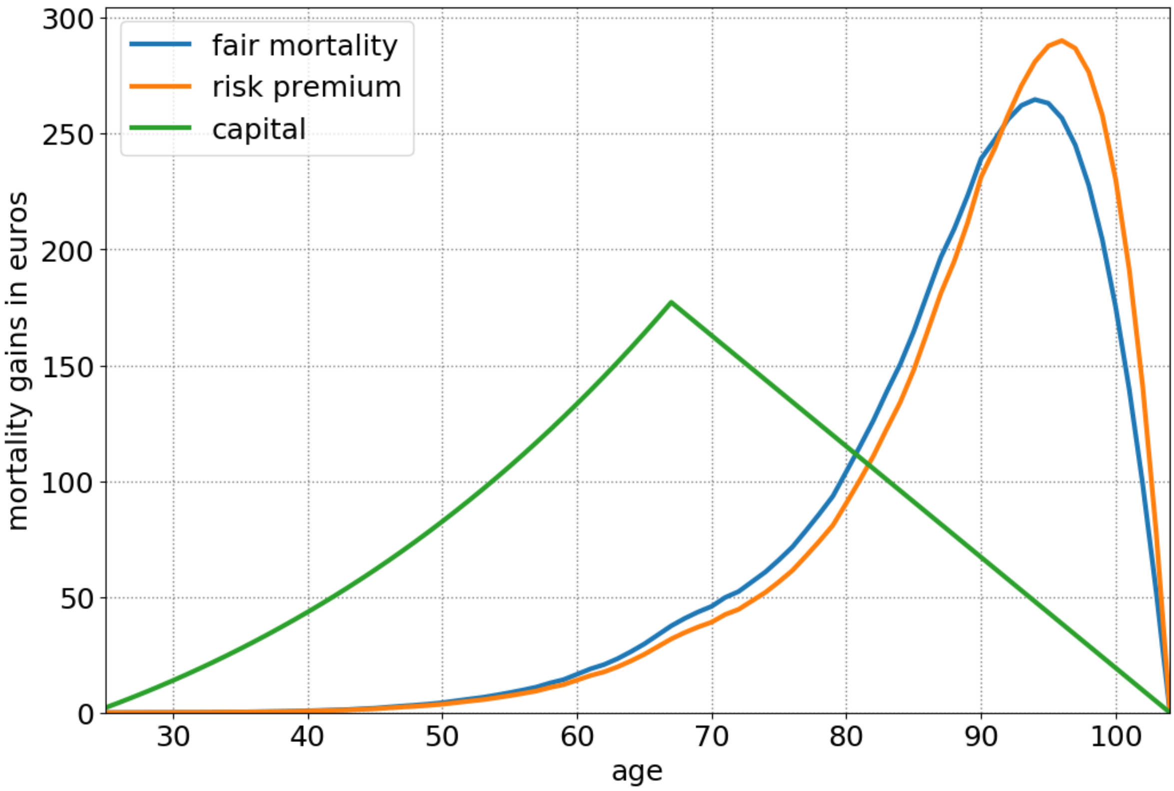

Figure 4.2: Comparison different mortality gain distributions for women.
Reading example: The graph shows the distribution of mortality gains when a single person dies. Consider age 80 in the graph. In this case a man dies of age 101 and the capital remaining in his balance is allocated to all living participants, in the case of allocating mortality gains according to capital the living pension fund participant of age 80 would receive approximately 115, when allocating mortality gains using the risk premium method the living female participant of 80 would receive approximately 91 and in case of using the fair mortality gains method approximately 104.

Distributing the mortality gains according to capital, shows the majority of the mortality gains are distributed to people around the age of retirement, which is logical since that is where their account balance would be at its highest. These figures also show that both the fair mortality gain method and the risk premium method allocate most of the mortality gains to the pension fund participants of a higher age. The fair mortality gains method and the risk premium method seem similar in distribution, however the fair mortality gains method gives more mortality gains to men and women younger than 91 and 92 respectively, whilst the risk premium method gives more mortality gains to the pension fund participants from that age and higher. One can see a disruption in the line of figure 4.1, this is caused by the mortality of the 101 year old man whose remaining account balance is allocated to the surviving pension participants.

4.1.3 Replacement rate

In order to illustrate the difference in benefit payments of the different mortality gains allocation methods, this section elaborates on the different replacement rates per age for a pension fund where the mortality gains are distributed according to capital and the risk premium method, the fair mortality gains method is left out of account since the risk premium method is very similar to the fair mortality gains method and due to the computational complexity of the fair mortality gains method. In order to demonstrate the difference in replacement rate, a deterministic economic scenario is used to reduce variation induced by stochastic economic scenarios.

Warm-up period

The warm-up period of the simulation is determined as described in section 3.4.1. Since the variable evaluated is the replacement rate, this variable is visualized in order to determine the warm-up period.

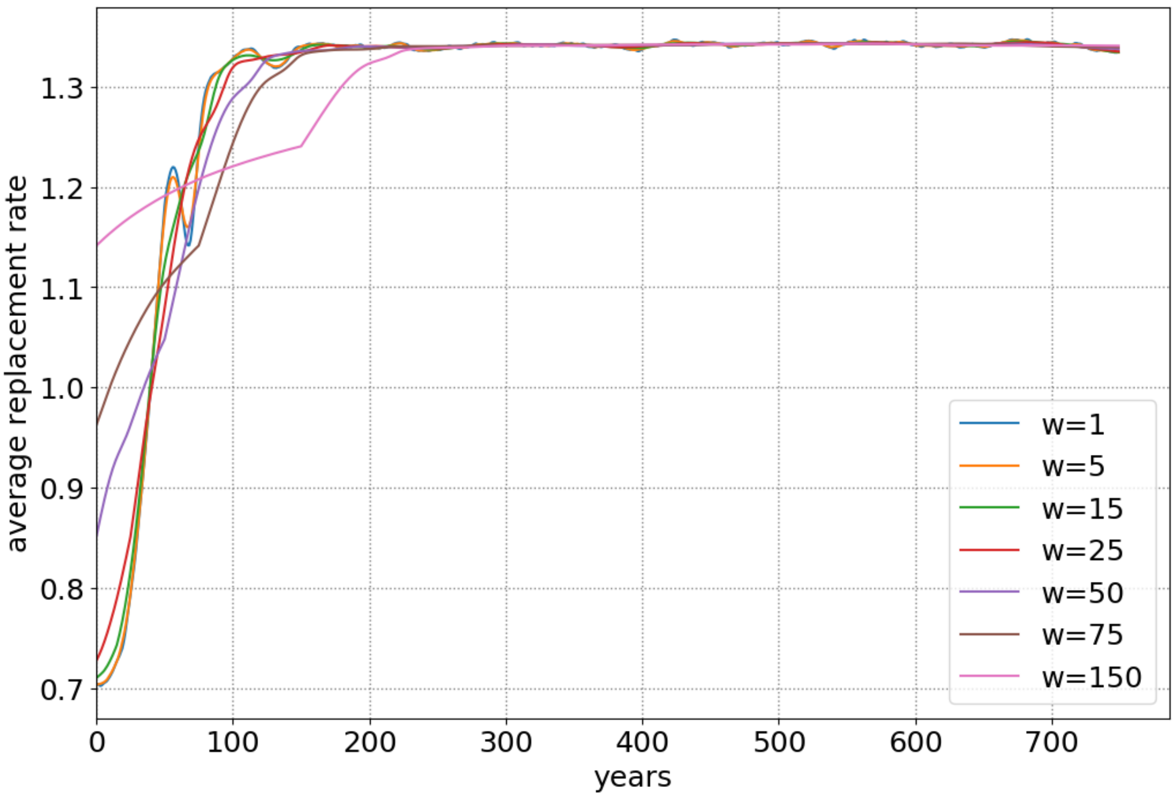

Figure 4.3: Welch Graphical Procedure to determine model warm-up time, where the weights w are as explained in section 3.4.1.
Reading example: the observed variable in the graph is the average replacement rate, averaged over all retirees. Considering year 200, the blue line ($w = 1$) represents the average replacement rate averaged over year 199, 200 and 201 and the orange line ($w = 5$) the average over 195 to 205.

Figure 4.3 is a visualization of the Welch Graphical Procedure using ten simulations of a length of 1000 years, where there is no distinction between salary classes and adjusted mortality factors based on salary classes with a pension fund population of 10000 pension fund participants. This figure illustrates the need of such a warm-up period because in the first 100-200 years of the simulation, there is variation induced by starting the simulation model. After 250 years the weighted average of the observed variable seem to have converged to a stable replacement rate. So taking a warm-up period of longer than 250 years should reduce the variation induced by starting the simulation.

Comparison replacement rate per age

To visualize the replacement rate per age given a certain mortality gain allocation, a simulation was performed with a duration of 500 years using a warm-up period of 300 years, not distinguishing between salary classes and corresponding differentiations in death probability, keeping the economic scenario constant, with a pension fund pool size of 10000 pension fund participants, whilst varying in the method of allocating mortality gains, using the distribution proportional to capital method and the risk premium method.
Figure 4.4 demonstrates that using this method of allocating mortality gains, favors the younger retirees in terms of replacement rate, whilst older retirees receive a lower replacement rate, that converges to a benefit payment of 0. One can argue if it is a desirable situation that when retirees reach a certain age, they receive close to zero benefit payments. In figure 4.4, the black bars show the average plus and minus the standard deviation of the replacement rate for that age, in the simulation run there has only been one pension fund participant of age 111, which is why the standard deviation is missing in that instance.
When using the risk premium method to allocate the mortality gains, the older pension fund participants receive a greater benefit payment than at a lower age, figure 4.5 demonstrates this. This is caused by the fact that older pension fund participants benefit more from the mortality gains when the risk premium method is used for allocating mortality gains. Using the risk premium method, the benefit payments do not drastically decrease as pension fund participants get older. It would seem as if figure 4.5 implies that the majority of the benefit payments are given to the higher aged pension fund participants, however when looking at the age distribution of the pension fund participants, one can see that this is not the case as shown in figure 4.6.
Figure 4.6 demonstrates that as pension fund participants age, the number of pension fund participants of a certain age decrease at an increasing rate, this is what one would expect since death probabilities increase as pension fund participants age.

Individual case

Information about the average replacement rate and its standard deviation per age group can give insights on the development of the received benefit payments as pension fund participants

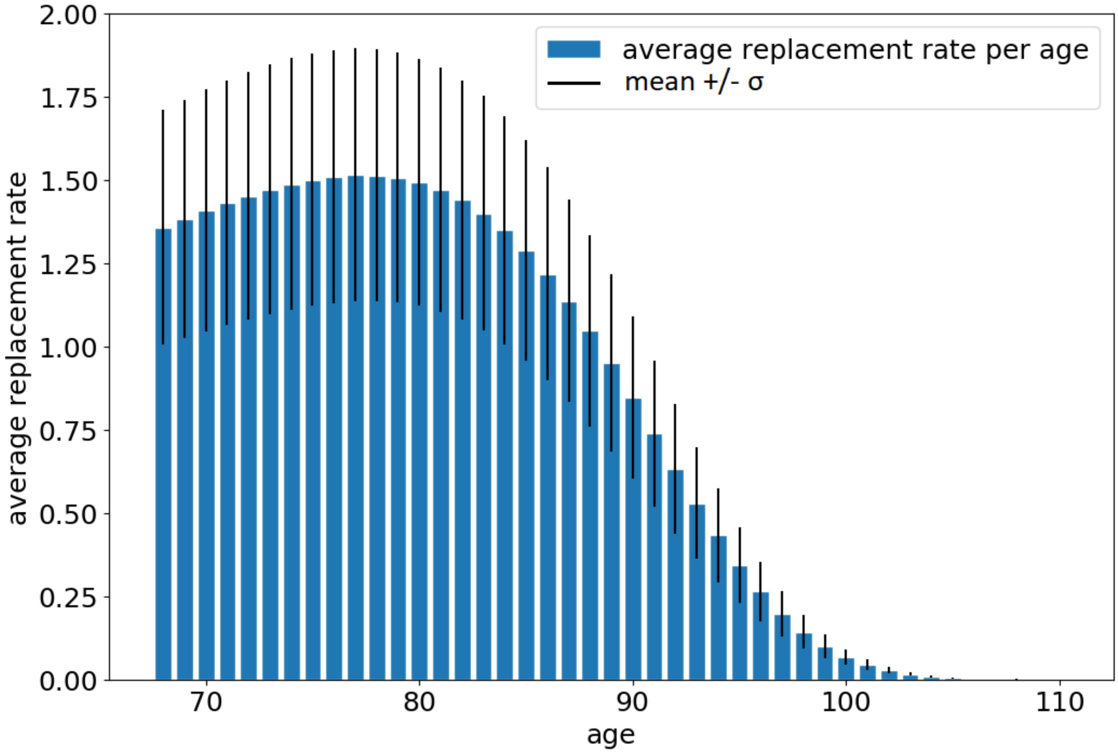

Figure 4.4: Average replacement rate per age with 10000 pension fund participants allocating mortality gains based on capital.
Reading example: The graph shows the replacement rate per age averaged over a simulation of 500 years and its standard deviation. Considering age 80, the replacement rates of all pension fund participants of age 80 in every simulation year are registered, at the end of the simulation these values are used to compute the average and standard deviation of the replacement rate for age 80.

age. However it can also be interesting to see what these figures mean for an individual pension fund participant. This can be done by tracking the replacement rate of the individual pension fund participant from age of retirement till death. In this section the replacement rates of three example pension fund participants will be shown, these pension fund participants will join the pension fund at an age of 25, 45 and 65, this directly demonstrates the influence of the way of allocating mortality gains on these individual pension fund participants their replacement rates. Figure 4.7 demonstrates the difference in benefit payments for the three individual pension fund participants. It should be noted that for this simulation a deterministic economic scenario is used with a yearly return on equity and bonds of 6% and 2% respectively and that no inflation is taken into account, also the lifespan of the tracked individual pension fund participants was forcefully extended to an age of 110 years, to better demonstrate the development of the annuity payments over time. The last earned salary for all pension fund participants in this case is 54288. In this simulation the pension fund participant joining at the age of 25 benefits most because

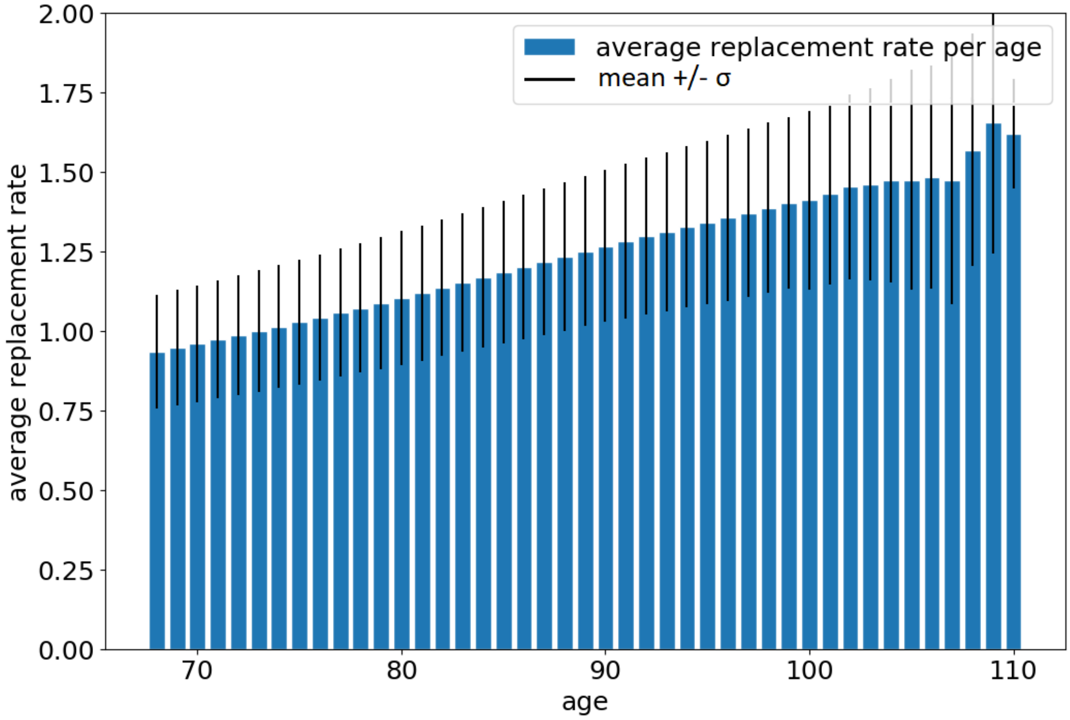

Figure 4.5: Average replacement rate per age with 10000 pension fund participants allocating mortality gains based on the risk premium method.
Reading example: The graph shows the replacement rate per age averaged over a simulation of 500 years and its standard deviation. Considering age 80, the replacement rates of all pension fund participants of age 80 in every simulation year are registered, at the end of the simulation these values are used to compute the average and standard deviation of the replacement rate for age 80.

of the steady positive equity and bond returns. All cases show that using the mortality gains allocating method of distributing proportional to capital yields the highest benefit payments for pension fund participants in the years directly after retirement, after that the received benefit payment declines rapidly as older pension fund participants do not benefit more from mortality gains using this allocation method, resulting that when an extreme age of for example 110 is reached, the benefit payment is close to zero. Note that on an age over 90, the annuity payment is still a fraction of the initial benefit payment. When allocating mortality gains according to the risk premium method, the benefit payment is lowest around retirement age and increases as one ages. As stated earlier, using different economic scenarios might change the shape of the lines in the graph. One of the strong points of the risk premium method is that the annuity payments do not converge to zero over time, preventing a period in which a pension fund participant would receive no or or little income. Since the risk premium method does not converge to zero as pension fund participants age, does not favor young pension fund participants over old pension fund participants and is less computational complex than the fair mortality gains method which is

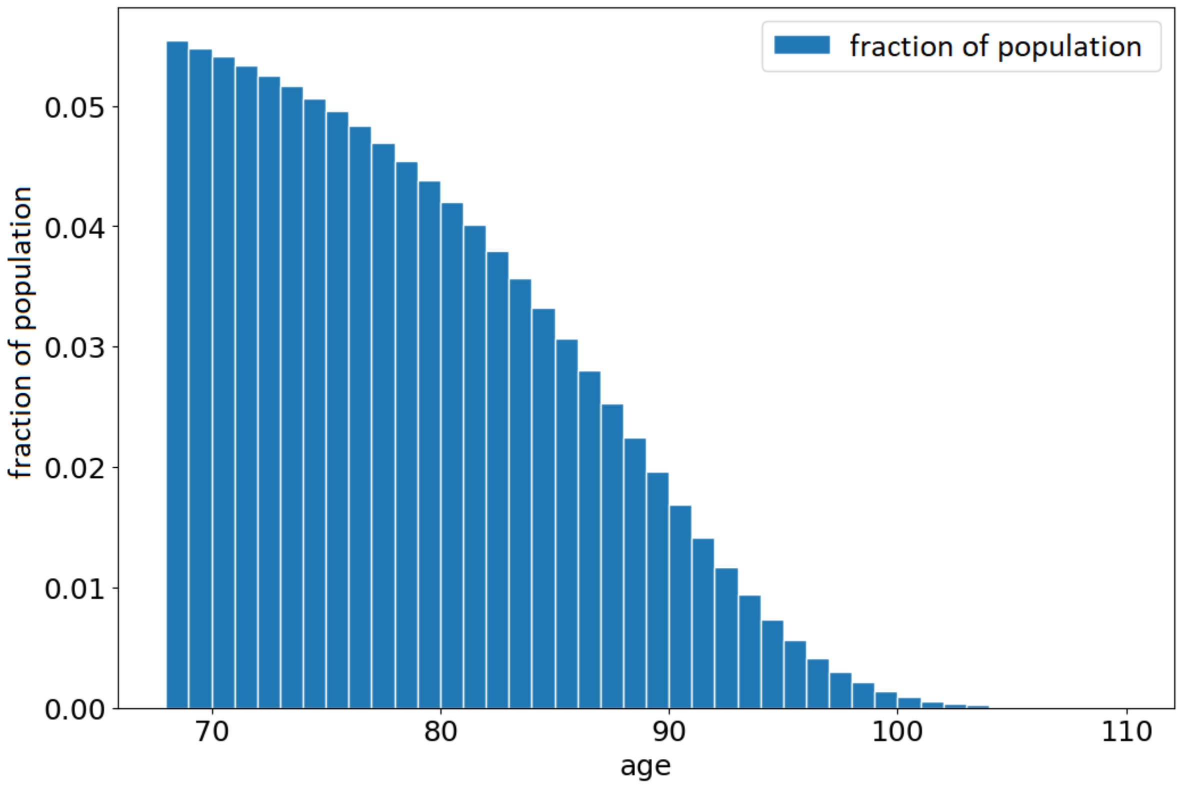

Figure 4.6: Age distribution with 10000 pension fund participants.
Reading example: this graph shows what fraction of the retired population has which age. Considering age 90, the graph shows that less than 2% of the population was of age 90, where the fractions are computed by counting the pension fund participants of year 90 over multiple years of simulation divided by total count of retired pension fund participants.

infeasible to implement using the Ethereum blockchain because of its computational complexity, the risk premium method is the preferred way of allocating mortality gains. From here on forward the risk premium method will be used for the subsequent analyses.

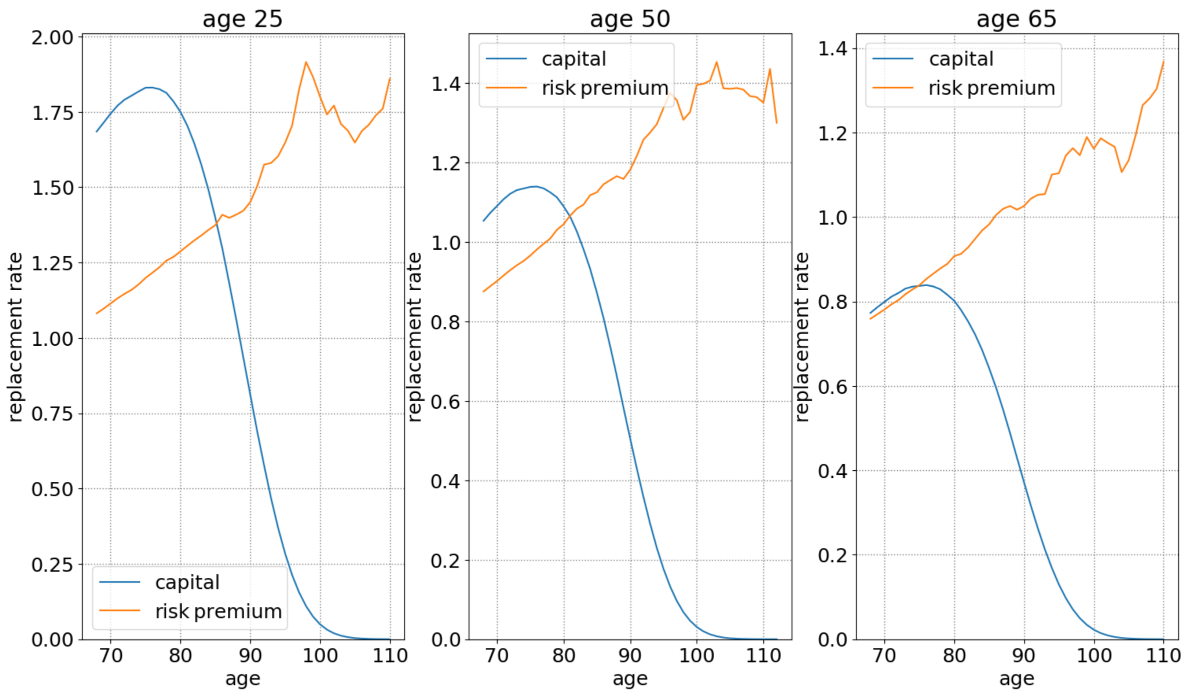

Figure 4.7: Replacement rate for three individual pension fund participants of age 25, 50 and 65. Allocating mortality gains using the risk premium method and proportional to capital. **Reading example:** These graphs show the replacement rate of three individual pension fund participants that join the Tontine pension fund on age 25, age 50 and age 65. Considering the graph in the middle with title age 50, at age 90, the graph shows that if the mortality gains are allocated proportional to capital, the replacement rate is just over 0.4. If the mortality gains are allocated using the risk premium method using the same simulation data, the replacement rate is approximately 1.2.

4.2 Minimum number of pension fund participants

How many people are needed to share longevity risk is a fundamental question in pensions and especially in this Tontine pension structure, since in this Tontine pension structure no buffers are held to compensate for drops in benefit payments of pension fund participants. In this section the average received benefit payments per age and its standard deviation is set against the different pension fund pool sizes. This gives insights in the pension pool size which is required to give pension fund participants relatively stable benefit payments, so that the variation induced by the pension pool size is sufficiently small.

Figure 4.8 shows the average replacement rate per age for different pension fund pool sizes. This is a data visualization of a simulation using the same parameters as in section 4.1.3, the only difference is that the simulation ran in 4.1.3 was run with 10000 pension fund participants whilst this simulation was run with a variable number of pension fund participants. Figure 4.8 shows that running a pension fund with only 10 pension fund participants results in a very

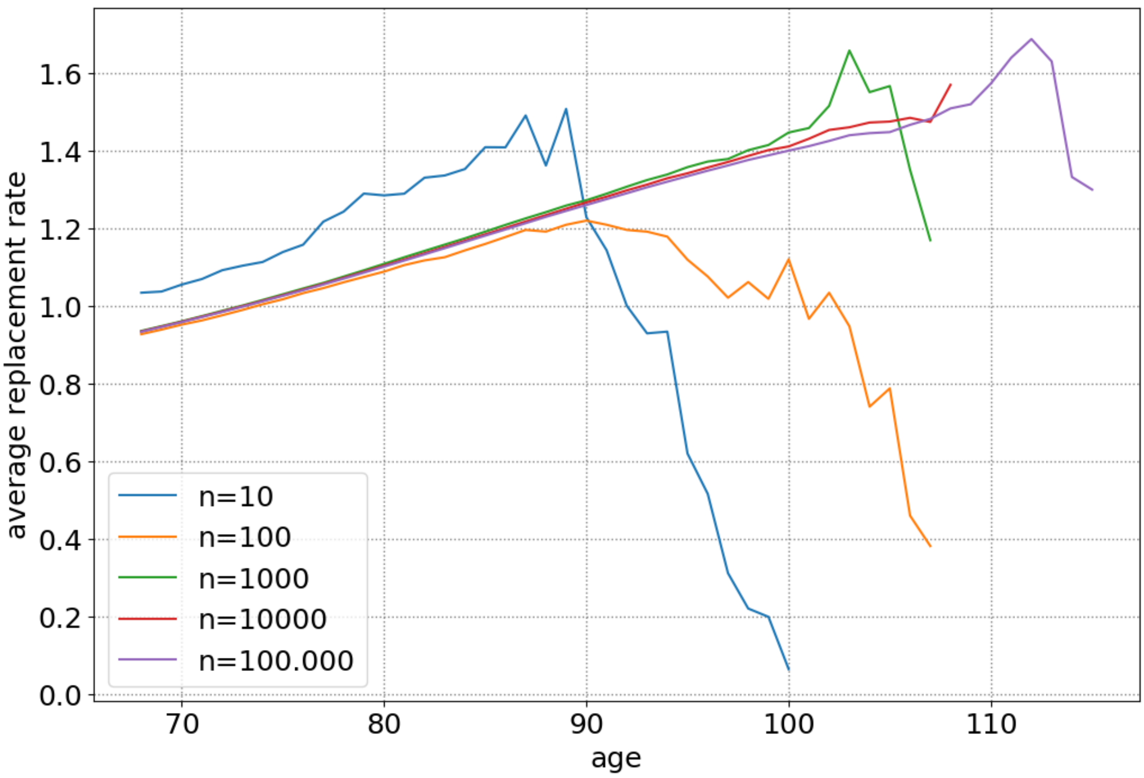

Figure 4.8: Average replacement rate per age for pension fund pool sizes n.
Reading example: This graph shows the replacement rate per age for different numbers of pension fund participants participating in the Tontine pension averaged over multiple simulation years. Considering age 95, the graph shows that with 1000 or more pension fund participants the average replacement rate is just below 1.3, while the average replacement rate of a Tontine pension fund with 10 or 100 pension fund participants is much lower. The average replacement rate for a given age is computed by doing multiple simulation years, registering the replacement rates of pension fund participants of a given age and averaging them at the end of the simulation.

fluctual average replacement rate as pension fund participants age. From 100 pension fund participants the average replacement rate is more or less equal to the average replacement rate of larger pool size till age 84. From age 84, it start to differ from larger pension pool sizes. From a pension pool size of 1000 pension fund participants the average replacement rate is approximately equal to that of larger pension fund pools for all ages, there is a deviation visible from age 100, however the number of pension fund participants reaching this age is very small considering a pension pool size of 1000 pension fund participants. The data shows that there is little difference between a pension pool size of 10000 and pension pools with more pension fund participants. When comparing the different pension pool sizes on standard deviation as shown in figure 4.9 it becomes clear that the pension pool size of 10 pension fund participants is the most volatile, except ages above 96, but this might be caused by the few pension fund

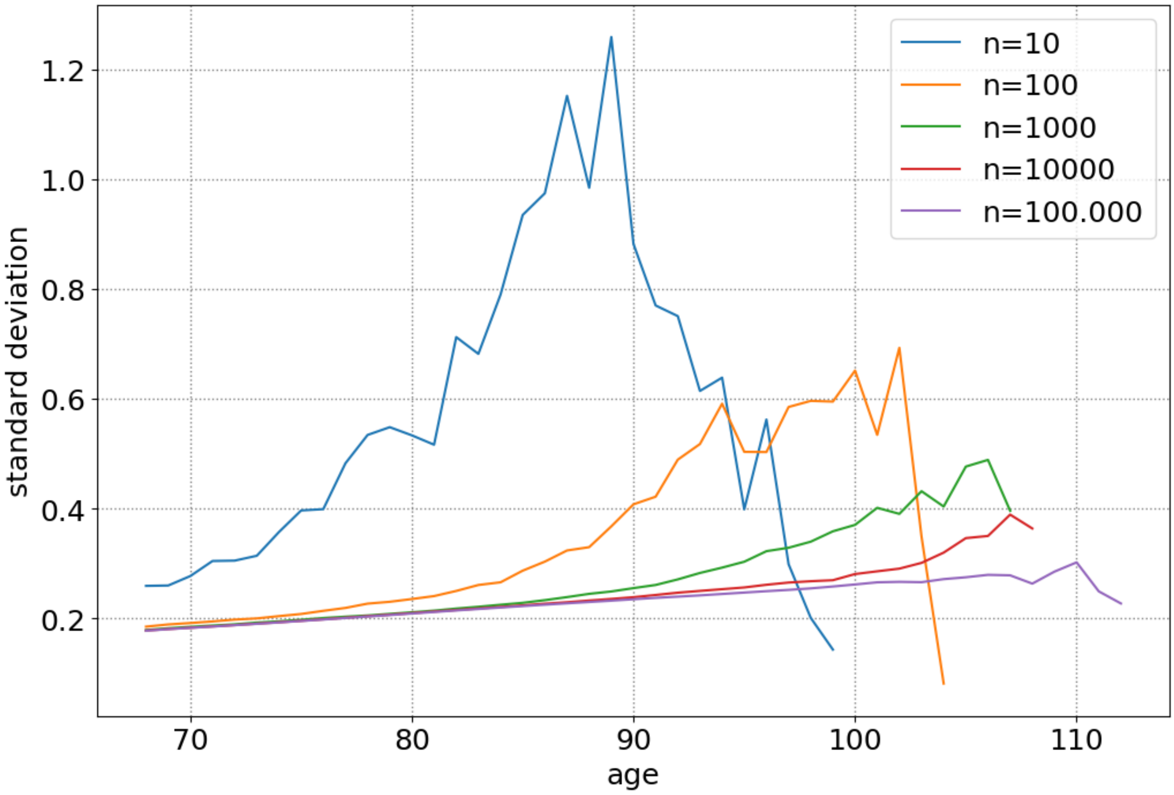

Figure 4.9: Standard deviation per age for pension fund pool sizes n.
Reading example: This graph shows the standard deviation of the replacement rate per age for different numbers of pension fund participants participating in the Tontine pension computed over multiple simulation years. Considering age 90, the graph shows a large difference in standard deviation. If the Tontine pension fund only has 10 pension fund participants, the standard deviation of the replacement rate is approximately 0.9, which is almost three times as high as when the Tontine pension fund has 100 pension fund participants. The standard deviation of the replacement rate for a given age is computed by doing multiple simulation years, registering the replacement rates of pension fund participants of a given age and computing the standard deviation of the registered replacement rates per age at the end of the simulation.

participants reaching this age in this situation. The standard deviation of the replacement rate of a pension fund with 100 pension fund participants seems to be approximately equal to the standard deviation of the replacement rate of pension pool sizes of larger than 100 pension fund participants till age 80. From this age, it is visible that the standard deviation of a pension pool of 100 pension fund participants is greater than that of pension funds with a higher number of pension fund participants. A pension fund of 1000 pension fund participants holds approximately the same standard deviation of the replacement rate till an age of 90 in comparison to the pension pools of larger sizes. Moving from a pension pool size of 10000 to a pension pool size of 100.000 yields almost no reduction in standard deviation.

Individual case

In order to see what effect the pension fund pool size has on the replacement rate of the individual, the same simulation setup was used as in 4.1.3, except for that in this simulation the risk premium method was used and the number of pension fund participants vary.

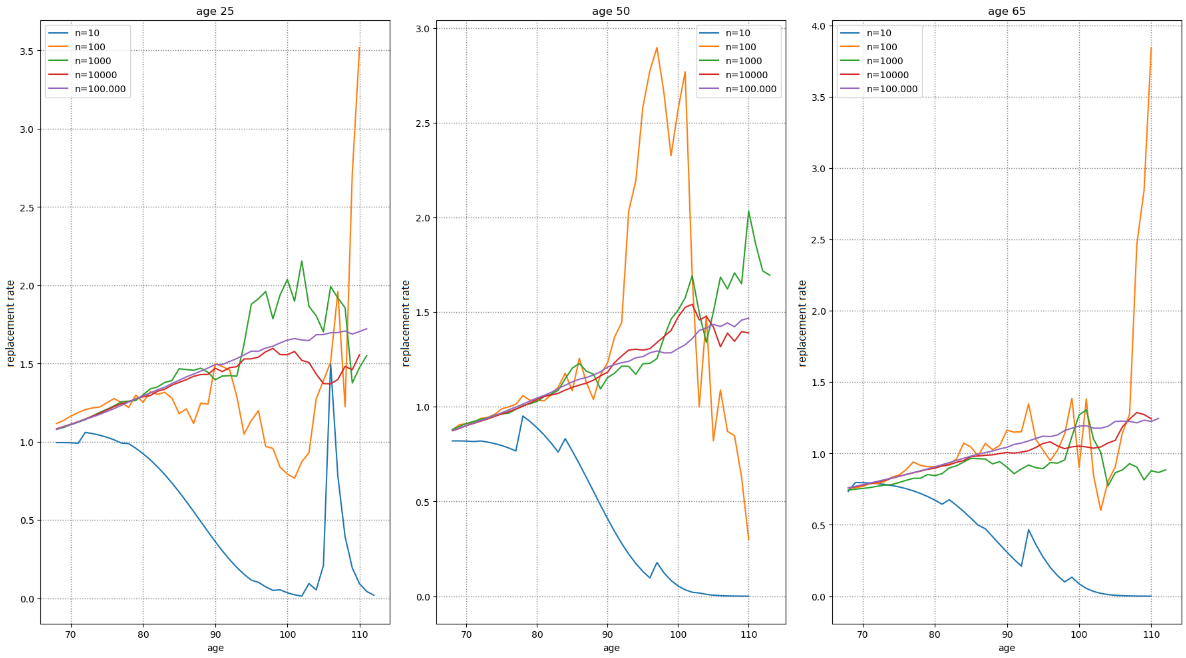

Figure 4.10: Pension replacement rates for three individual pension fund participants of age 25, 50 and 65. Using the risk premium method. Varying the pension pool size of the Tontine pension fund.
Reading example: These graphs show the replacement rate of three pension fund participants joining the Tontine pension fund at age 25, 50 and 65 for different numbers of pension fund participants. Considering the left graph of a pension fund participant that joined the Tontine pension at age 25 evaluated at age 90. With only 10 pension fund participants participating in the Tontine pension, the replacement rate of the pension fund participant that joined at age 25 has declined to just below 0.5 due to the absence of mortality gains, while a pension fund with 100.000 pension fund participants has provided the pension fund participant that joined at age 25 with a stable income from mortality gains, yielding a replacement rate of 1.5 at age 90.

Figure 4.10 demonstrates why the pension pool size is important. If it is too small, for example with 10 pension fund participants, an individual will not experience a constant flow of mortality gains, this can result in a replacement rate that becomes low at some point, might peek when someone deceases and may eventually converge to zero. Figure 4.10 demonstrates the same as implied by figures 4.8 and 4.9, the more people participating in a Tontine pension, the more stable the stream of benefit payments becomes.

4.3 Economic volatility

In a DB arrangement as described in 2.1.3, pension fund participants buy a certain pension claim. The individual pension fund participant does not hold economic risk individually. The economic risk is shared among pension fund participants with a DB arrangement. When the pension fund its coverage ratio reaches a certain threshold, the pension fund may decide to index the pensions of the pension fund participants, do nothing or even cut their pensions. The pension fund is responsible to hold buffers, so that even in times of low or negative investment returns, the pensions of pension fund participants can be indexed or remain unchanged. In the Tontine pension fund as defined in this research, the pension balance of pension fund participants can be traced to the individual. The money of these pension fund participants is invested according to their investment preferences and life-cycle as explained in section 3.3. This means that the pension balance of the individual pension fund participants is highly influenced by investment returns. This section shows how large the influence of investment returns is on the received benefit payments of the retired pension fund participants. In order to provide insights into the influence of investment returns on the received benefit payments of pension fund participants, 2000 economic scenarios were generated according to the parameters used for the "haalbaarheid-stoets" of the DNB as explained in 3.3. To give insights into the influence of investment returns on the received benefit payments of pension fund participants the 5th, 50th (median) and 95th percentile best economic scenarios (computed from the equity returns) are set out against the average received replacement rate per age. All pension fund participants pay a fixed percentage of their salary as premium regardless of the economic scenario. Pension fund participants bring a certain amount of capital dependent on their age to their pension balance when joining the Tontine pension fund, this is also not influenced by the economic scenarios.
The percentiles of the economic scenarios are determined by computing $\prod_{y=1}^{l} \tilde{e}(y)$ for each economic scenario. Where $\tilde{e}(y)$ is the equity return in year y, where l is the simulation length. The resulting values are in turn sorted from highest to lowest, after which the 5th, 50th (median) and 95th percentile best economic scenarios are chosen which are in turn used in the simulation.
Figure 4.11 demonstrates the volatility of the Tontine pension fund. In the economic scenario of the 95th percentile, a pension fund participant can get a replacement rate of over 4, this means that the pension fund participant receives more than 4 times its last earned salary. In the median scenario the replacement rate is varying from 1 to 2. In the economic scenario of the 5th percentile, pension fund participants will receive a replacement rate of less than 0.5.

Individual case

As previously explained, the Tontine pension is dependent on the investment returns. To illustrate the effect that the investment returns have on an individual pension fund participant and to see whether or not there is a difference in the influence of the investment returns on the benefit

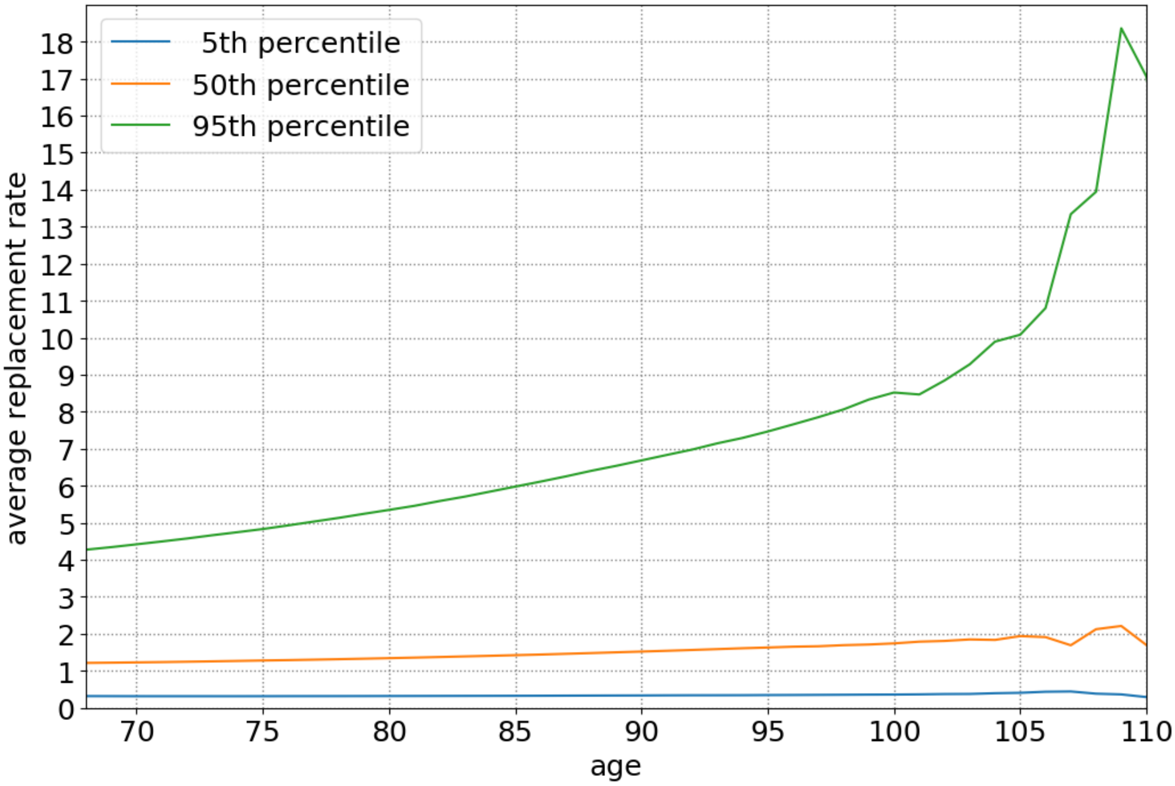

Figure 4.11: Evaluation of the influence of investment returns on average replacement rate. **Reading example:** This graph shows the replacement rate of pension fund participants of a certain age averaged over multiple simulation years given the different economic scenarios. Considering age 85, the replacement rates of pension fund participants of age 85 are registered every simulation year and averaged at the end of the simulation. In the economic scenario that corresponds with the 95th percentile, the average replacement rate of pension fund participant of age 85 are approximately 6, whilst in the 50th percentile it is approximately 1.4.

payments of pension fund participants joining on different ages, the benefit payments of three individual pension fund participant in three different economic scenarios are visualized in figure 4.12.

Figure 4.12 illustrates the volatility of the Tontine pension as currently modeled. The younger the age of joining the pension fund, the higher the volatility. That pension fund participants that join on a younger age experience a replacement rate that is more volatile due to investment returns is only logical considering that their premium payments have a longer time to be invested exposing them to investment risks for a longer period of time.

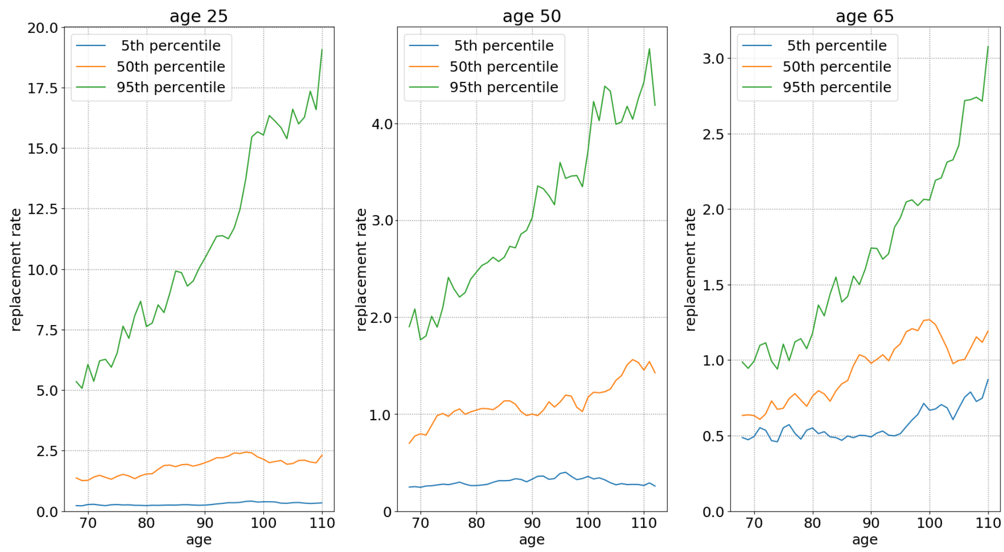

Figure 4.12: Evaluation of the influence of investment returns on replacement rates of a pension fund participants that join the pension fund on an age of 25, 50 and 65.
Reading example: These graphs show the replacement rate of pension fund participants that join the Tontine pension at age 25, 50 and 65 under different economic circumstances. Considering the left graph, a pension fund participant joins the Tontine pension at age 25, pays premium till the age of retirement and then starts to receive benefit payments till death. At age 100, the replacement rate in the scenario of the 95th percentile is higher than 15, whilst the replacement rate in the scenario of the 50th percentile is under 2.5.

4.4 Adverse selection

In this section the effects of adverse selection are discussed. In the Tontine pension fund, the aim is to allocate the mortality gains as fair as possible for all pension fund participants. This is done by heavily relying on death probabilities and survivor rates. In this study it is assumed that the death probabilities provided in these mortality tables are correct. It is unlikely that the population in a pension fund is perfectly homogeneous. This is why, as discussed in section 3.3, data from the CBS is used to distinguish between different salary groups with an accordingly adjusted death probability. In case someone wants to benefit from these differences in death probability, someone tries to join a pension pool that has a lower life expectancy so that this individual can benefit more from mortality gains. It is beyond the scope of this research to investigate how to prevent pension fund participants from exploiting a non-homogeneous group. This research will address the impact of treating a non-homogeneous group as a homogeneous group and how the received replacement rate differs from when a non-homogeneous group would be treated as a non-homogeneous group. This will be done by introducing participant classes as discussed in 3.3. The simulations are done as described in table 4.1.

In order to compare the influence of adverse selection a simulation will be run where a non-

Adjustment factor	A. Non-homogeneous group treated homogeneously	B. Non-homogeneous group treated non-homogeneously
Mortality gains	Middle salary class	Own salary class
Annuity factor	Middle salary class	Own salary class
Death events	Own salary class	Own salary class

Table 4.1: Performed simulations

homogeneous group will be treated as homogeneous (simulation A). This is done by computing mortality gains and annuity payments based on the middle income salary class, whilst the death events will be generated based on the salary class of each individual pension fund participant. This simulation data is then compared to simulation data where the mortality gains and annuity factor are computed based on the salary class of the pension fund participant and death events are generated based on the specific salary class (simulation B), wheres pension fund participants with a low income have a higher probability of death than the pension fund participants with a higher income. To analyze the difference in replacement rates between a pension fund as in simulation A to that of a pension fund as in simulation B, the replacement rates of 9 individuals are observed of pension fund participants joining the pension fund on ages 25, 50 and 65 belonging to a low, middle and high salary class. Where the low salary class means 66.67% of the average salary, middle means 100% of the average salary and high means 133.33% of the average salary. Figure 4.13 shows the replacement rates for nine pension fund participants that have joined the pension fund on different ages. The pension fund participants are of all modeled salary classes. This figure demonstrates that in a non-homogeneous pension pool that is treated like a homogeneous pension pool (simulation A), pension fund participants with a higher salary are favored whilst pension fund participants with a lower salary are disadvantaged in comparison to a "fair" pension fund where mortality gains and annuity payments of a non-homogeneous pension pool are computed with the class specific mortality gains and annuity payments. Treating a non-homogeneous pension pool as non-homogeneous yields the greatest benefits for pension fund participants with a lower salary class that have a lower life expectancy.

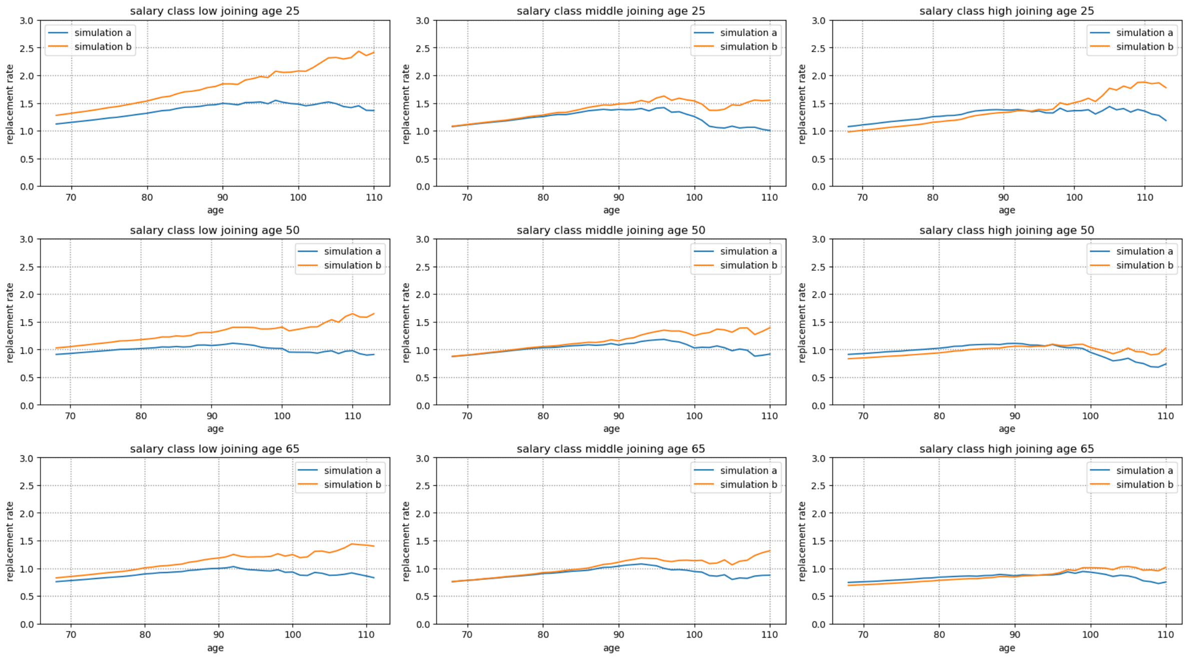

Figure 4.13: Replacement rates for nine individual pension fund participants with the age of joining the pension fund of 25, 50 and 65 from a low, medium and high salary class. Using simulation parameters as described in table 4.1.
Reading example: These graphs show the replacement rate of pension fund participants that join the Tontine pension at age 25, 50 and 65. In this graph the pension fund participants are placed in the salary classes low, middle and high. Based on the salary class the life-expectancy of the pension fund participants is adjusted. The Tontine pension can manage different pension fund participants with different probabilities of death by using these different probabilities for the allocation of mortality gains and for computing the annuities, so minimizing risk subsidization. For example, the graph in the top left shows a pension fund participant that joins the Tontine pension at age 25, pays premiums till the age of retirement and starts to receive benefit payments at the age of retirement. This pension fund participant is of a low salary class, meaning this pension fund participant has a lower life expectancy. The orange line labeled simulation b shows the replacement rates this pension fund participant would receive if the adjusted probabilities of death are used for the computation of the allocation of the mortality gains and the annuity payments, which is higher than the blue line labeled simulation a. The line labeled simulation b is higher than the line of simulation a, since if a pension fund participant in the Tontine pension has a lower life expectancy, the pension fund participant should receive higher benefit payments than someone with the same balance with a higher life expectancy.

4.5 The startup of a pension fund

In the previous sections of this chapter, the Tontine pension was analyzed using data from after the warm-up period using the procedure as described in section 3.4.1. The previous sections assume a constant number of pension fund participants and that pension fund participants bring a certain amount of capital on joining the pension fund based on their salary. In practice it is unreasonable to assume that from the start of a Tontine pension product it is filled with the ideal amount of pension fund participants from different ages. Therefore this section elaborates on how a starting Tontine pension product will influence the average replacement rate, comparing a different constant influx of pension fund participants. One of the assumptions in the previous analysis is that new pension fund participants join the pension fund with a certain amount of capital based on their salary, in practice, it may be the case that even older pension fund participants join the Tontine pension without bringing any initial capital. This section also elaborates on the difference in replacement rate in a starting pension fund where new pension fund participants either bring or do not bring initial capital. To analyze how a starting fund influences the replacement rate, data from simulations without a warm-up period, with a simulation length of a 100 years are used.

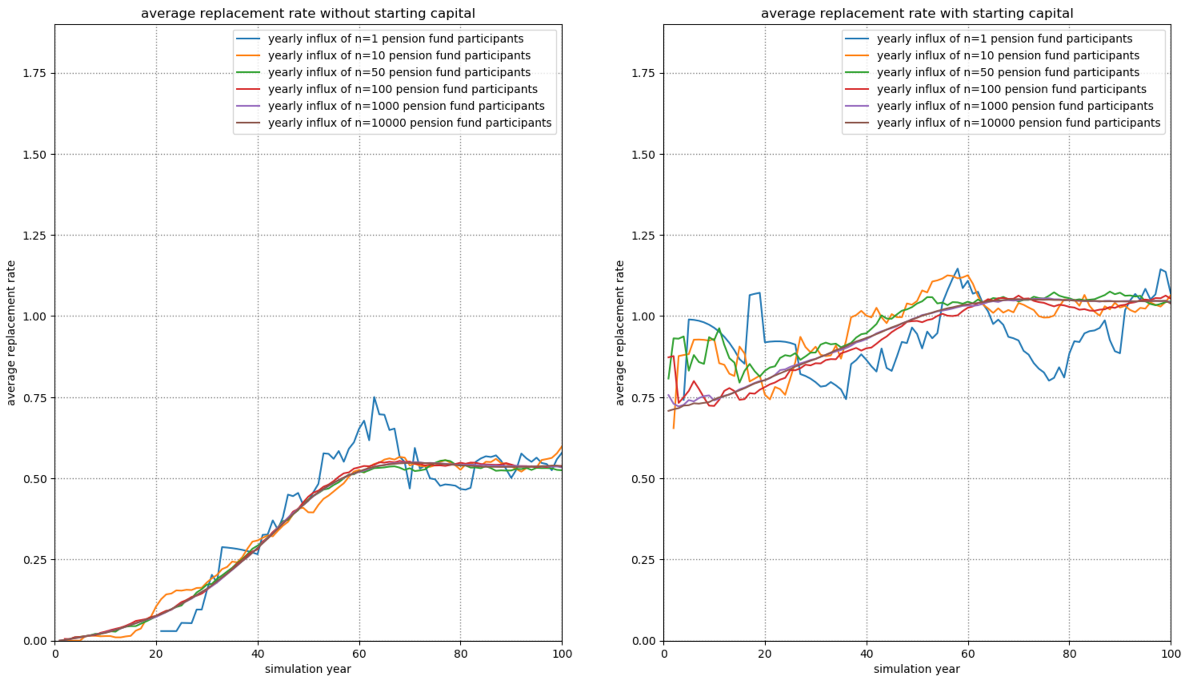

Figure 4.14: Average replacement rate over a simulation of 100 years using an annual influx of 1, 10, 50, 100, 1000 and 10000 pension fund participants.
Reading example: These graphs show the average replacement rate per simulation year in a starting Tontine pension fund. The left graph shows a starting Tontine pension fund where new pension fund participants do not bring an initial capital. In the left graph the new pension fund participants bring an initial capital based on their age and salary.

Figure 4.14 shows that in a starting Tontine pension pool, where people do not bring an initial capital based on their salary, people close at the age of retirement don't have to expect a high replacement rate, this is logical since it takes time for the pension fund participants to build capital. The average replacement rate does however seem to stabilize over time. In the simulation where new pension fund participants bring initial capital, the average replacement rate increases over time, it is likely that this is caused by the relatively positive economic scenario used for the simulation. The replacement rate in the scenario where new pension fund participants bring initial capital seems to stabilize over time just as in the scenario without initial capital.

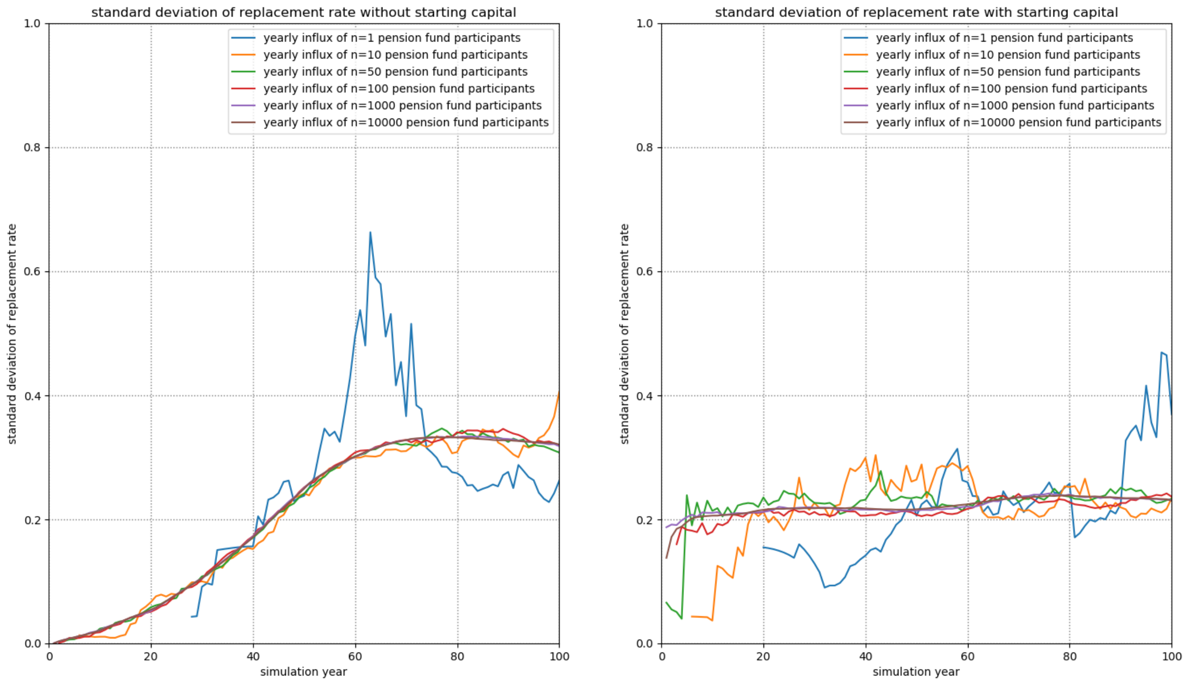

Figure 4.15: Standard deviation of replacement rate over a simulation of 100 years using with an annual influx of 1, 10, 50, 100, 1000 and 10000 pension fund participants.
Reading example: These graphs show the standard deviation of the replacement rate per simulation year in a starting Tontine pension fund. The left graph shows a starting Tontine pension fund where new pension fund participants do not bring an initial capital. In the left graph the new pension fund participants bring an initial capital based on their age and salary.

Figure 4.15 shows that regardless of if new pension fund participants bring initial capital, the higher the influx of pension fund participants, the more fluent the line of the standard deviation of the replacement rate becomes. In the scenario where pension fund participants do not bring initial capital, it seems that the standard deviation of the replacement rate increases in the first years of the simulation. This is logical as the pension fund participants first need to build capital in order to receive a higher replacement rate, as the average replacement rate increases, also the standard deviation increases, this does however seem to stabilize over time. Figure 4.15 shows that the scenario where pension fund participants bring initial capital is less volatile in general.

4.6 Increased life-expectancy

One of the challenges that current pension funds in the Netherlands have to deal with is that their pension fund participants get older whilst less people are born, this means that the pension costs increase whilst the number of pension fund participants paying for these costs decrease. In this section the influence of an increase in life expectancy on the average replacement rate is analyzed. Using the mortality tables as described in section 3.3, running the simulation from 2016 to 2186, taking into account the warm-up period as the procedure described in section 3.4.1. The simulation data shows that the average age of the pension fund participants increases as shown in figure 4.16. The influence of this increase in age on the average replacement rate is shown in

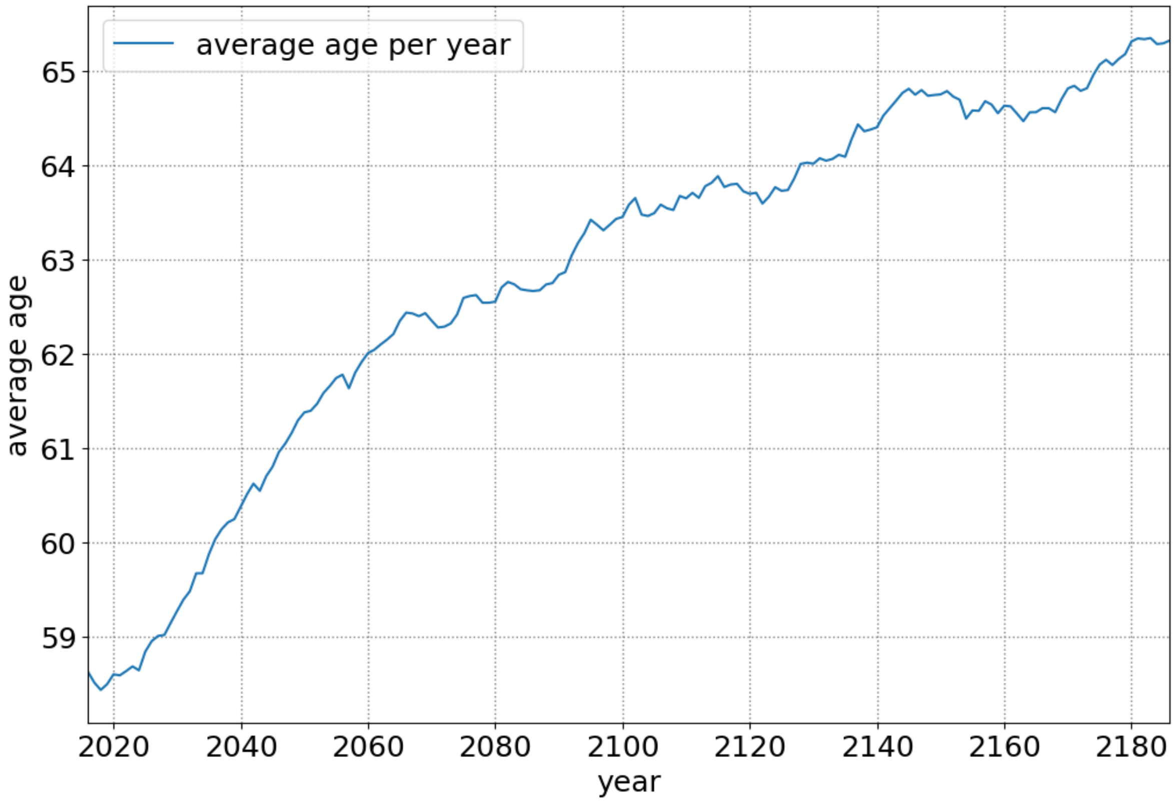

Figure 4.16: Average age of the pension fund in a pension fund with 10000 pension fund participants per simulation year.

figure 4.17, where the black bars show the average replacement rate plus and minus the standard deviation. Figure 4.17 demonstrates that as the average age of the pension fund participants increases, the average replacement rate decreases. Since in the Tontine pension fund the annuity payments and the mortality gains are computed based on the death probabilities, even with an increasing life-expectancy, the Tontine pension should remain fair for its participants.

If the pension age in the Tontine pension fund linearly increases over time, from for example 68 in 2016 to 76 in 2186, it is possible to keep the average replacement rate approximately

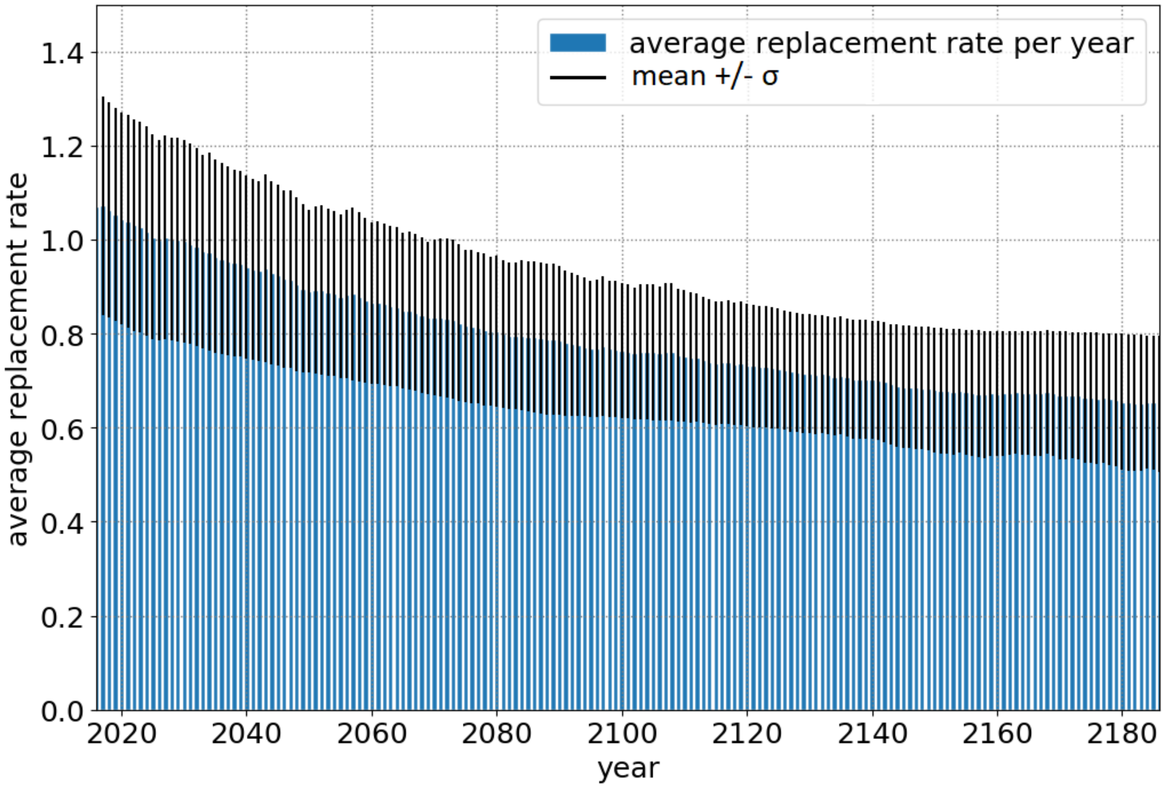

Figure 4.17: Average replacement rate of the pension fund in a pension fund with 10000 pension fund participants per simulation year, where the black bars show the average plus and minus the standard deviation of the replacement rate.

stable over time as figure 4.18 demonstrates, given the deterministic economic scenario of the simulation. It will however remain a trade-off between age of retirement and replacement rate. Offering pension fund participants a say in their own retirement age might offer a solution here, so that the decision of the trade-off lays at the pension fund participant. One of the problems with letting the pension fund participant decide this for themselves is that by choosing a low age of retirement, less premium payments are done, the annuity payments start earlier and thus the individual pension fund participant builds less capital, meaning that when a pension fund participant with a lower pension age deceases there are also less mortality gains to allocate, hence negatively influencing the average replacement rate of the fund as a whole.

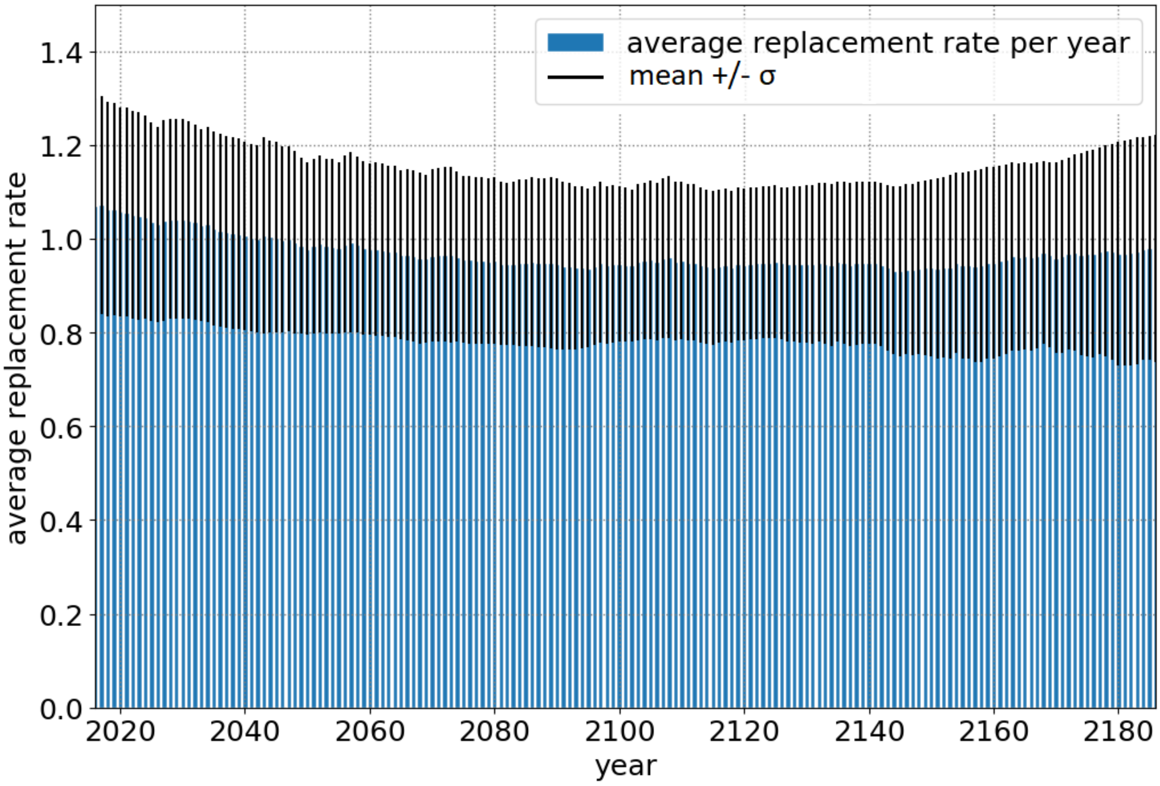

Figure 4.18: Average replacement rate of the pension fund in a pension fund with 10000 pension fund participants per simulation year, with a linearly increasing pension age from 68 to 77 over year 2016 to 2186, where the black bars show the average plus and minus the standard deviation of the replacement rate.

4.7 Draining pension fund

An important factor to take into account in a Tontine pension fund, is what happens if pension fund participants leave the pension fund. To analyze the influence of a draining Tontine pension fund on the replacement rate of the remaining individuals. The Tontine pension fund in this simulation starts with 10000 pension fund participants, the number of pension fund participants linearly decreases over time at different rates. The replacement rates of pension fund participants joining at an age of 25, 50 and 65 are studied with the use of the simulation model. The simulation model is first run till the steady-state is reached as explained in section 3.4.1, after which the three pension fund participants are added to the pension fund of age 25, 50 and 65 after which the number of pension fund participants is linearly reduced to three remaining pension fund participants at different rates.
Figure 4.19 shows the replacement rate of a pension fund participant joining the pension fund at an age of 25, 50 and 65. From the moment these pension fund participants join the pension fund pension fund, pension fund participants start leaving the pension fund at different rates. In

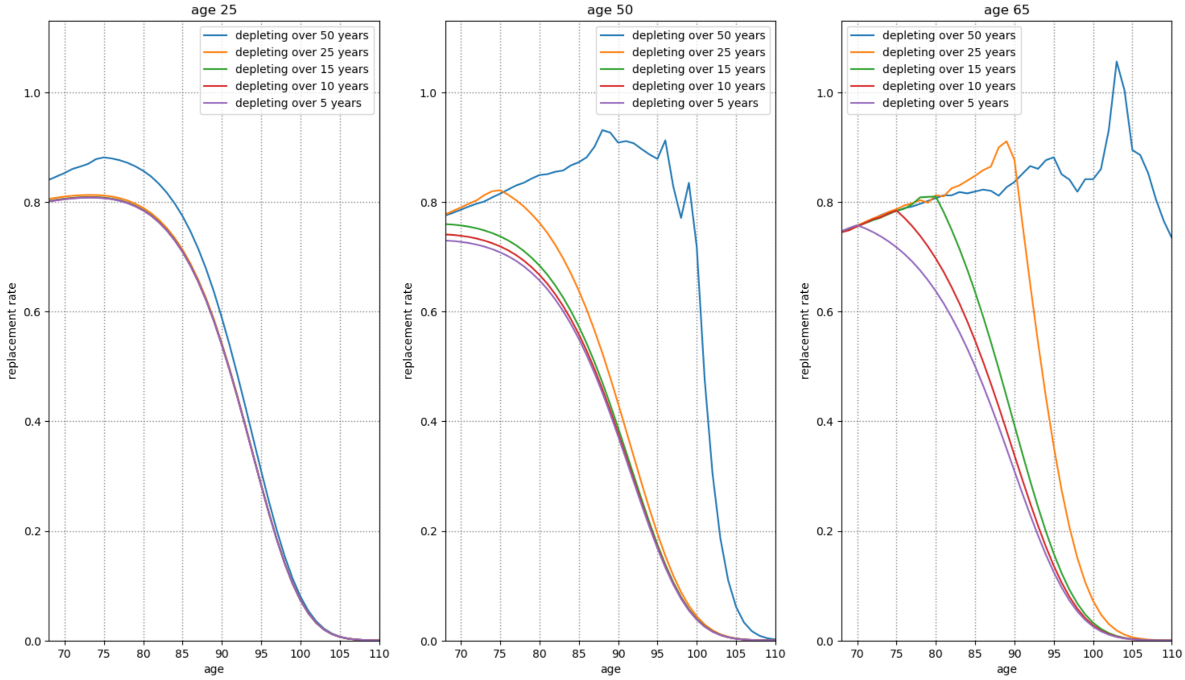

Figure 4.19: Replacement rates for pension fund participants joining at age 25, 50 and 65 where the pension fund population depletes linearly over time at different rates.
Reading example: These graphs show the average replacement rate for pension fund participants joining at age 25, 50 and 65. These three pension fund participants join at $t = 0$, after which the number of pension fund participants linearly reduces to the last three remaining pension fund participants at different rates. For example the right graph shows a pension fund participant that joined the Tontine pension at age 65, paid premium till the age of retirement and then starts to receive benefit payments. One can see that for example at age 90 the replacement rate for this participant is less when the number of pension fund participants reduce at a higher rate.

this simulation leaving the pension fund is not penalized. The faster the pension fund population depletes, the more impact it has on the received replacement rate. If the pension fund population depletes close to the age of retirement, the effects of the depletion rate on the received replacement rate will be more visible than when the pension fund depletes at a young age. Eventually as pension fund participants get older the received mortality gains and return on investments do not outweigh the benefit payments. Therefore the benefit payments will decrease over time and eventually converge to zero.

CHAPTER 5

BLOCKCHAIN APPLICATION

This chapter describes how the logic as described in chapter 4 is translated to a blockchain application and the challenges involved with this process. This chapter describes what kind of blockchain is used and why. This chapter also reflects on the technical requirements, limitations and involved costs. In this chapter subquestion 5 and subquestion 6 are answered.

5.1 Blockchain

In order to share longevity risk on the blockchain, it is important to create a trustworthy and transparent application which is able to scale and can sustain high traffic at a low cost. Creating a DAO is beyond the scope of the research, because it is not the aim to create a decentralized organization that is fully autonomous, instead the aim is to share longevity risk on the blockchain. To share longevity risk on the blockchain a decentralized application is a more logical choice since it doesn't makes the programmable logic overly complex.
Bitcoin is currently the best known cryptocurrency, however there are cryptocurrencies that use more mature blockchains. The platform of choice for creating a Tontine pension application on the blockchain in which longevity risk can be shared is Ethereum since Ethereum's underlying blockchain is most mature in terms of community and development tools and supports the execution of advanced programmable logic.

5.2 Functional design

The information in this section is confidential and therefore unavailable in the public version.

5.3 Smart contracts

The smart contract that contains the logic of the Tontine Pension fund is designed such that one can initiate a tonChain pension contract with an empty pool of pension fund participants. Subsequently users can become a pension fund participant by subscribing to the contract sending their identity, an initial deposit and an investment preference. Storing all logic in one smart pension contract has the benefit that no communication has to take place between different smart contracts on the blockchain, which is more efficient. In turn pension fund participants can pay premium to their individualized pension balance and change their investment preference to their desire. The pension fund participants can at any time see how much money their individualized pension balance is worth and in which assets their money is invested by simply calling a function from the smart contract.
The largest part of the work done in the smart contract is done by a contract function called update, which updates the entire smart contract in batch, allocating mortality gains, paying annuities and re-balancing investment preferences. These computations are done in batch because this helps making the contract more efficient and so cheaper to manage. Currently there is no way to build a trigger in a smart contract so that the smart contract does an automatic monthly update of the values, rather the update function has to be called manually. It is however possible to make the function logic so that it can only be called periodically, for example monthly. After the update function is executed, retired pension fund participants can withdraw their benefit payment from the individualized benefit balance in the pension contract.

5.3.1 Computational limitations

When writing the Tontine pension fund smart contract, some technical limitations were encountered. One of the first problems encountered when creating a smart contract in Solidity[1] is that it only permits integers and does not work with decimal values. This problem can be worked around by storing the values in the blockchain multiplied by a factor in the Tontine pension fund smart contract, all values are stored multiplied by a factor 10^9, for example the value 100.32 would be stored as 100320000000. This means that when computations are done, some rounding errors are made, but that these are relatively small. Solidity in which the smart contract is written also has no build in protection against the overflow and underflow of numbers. For example if one would subtract uint[2] 10 from uint 5, some arbitrary number will be returned because the result would be out of the domain that uint allows, if one would use integers in the same equation it would return -5. This problem can be solved by writing safeguards for doing computations or by using the SafeMath library [9] for doing computations. Another challenge involved with coding in

[1] Solidity is a contract-oriented programming language which can be compiled to bytecode. This bytecode can be interpreted by the Ethereum Virtual Machine.

[2] uint is a data type similar to integers, however a uint cannot be negative. The data type int can vary from –2147483648 to 2147483647, whilst uint can vary from 0 to 4294967295 on a 32-bit machine.

Solidity is that it is not possible to take the root of a number. Given the computational limitations and the consideration of code efficiency, it is not feasible to implement the fair mortality gains method as described in section 2.1.8, this is why the risk premium method is implemented for the Tontine pension fund smart contract as described in section 4.1.1, which is a way of allocating mortality gains very similar to the fair mortality gains method, but is less computationally complex.

5.3.2 Scalability

An important factor that needs to be taken into account when developing smart contracts on the blockchain is scalability. In general scalability is less of a problem in private blockchains than in public blockchains. For this study a smart contract is developed taken into account that it should be possible to deploy it on a public blockchain. This means that in programming the pension contract logic, coding efficiency is taken into account as much as possible. Deploying such a contract on a private chain should not lead to any complications.
The scalability of smart contracts is limited by the maximum amount of gas[3] that can be used for a single transaction. Meaning that the amount of computational power that one can use in a single transaction is limited. This gas limit can be increased over time.
Deploying the contract does not exceed the current gas limit. Storing additional pension fund participants in the tonChain contract does not require much computational power. Also depositing premium and withdrawing benefits require little computational power. The two functions used by the tonChain pension contract that require the most computational power are the allocation of the mortality gains and the annuity computations.
Allocating the mortality gains in the tonChain smart contract requires two loops over all pension fund participants and thus the computational costs of this function increases as more pension fund participants join the Tontine pension fund. Computing annuities for the retirees in the Tontine pension fund requires looping over all retired pension fund participants and dividing the individual balance of each retiree by the sum of the element wise multiplied survivor rate and interest term structures, which is a function that is relatively computationally intensive and requires more computational power as more pension fund participants retire. These functions can cause problems on the blockchain since as the Tontine pension fund gets a larger pool of pension fund participants and a larger number retired pension fund participants the functions may eventually exceed the gas limit that can be used for computations on the blockchain.
The most computationally intensive function of the tonChain smart contract is allocating the mortality gains. Allocating the mortality gains requires to loop over all pension fund participants. This means that the gas consumption of the smart contract scales with the number of pension

[3]Executing transactions on the Ethereum blockchain costs a certain amount of computing power. This computing power is called gas. It is possible to set a gas limit to restrict the maximum amount of processing power one transaction can perform. A higher gas price can be offered to make it more likely for a certain transaction to be executed.

Attribute \ Chain	Private	Public in 2016	Public in 2018
Cost of full update	fixed	\$0.05 to 0.15 p.p.	\$5.00 to 15.00 p.p.
Cost of contract deployment	fixed	\$6.50	\$650.00
Cost of 1 user subscription	fixed	\$0.20	\$20.00
Cost of 1 premium deposit	fixed	\$0.05	\$5.00
Cost of 1 allocation update	fixed	\$0.08	\$7.50
Yearly cost for 1 active user	fixed	\$3.30	\$330.00

Table 5.1: Overview of estimated costs per user, where a pension fund participant that is not retired changes its asset allocation each month whilst retirees do not. These estimated costs are based on an exchange rate of 1000 Dollar equals 1 Ether and the average gas price in January 2018 on the Ethereum blockchain.

fund participants. This can lead to problems on the blockchain since there is a upper limit on how much computational power can be done in one task.

The computational complexity of both allocating the mortality gains as computing annuity payments can be reduced by computing these values outside of the blockchain and then loading these data from an oracle, this would however require these oracles to be reliable.

In the pension contract as initially designed, it was infeasible to update the contract for more than approximately 25 users, because in that case the computations in the update function would exceeds the gas limit allowed in a single transaction. This problem was circumvented by updating users in batches using multiple transactions rather than in one large transaction.

5.3.3 Costs

The costs are a very important aspect to take into account when managing a Tontine pension fund on the blockchain. The tonChain smart contract is designed to function on the Ethereum public blockchain. Using the public instance of the Ethereum blockchain would mean that transaction fees are directly related to the Ether price, making the costs involved with managing such a smart contract very volatile. In 2016 the Ether price was just 10 dollar versus 1000 dollar in January 2018. If the consensus algorithm of the Ethereum blockchain would change, this is likely to influence the transaction costs significantly. A cost estimation based on the gas usage of the functions from the tonChain smart contract is shown in table 5.1.

Table 5.1 shows the volatility of transaction costs when using public Ethereum. As the price for computational power increases, managing a pension fund on the Ethereum blockchain becomes rapidly less attractive from an operational costs perspective. If one would choose not to use a public blockchain like Ethereum but rather a private blockchain, the transaction costs would become fixed and so low that these costs would become neglectable, the costs involved would in that case be the costs of running the server(s) that hosts a private blockchain, using a private blockchain would also allow for setting ones own rules in terms of maximum computational power per transaction and consensus algorithm.

5.4 Trust

One of the fundamental aspects of any smart contract is, that what is written in the code is the law and must be uphold at all times. Also smart contracts are immutable and transparent, which means that no one can change the rules of the smart contract and everyone knows beforehand what they are agreeing to. This means that a Dapp behaves in the way it is coded and if done properly does not require a trusted third party. This is an interesting paradigm for a Tontine pension fund managed on the blockchain, especially in areas with low trust in pension service providers. By managing a Tontine pension fund on the blockchain, one can assure that at a certain age the pension fund participant receives annuity payments based on the smart contract logic, investment rules e.g. can all be recorded in the blockchain. However current technology still doesn't allow for the management of assets directly on the blockchain and still requires trust from an intermediary. The same goes for the party providing mortality tables and actual mortalities of pension fund participants. This study did not focus on solving the problem of interaction with external sources on the blockchain and how to ensure that these sources are reliable, however these factors are still important to note since these factors are rather fundamental in creating a fully autonomous Tontine pension fund on the blockchain.

Chapter 6

Conclusion

In this chapter the insights gained from the previous chapters are used to answer the subquestions. The first four subquestions are answered by the insights obtained from chapter 4, the 5$^{\text{th}}$ and the 6$^{\text{th}}$ subquestions are answered in chapter 5. The subquestions are in turn used to answer the main question of this research.

6.1 Actuarial analysis

Based on the feasibility study conducted in chapter 4 it is possible to answer the first four subquestions of this study about the Tontine pension fund.

1. How can longevity risk be shared in a fair way in a pension fund using the Tontine principle?
2. How does the number of pension fund participants in the Tontine pension fund influence the benefit payments of its participants?
3. How volatile are the benefit payments in the Tontine pension fund given different economic scenarios?
4. How does the life expectancy influence benefit payments of its participants, in both homogeneous and non-homogeneous populations?

The first subquestion to find a fair way of sharing longevity risk in a Tontine pension fund can be answered by using either the fair mortality gains method or the risk premium method. The fair mortality gains method and the risk premium method have a way of allocating mortality gains that are very similar to each other and are both more fair than allocating mortality gains proportional to capital. The risk premium method as logically deduced in section 4.1.1 is easier to compute than the fair mortality gains method, especially as more pension fund participants join the Tontine pension fund. The computational complexity of the fair mortality gains method makes using this algorithm infeasible on the Ethereum blockchain. Because the risk premium method is both fair and fairly easy to compute, this is the preferred mortality gains allocation method. Another benefit of the risk premium method over allocating mortality gains proportional to capital is that with the risk premium method, the received benefit payment does not converge to zero over time, which is arguably more desirable.

The second subquestion regarding how the number of pension fund participants influences the benefit payments of its participants can be viewed from different perspectives. In chapter 4 a starting fund is analyzed, where pension fund participants join the Tontine pension fund over time, the minimum number of pension fund participants required for a stable average replacement rate is analyzed and what happens when pension fund participants leave the Tontine pension fund is analyzed, where leaving the Tontine pension fund is not penalized.

How many pension fund participants are required for a Tontine pension fund can be answered by the analysis in section 4.2. As one would assume, the larger the number of pension fund participants, the more stable the average replacement rate becomes and the lower the standard deviation of the replacement rate. The study implies that a pension fund with 1000 pension fund participants is more or less stable and that from a larger number than 10000 pension fund participants the increase in stability is marginal.

How a starting Tontine pension fund with an annual influx of new pension fund participants influence the replacement rate can be answered by the analyses from section 4.5. The analyses shows that the more people join the pension fund, the more stable it becomes. The analyses shows that the average received replacement rate gradually increases over time to a certain point, both when pension fund participants bring or do not bring initial capital, this can be explained by the deterministic economic scenario. The higher the influx of pension fund participants, the smoother this transition goes. For example if a yearly influx of one pension fund participant is assumed, the average replacement rate is still unstable after a simulation period of a 100 years, whilst if there is a yearly influx of over a 100 pension fund participants, the average replacement rate seems to be fairly stable, even while transiting to a higher average replacement rate due to the economic scenario. The influence of pension fund participants leaving the Tontine pension fund at different rates on the replacement rate can be answered using the analyses from section 4.7. This section describes how the outflow of pension fund participants influences the received replacement rate of the individual where the outflow of pension fund participants are considered at different rates. It is important to note that leaving the pension fund is not penalized in these analyses. The influence of the speed at which the participants leave the Tontine pension fund on the replacement rate of the individuals is stronger at a higher age. The more people leave, the less mortality gains an individual can expect and the closer the situation becomes to only saving for the individuals own retirement without the benefits of risk sharing. This means that

using the method of allocating mortality gains as described in section 4.1.1 the height of the benefit payments of the retirees eventually converges to zero as pension fund participants age, given that the number of pension fund participants remaining in the Tontine pension fund is sufficiently low. One can argue if it should be allowed for pension fund participants to withdraw their money from this pension scheme and if so, if this should be penalized and by what amount. If the decision is made, to let pension fund participants withdraw their money, depending on the penalty, it may be advisable to freely let the remaining pension fund participants withdraw their funds if the number of pension fund participants goes below a certain threshold. This threshold can be derived from the minimum number of pension fund participants required for a Tontine pension scheme as described in section 4.2.
The third subquestion regarding the economic volatility of a Tontine pension fund can be answered from the analysis in section 4.3. Since the Tontine pension fund as described in section 3.3 does not build a buffer in times of high economic returns, but rather pays benefit payments to pension fund participants based on their individual balance, survival rates and interest term structures, the benefits are highly volatile. If there is a higher investment return this directly reflects in the replacement rates, but this also happens in times of lower or negative investment returns. Using the economic scenarios as described in section 3.3, the average replacement rate per age varies as much as shown in figure 4.11. In a very positive economic scenario, like the 95th percentile scenario, the average replacement rate varies from 4 to over 18 depending on age. In an average economic scenario as the 50th percentile, the average replacement rate varies from approximately 1 to 2 depending on age. In a bad economic scenario like the 5th percentile, the average replacement rate is less than 0.5.
The fourth subquestion on how the life expectancy influence benefit payments of its participants, in both homogeneous and non-homogeneous populations, is answered by looking into how a general increase in life-expectancy influences the replacement rate as analyzed in section 4.6 and by analyzing the difference in replacement ratio between a non-homogeneous population treated as a homogeneous population and by treating a non-homogeneous population as a non-homogeneous population in section 4.4, discussing how to cope with adverse selection by using adjusted death probabilities.
In order to analyze whether or not a pension fund participant can benefit from having a different probability of death than that of the other pension fund participants, simulations are conducted where pension fund participants are classified in a low, medium and high salary class according to data from the CBS. Simulations were conducted with the parameters as specified in table 6.1. The simulation results of simulation A and simulation B are compared to each other. One can see that there is some difference in replacement rate. In simulation A, people with a higher probability of death typically have a lower replacement rate than in simulation B. In the middle salary class where the adjusted death probability of the pension fund participant is approximately the same as that of the original death probability, the replacement rate on a lower age is approximately the

Adjustment factor	A. Non-homogeneous group treated homogeneously	B. Non-homogeneous group treated non-homogeneously
Mortality gains	Middle salary class	Own salary class
Annuity factor	Middle salary class	Own salary class
Death events	Own salary class	Own salary class

Table 6.1: Performed simulations

same in both simulation A as simulation B, however on a higher age the pension fund participant has a slightly higher replacement rate. For pension fund participants of the high salary class, which is the class with the lowest probability of death, it is beneficial to join a pension fund where the pension fund is treated as homogeneous (simulation A). Especially at an early age the pension fund participant of salary class high has a higher replacement rate. So there is an incentive for people with a low probability of death to join a Tontine pension fund where in general the probability of death is higher, this problem of adversed selection can be overcome by computing the allocation of mortality gains and the annuity payments based on the adjusted death probabilities, however the challenge remains in collecting the correct probabilities of death and allocating pension fund participants to the right class with the correct probability of death. How the Tontine Pension Fund can handle an aging population can be answered from the analysis in section 4.6. As source for the death probabilities the data from the AG2016 tables are used as described in section 3.3. These tables contain the death probabilities used for the Tontine simulation model. The prediction of the death probabilities in this table go from the year 2016 to the year 2186. When doing a simulation over this time period, a clear increase in the average age of the pension fund participants takes place. Since the Tontine pension fund calculates the benefit payments based on the survival rate and an interest term structure, the only problem that occurs from an aging population is that the pension fund participants on average receive lower benefits. One can argue how to respond to this phenomena. One option is to do nothing and let pension fund participants receive lower benefit payments. An easy way to keep the average benefits at a constant rate with an aging pension fund population is by increasing the age of retirement. In the analysis described in section 4.6 the pension age is increased from 68 to 76 in a period from 2016 to 2186, in this situation the average received replacement rate is more or less stable. One can argue if this increase in pension age is realistic given the fact that it is highly likely that the mortality tables will change over time. It does however show that it is possible to use the pension age as a medium to keep the average replacement rate on a certain level.

6.2 Blockchain

Based on the information provided in chapter 5 it is possible to answer subquestion 5 and 6.

5. Is it possible to share longevity risk in a Tontine pension fund managed on an Ethereum blockchain?

6. What are the advantages and disadvantages of sharing longevity risk in a Tontine pension fund managed on an Ethereum blockchain given blockchains decentralized nature?

To answer the 5^{th} subquestion, yes it is possible to create a Tontine pension fund on the blockchain in which longevity risks are shared. Taken into account the computational limitations, longevity risk can be shared using the risk premium method as described in section 4.1.1. Because of the limited computational power a single transaction can contain it is important to update the data in the pension fund batch wise, such as allocating mortality gains, updating investment preferences etc. The costs are closely related to the Ethereum price. The computational power required for managing the Tontine pension scales approximately linearly as more pension fund participants join the pension fund. There are still more challenges to solve in order to make this Tontine pension fund a fully autonomous pension product managed on the blockchain such as actually investing in assets on the blockchain and ensuring that information from third parties are provided in a reliable way. To answer the 6^{th} subquestion, a Tontine pension product managed on the blockchain enables a transparent pension product which is accessible to anyone with an internet connection, it is sensory resistant and all rules of allocating mortality gains, premium payments and investment preferences can be stored in this pension product. Code is law, which means that everyone can know what they are agreeing to when making use of a Tontine pension product and as long as all logic is conducted on the blockchain, no trust is required in any third party. As it is accessible to anyone with an internet connection, it exceeds the borders of a single country. Blockchain technology is still maturing. As the technology matures, it is likely that solutions will be found for some of the current downsides of the technology such as limited computing power and transaction costs that vary with the Ether price. If the transactions cost on a public chain could be reduced to close to that of a private chain, this pension product would allow pension management for a fraction of its current costs, also allowing for people in low income countries to buy affordable pension products. The Tontine pension product also has some downsides, as the investment risks lay with the individual and the pension fund does not build buffers for periods of protracted low economic returns. If the pension product is deployed on the blockchain, the logic cannot be changed and it is available to anyone with an internet connection, this also means that governments/supervisors cannot govern this pension product, unless this is build within the programmable logic. Governments/supervisors not being able to govern this pension product can be seen as both an advantage as a disadvantage. A clear disadvantage of a blockchain managed pension product is that no technology is flawless and if there is a large sum of money in a Tontine pension product managed on the blockchain, it invites attackers to try to exploit the technology's weaknesses by for example a 51% attack, searching for hash collisions or by looking for exploits in the logic of the smart contract.

6.3 Main question

The conclusions of the subquestions are used to answer the main question:

"How can longevity risk be shared on the blockchain among pension fund participants using the Tontine principle?"

It is possible to share longevity risk on the blockchain in a Tontine based pension product using an application as described in chapter 5. The dynamics of such a pension product are vastly discussed in chapters 3 and 4. Before the Tontine pension product can be developed in a fully autonomous application or an autonomous organization, the technology needs to mature and tackle some challenges that were left out of scope in this study such as managing assets on the blockchain and ensuring that interaction with the blockchain goes in a reliable matter. Also as the technology evolves, it is likely for transaction costs to go down. Offering a Tontine pension product on the blockchain, enables everyone with an internet connection access to a pension product. Using blockchain technology offers everyone access to an affordable pension product, also in lesser developed countries as long as there is access to the internet. A pension product managed on the blockchain offers a pension product that is fair, transparent, trustworthy and where the pension rules cannot be intermediately changed. The characteristics of the blockchain offer a trustworthy pension product in a trustless environment.

CHAPTER 7

Discussion and future research

This chapter discusses the conducted research, the validity of the research and the limitations of the research. As the research consisted from an actuarial part, where a simulation model was developed and used to gain insights on the behavior of a Tontine pension product and a prototype of a Tontine pension product managed on the blockchain, these parts are discussed in separate sections, namely actuarial and blockchain application.

7.1 Actuarial

The actuarial part consists of two parts, the developed simulation model that simulates the behavior of a Tontine pension product and a feasibility study where the simulation model is used to generate data under different circumstances, which can be visualized and interpreted.

7.1.1 Simulation model

A simulation model is used to simulate the behavior of the Tontine pension product. Using a simulation model is a logical choice in this case, because experimenting with this Tontine pension product is costly and takes too long, as the actuarial analysis conducted on the generated data typically has a timespan of multiple years. In a simulation model, it is always a trade-off between the complexity of the model and the development time.
The interval of the simulated data generated by the simulation model is fixed and is on an annual basis. An alternative to this would be to choose a different timespan, or to use event driven simulation. Event driven simulation could add the benefit of being able to analyze the transaction behavior done in the Tontine pension fund, for example when pension fund participants actually do pay their contribution, how long it takes for this money to be invested according to their

life-cycle, etc. This was not taken into account in the simulation model, but this doesn't undermine the insights provided in the feasibility study, as these factors are beyond the scope of this study. The Tontine pension product has the potential of being able to let pension fund participants from different countries with different life-expectancies share longevity risk, the simulation model can use adjusted mortality probabilities, however it cannot incorporate multiple mortality tables and sample participants from different countries with different wealths. As this study is scoped to be limited to only the Dutch market, this causes no problems for the conducted research, however it could be interesting to incorporate this feature for further studies.
Inflation is taken out of consideration in the simulation model. This makes the results easier to interpret as they don't have to be corrected for inflation, however to give a more genuine reflection of reality this could be taken into account.
The Tontine pension fund participants in this simulation are replaced on death by different participants. This assumption is made so that the behavior of the Tontine pension fund can be studied more easily given a certain pension fund population. This is a simplification that makes this simulation model further relinquished from reality, since in practice, not all deceasing pension fund participants will be immediately be replaced, however not incorporating this does not undermine the purpose of the defined simulation model.
The defined Tontine pension product as described in section 3.3 does not build buffers to compensate pension fund participants in difficult economic times. This was a conscious decision, to keep the pension product transparent and simple. Section 4.3 clearly shows the economic volatility of this pension product. One could argue whether or not a pension fund participant needs to receive a benefit that is higher than the last earned salary. In practice it might be interesting to implement a damping algorithm that limits the benefits of the pension fund participants at a certain level and uses the surplus to compensate pension fund participants in times of protracted low economic returns.
The simulation model does not incorporate a spouse or children into its logic. This was intentionally left out of the scope of the product. One could argue whether or not it is desirable to incorporate a spouse, for example a survivor's pension, into the Tontine pension. As it is currently not included in the Tontine pension product, it is logical not to include it into the simulation model.

7.1.2 Feasibility study

The feasibility research consists of seven parts, the allocation of mortality gains, the minimum number of pension fund participants, the economic volatility, adverse selection, the startup of a pension fund, increased life-expectancy and a draining pension fund. All these parts will be discussed separately.

Allocation of mortality gains

In this research three methods for allocating mortality gains are studied. Allocating mortality gains proportional to capital, the fair mortality gains method and the risk premium method. Looking at the allocation of the mortality gains according to capital, it is fair to say that this method is not prefered as the benefit payments converge to zero over time and since younger pension fund participants are favored over older pension fund participants. The allocation of the fair mortality gains method is compared to the risk premium method, which look very similar, however in this study the effects of using the risk premium method are not compared to the fair mortality gains method using simulation data. One could argue about the relevance of this data as the fair mortality gains method is infeasible to implement on the Ethereum blockchain, but it might provide new insights on the matter.
The risk premium method as logically deduced in section 4.1.1 combined with the way of computing annuity payments as defined in section 3.3 makes pension fund participants of a higher age have a higher replacement rate. This phenomena is more desirable than a benefit payment that converges to zero over time, one could argue whether or not this way of allocating mortality gains and computing benefit payments should be altered so that on average the replacement rate is stable over time instead of increasing.
The replacement rate in this section is not compared to what pension fund participants would receive in comparison to current pension products given the same economic scenario. This comparison is not conducted because of the time it would require to build an alternative simulation model. It would be interesting to do this comparison so that one can see whether or not a Tontine pension product is more desirable than current pension schemes used in the Netherlands.
One can think of countless ways of allocating the mortality gains, however in this research only three types are compared. In order to keep the complexity of this research manageable, no different ways of sharing longevity risks on the blockchain are studied. The fair mortality gains method and the risk premium method both offer fair ways of sharing mortality risk, however it might be interesting for future research to study alternative ways of sharing risks on the blockchain, by for example using different calculations for sharing the mortality risks.

Minimum number of pension fund participants

The minimum number of pension fund participants required for a stable stream of benefit payments is analyzed for a pension fund population of 10, 100, 1000, 10000 and 100.000 pension fund participants. Assuming a static pension fund population is a simplification of reality, as in reality it is more likely that this number of pension fund participants would vary. In order to get a more realistic reflection of reality it would be interesting to let the number of pension fund participants vary according to a certain distribution with a certain stochastic outflow and influx of pension fund participants, rather than keeping the number of pension fund participants static and re-sampling them on death. In this study, using such distributions are left out of scope since it would require research on what kind of distribution should be used for an influx and outflow of pension fund participants for this dynamics to be realistic which is not in the scope of this research. It is important to note that new pension fund participants are sampled according to a probability distribution, but this is only done at model initiation and to replace deceased pension fund participants. In section 4.5 and section 4.7 the number of pension fund participants actually does change over time, however not according to a studied distribution but with an deterministic number of influx or outflow of pension fund participants.

Economic volatility

In order to analyze the economic volatility of the Tontine pension product, economic scenarios are used that are generated in a similar way as economic data used for the "haalbaarheidstoets" of the DNB. The economic scenarios generated are only of a limited timespan and are repeated over time as the simulation length exceeds the timespan of the generated economic data. Making these scenarios cyclic was one of the trade-offs made to make the simulation model less complex, however it does make the simulation model less realistic.

Adverse selection

The problem of adverse selection in the feasibility study is limited to looking into how one can deal with a non-homogeneous group of pension fund participants using mortality probabilities that are subgroup specific and how large the difference is from treating a non-homogeneous group as homogeneous. This study does not look if pension fund participants should be allocated to subgroups or what these subgroups should be. This study does show that it is possible to create subgroups and treat these subgroups in a fair way. The correctness of treating the pension fund population and the potential subgroups of a pension fund population in a fair way is highly dependent on the correctness of the mortality probabilities.

The startup of a pension fund

It is not realistic to assume that a pension fund starts with the ideal amount of pension fund participants at initialization. Which is why it is important to analyze how a starting pension fund behaves in terms of the replacement rate of its participants. In this study this is done with a deterministic number of pension fund participants joining the pension fund. To make the simulation data more realistic, it could be possible to let pension fund participants join with a stochastic number, or choosing an alternative way to the yearly linear increase.
The pension fund participants in the starting Tontine pension fund are sampled according to the same distribution as any other simulation, however in practice certain people might be more interested in joining such a pension product than others, changing the distribution of people joining the pension fund. The distribution of people joining the Tontine pension highly influences

the pension fund population and the received replacement rate. It may be interesting to study how different distributions of people joining the Tontine pension influence the received replacement rate and the development of the replacement rate over time. In order to give a good reflection of reality, research should be conducted on what kind of people are most interested in this Tontine pension product, so that the distribution of sampled participants can be changed accordingly, this is however out of the scope of this research.

Increased life-expectancy

The AG2016 table as used in the simulation model shows an increase in average age over time, thus an increased life-expectancy. This is a challenge current pension funds also have to cope with. In the feasibility study different methods are explored for dealing with this increase in life-expectancy, as simply doing nothing and by doing so lowering benefit payments or gradually increase the retirement age over time and by doing so keeping the benefit payments at approximately the same level. Increasing the retirement age might not be an ideal solution as this might evoke friction from the pension fund participants. Alternative options could offer solutions to the increase in life-expectancy, for example giving the pension fund participants some freedom in choosing a retirement age, or by demanding additional premium payments to ensure a certain level of replacement rate. These options would require additional research as this might offer challenges in terms of adverse selection or unfeasibly high premium payments. There may be more unexplored ways of dealing with an increase in life-expectancy that might be interesting for future research.

Draining pension fund

A Tontine based pension product requires a minimum number of pension fund participants in order for it to function as a good way of sharing longevity risk. Therefore it is very important to know what happens when pension fund participants start leaving the Tontine pension fund. In the feasibility study, the replacement rate of pension fund participants of different ages are studied as pension fund participants leave the Tontine pension at different rates. These rates in the feasibility study are deterministic, in reality it is however unrealistic to assume that these rates are constant. These constant rates do show the effect of pension fund participants leaving the pension fund on the remaining pension fund participants.

The pension fund participants leaving the Tontine pension fund are randomly sampled regardless of their characteristics, in practice the characteristics of a pension fund participant may actually influence the likelihood of such a participant leaving the Tontine pension fund.

A way of discouraging people for leaving the Tontine pension fund is by simply not allowing them to leave, or by penalizing leaving. The actuarial dynamics behind such a penalty are yet to be determined. In order to determine this penalty and what kind of influence this has on the pension fund participants, additional research is required.

7.2 Blockchain application

To test if it is possible to share longevity risk in a Tontine pension product managed on the blockchain, a prototype of this Tontine pension product is developed. Building a prototype containing all the logic and dynamics of such a Tontine pension product is a valid way for testing whether or not it is possible to share longevity risk in a Tontine pension fund on the blockchain. However for building this prototype some simplifications and assumptions are made.

For the Tontine pension product managed on the blockchain it is assumed that the infrastructure for communication with the outside world is facilitated and that this can be done in a reliable way. External sources are for example mortality tables, actual mortalities and the investments done by the investment manager. Solving these challenges of information provision is outside of the scope of this research. Still this information provision is a key element for a fully functional Tontine pension product managed on the blockchain.

Testing the Tontine pension product managed on the blockchain in this research has not been done with the use of real people and real money. Instead a testing environment was used that provides conditions equal to that of the Ethereum blockchain. In order to test the viability of a Tontine pension product used by pension fund participants, it is advisable to pilot this product with actual money or at least actual users, rather than testing it with hypothetical money and hypothetical pension fund participants, because when testing it with actual pension fund participants, unforeseen situations might occur.

The study discusses how currency risks reflect in the operational costs, however the possibilities of hedging these currency risks are not explored. In the build prototype, premium payments are done to an investment balance, which are in turn invested. When someone retires, assets are sold periodically and the value of these assets is transfered to a withdrawal account of which the retiree can withdraw money. Whenever value in the blockchain application is stored in actual Ether, it makes this value prone to currency risk. If it is undesirable to expose the pension fund participants to these currency risks, additional research should be conducted on how to hedge the risks of a varying Ether price.

BIBLIOGRAPHY

[1] *Blockchain moet pensioen goedkoper maken | het financieele dagblad*.
https://fd.nl/economie-politiek/1222557/blockchain-moet-pensioen-goedkoper-maken.
(Accessed on 12/07/2017).

[2] *Cryptocurrency market capitalizations | coinmarketcap*.
https://coinmarketcap.com/.
(Accessed on 12/20/2017).

[3] *Ethereum and transactions as balances and not utxo | crypto-news.net*.
https://www.crypto-news.net/ethereum-and-transactions-as-balances-and-not-utxo/.
(Accessed on 04/13/2018).

[4] *Ethereum white paper*.
https://github.com/ethereum/wiki/wiki/White-Paper.
(Accessed on 01/19/2018).

[5] *Federal information processing standards publication secure hash standard (shs)*.
https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf.
(Accessed on 07/02/2018).

[6] *Math forum: Ask dr. math faq: The birthday problem*.
http://mathforum.org/dr.math/faq/faq.birthdayprob.html.
(Accessed on 06/14/2018).

[7] *Model en modelparameters onderliggend aan scenarioset - open boek toezicht*.
http://www.toezicht.dnb.nl/3/50-233690.jsp#.
(Accessed on 01/12/2018).

[8] *Opbouw pensioenstelsel | pensioen | rijksoverheid.nl*.
https://www.rijksoverheid.nl/onderwerpen/pensioen/opbouw-pensioenstelsel.
(Accessed on 11/30/2017).

[9] *Openzeppelin/zeppelin-solidity.*
https://github.com/OpenZeppelin/zeppelin-solidity/blob/master/contracts/math/SafeMath.sol.
(Accessed on 04/09/2018).

[10] *Over bitcoin - bitcoin.nl.*
https://bitcoin.nl/over-bitcoin#blockchain.
(Accessed on 12/16/2017).

[11] *Pensioenfederatie herkent zich in voorlichting raad van state over doorsneesystematiek.*
https://www.pensioenfederatie.nl/actueel/nieuws/2017/openbaar/juli/pensioenfederatie-herkent-zich-in-voorlichting-raad-van-state-over-doorsneesystematiek.
(Accessed on 01/19/2018).

[12] *Proof of stake faq · ethereum/wiki wiki.*
https://github.com/ethereum/wiki/wiki/Proof-of-Stake-FAQ.
(Accessed on 04/13/2018).

[13] *Proof of work vs proof of stake: Basic mining guide - blockgeeks.*
https://blockgeeks.com/guides/proof-of-work-vs-proof-of-stake/.
(Accessed on 12/16/2017).

[14] *quorum-docs/quorum whitepaper v0.1.pdf at master · jpmorganchase/quorum-docs.*
https://github.com/jpmorganchase/quorum-docs/blob/master/Quorum%20Whitepaper%20v0.1.pdf.
(Accessed on 06/08/2018).

[15] *Understanding serenity, part 2: Casper - ethereum blog.*
https://blog.ethereum.org/2015/12/28/understanding-serenity-part-2-casper/.
(Accessed on 04/13/2018).

[16] *Website koninklijk actuarieel genootschap en actuarieel instituut - home - overzicht.*
https://www.ag-ai.nl/.
(Accessed on 12/01/2017).

[17] *Welke soorten pensioenregelingen zijn er? | je werknemerspensioen | afm consumenten.*
https://www.afm.nl/nl-nl/consumenten/themas/producten/pensioen/werknemerspensioen/soorten.
(Accessed on 11/30/2017).

[18] *Wet verbeterde premieregeling - open boek toezicht.*
http://www.toezicht.dnb.nl/2/50-235754.jsp#.

(Accessed on 07/04/2018).

[19] *What is hashing? under the hood of blockchain - blockgeeks*.
`https://blockgeeks.com/guides/what-is-hashing/`.
(Accessed on 12/12/2017).

[20] *Wie is apg - apg*.
`https://www.apg.nl/nl/wie-is-apg`.
(Accessed on 12/07/2017).

[21] F. CHYBALSKI AND E. MARCINKIEWICZ, *The replacement rate: An imperfect indicator of pension adequacy in cross-country analyses*, (2015).

[22] CPB, *Eindrapportage voor- en nadelen van de doorsneesystematiek*, (2013).

[23] A. DE WAEGENAERE, B. MELENBERG, AND R. STEVENS, *Longevity risk*, (2010).

[24] N. DRAPER, *A financial market model for the netherlands*.
`http://arno.uvt.nl/show.cgi?fid=136397`, 3 2014.
(Accessed on 01/12/2018).

[25] J. B. FORMAN AND M. J. SABIN, *Tontine pensions*, University of Pennsylvania Law Review, (2015), pp. 755–831.

[26] GROEIFABRIEK, *GroeiFabriek Innovation Manual*, APG Group N.V., Amsterdam, 2017.

[27] W. K. HANEVELD, M. H. STREUTKER, AND M. H. VAN DER VLERK, *Indexation of dutch pension rights in multistage recourse alm models*, (2009).

[28] T. HULSHOFF, H. DE RUITER, G. SIEGELAER, A. SLAGER, AND T. STEENKAMP, *Het ontwerp van een dc pensioenregeling*, (2015).

[29] P. JAYACHANDRAN, *The difference between public and private blockchain - blockchain unleashed: Ibm blockchain blog*.
`https://www.ibm.com/blogs/blockchain/2017/05/the-difference-between-public-and-private-blockchain/`.
(Accessed on 12/20/2017).

[30] A. M. LAW, *Simulation Modeling and Analysis (Int'l Ed)*, McGraw Hill Higher Education, 2014.

[31] S. NAKAMOTO, *Bitcoin: A peer-to-peer electronic cash system*, (2008).

[32] M. NOFER, P. GOMBER, AND O. HINZ, *Blockchain*, (2017).

[33] S. D. PROMISLOW, *Fundamentals of Actuarial Mathematics (Wiley Desktop Editions)*, Wiley, 2015.

[34] R. L. RANSOM AND R. SUTCH, *Tontine insurance and the armstrong investigation: A case of stifled innovation, 1868-1905*, The Journal of Economic History, 47 (1987), pp. 379–390.

[35] E. RIES, *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*, Currency, 2011.

[36] O. W. STEENBEEK, *Costs and benefits of collective pension systems*, Springer, Berlin New York, 2007.

[37] D. R. STINSON, *Cryptography: Theory and Practice, Third Edition (Discrete Mathematics and Its Applications)*, CRC Press, 1995.

[38] M. SWAN, *Blockchain: Blueprint for a New Economy*, O'Reilly Media, 2015.

[39] S. VERMEEND, *Blockchain - de technologie die de wereld radicaal verandert*, Einstein Books, 2017.

[40] G. WOOD, *Ethereum: A secure decentralised generalised transaction ledger*.
`http://gavwood.com/paper.pdf`.
(Accessed on 01/19/2018).