FACULTY
OF MATHEMATICS
AND PHYSICS
Charles University

MASTER THESIS

Bc. Kseniya Kuzminskaya

Acceleration of calculations in life insurance

Department of Probability and Mathematical Statistics

Supervisor of the master thesis: RNDr. Janeček Martin, Ph.D.
Study programme: Mathematics
Study branch: Financial and Insurance Mathematics

Prague 2017

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In Prague date 20.7.2017 signature of the author

Title: Acceleration of calculations in life insurance

Author: Bc. Kseniya Kuzminskaya

Department: Department of Probability and Mathematical Statistics

Supervisor: RNDr. Janeček Martin, Ph.D., Department of Statistics and Probability, VŠE

Abstract: One of the major issues in practical calculation in life insurance - typically value of liabilities, value of the company, pricing,... - is that the calculations run times are very long. This work should investigate possible approaches how to speed up the calculation (cluster analysis, flexing,...). If the results are positive, it can have crucial practical applications

Keywords: life insurance, stochastic interest rate scenarios, cash flow calculation, cluster analysis

I would like to thank my supervisor RNDr. Janeček Martin, Ph.D. for his valuable advice during my studies and dedication to helping me succeed. Thank you to my consultant RNDr. Branda Martin, Ph.D. for inspiration and knowledge.

Contents

Introduction

Currently, insurance companies take a significant amount of time to process liabilites and cash flow. This thesis is based on the technique of accelerating the valuation of said processes. The aim of this work is to calculate the future cash flow of the company faster with reasonable error.

Among the actuarial tasks belong risk management, internal model calculation, dynamic asset/liability management which requires calculations of company's future cash flow with a wide range of possible scenarios. Economic scenararios may include a variety of economic indicators, from inflation to credit risks. For every contract is calculated cash flow with some possible scenarios. Because there are many variables to project which create many calculations, this has become a timely process.

This thesis will show the work on the implementation of standard cash flow calculations compared to the usage of proxy function and cluster analysis. We will to simulate the interest rate scenarios using a CIR model. We want to show the result comparison between the standard technique, proxy function and cluster analysis. We will simulate the rates and chose random model points to show the different time achievements between each technique. This thesis was implemented in Wolfram Mathematica Software.

1. Valuation of Liabilities in Insurance Companies

1.1 General introduction

Life insurance is a protection against financial loss that, for example, would result from the premature death of an insured. The insurer makes contracts with policy holders and promises to pay a named benificiary the benefit in exchange for a premium, upon the death of an insured person or during the agreed period.

In this work, we will focus on Universal Life products (UL). UL is a hybrid life insurance policy which combines elements of term life insurance with an investment savings option. Premiums within an universal life insurance policy are broken down by the insurance and saving component which is known as the cash (capital) value. The success of an universal life insurance plan depends on the investments in the plan the insurer chooses and market performance.

Within this thesis, our focus will center on the policies where a claim benefit is paid in the case of death or maturity. Such insurance products are called Endowment ("smíšené pojištění") and are the most common type of Czech life insurance ([1], [2]).

1.2 Main Principles of Fair Value

One of the main issues for insurance mathematicians is to determine the value of the liabilities of policy contracts. To calculate the value of liabilities for insurance companies, there are a variety of methods to choose from. Examples of these methods are Statutory valuation approach, Embedded Value (EV) approach, and Fair Value (FV) approach, which can be found in [2].

In this work, we will focus mainly on fair value - stochastic approach. Using the fair value approach, the estimated liabiltity results are more precise and closer to real situation with more economic scenarios. Assumptions for calculation are used on the best estimate level. Lapses and all other expected events are included ([2]). The stochastic fair value approach can be understood as the expected present values of the stochastic simulations of future cash flows.

In 1999, the IASB released an insurance issues paper in which it defined "fair value" as ([1]):
"The amount for which an asset could be exchanged or a liability settled between knowledgable, willing parties in an arm's length transaction"

Calculation of annual cash flow during the year t is ([2]):

$$CF_t = (P_t - C_t - E_t)(1 + rfr_t) - Claims_t, \tag{1.1}$$

where

P_t	premium paid at the beggining of the year t,
C_t	commisions paid to agents at the beginning of the year t,
E_t	expenses paid at the beginning of the year t,
rfr	risk free interest rate related to policy year t,
$Claims_t$	benefit outflows assumed to be paid at the end of the year t.

The present value of such cash flows is

$$FV_t = \sum_{t=1}^{n} \frac{CF_t}{(1 + rfr_t)^t}, \tag{1.2}$$

where

FV_t Fair Value of liabilities at the end of the policy year t.

1.3 Assumptions

Here we will discuss the issues for the assumptions used in calculation.

All assumptions used for the cash flow projection are to be on the best estimate level, which is understood to be their expected value.

1.3.1 Mortality

Underlying mortality assumptions are based on mortality tables, which are statistical tables of expected annual mortality rates. Insurance companies use mortality tables in order to determine proper premiums and fees that must be charged to individuals seeking insurance.

The insurance companies should take into account their experience in last years for mortality assumptions. Mortality experience tables might be split according to sex and age of the insured person, as well as smoker status, type of policy, etc ([2]).

Information gathered on an individual is taken into account when under consideration for the insurance. A selected mortality table includes mortality data on individuals who have recently purchased life insurance. These individuals tend to have lower mortality rates than individuals who are already insured, due to the fact that they have most likely just passed certain medical exams required to obtain insurance.

1.3.2 Lapses

Lapses are the cancellation of coverage and it can be an important component for the pricing of long term cash flows. The policyholder is allowed to cancel his policy at any time. As well as mortality experience, the companies should take into account their recent and reliable experience.

Once a policy lapses, the insurer is not under any legal obligation to provide the benefits stated in the policy. The insurance company returns the savings component deducted by some surrender fee to the policyholder.

Lapses analysis is usually built according to the policy year of insurance, type of product or calendar year of the policy inception ([2]).

1.3.3 Commissions

Commissions are usually based upon the size of the policy the agent is selling (means the size of annual premiums) and by the type of product.

There are two forms of commission payments to life insurance agents: first year (or initial) commission payments and renewal commissions payments.

The initial commissions payment is a payment that is equal to a percentage of the total annual premium that will be made on the policy during the first policy year.

A renewal commission is a commission paid for a specific number of years after the first policy year. The number of years that a renewal is paid vary between the companies, but frequently it is a significant number of years.

There can be claw back provision, which allows companies to return some amount of money back from the agents due to the withdrawal by the insurer of the policy agreement. Usually it concernes the initial commissions during the first years.

1.3.4 Expenses

General and administrative expenses typically refer to expenses that are still insurred by the company regardless of whether the company produces or sells anything. Examples of expenses can be product advertisement, salaries, building rent etc.

Expenses can be split into initial, such as medical underwriting, or renewal, for example payments to company's accountants.

Administrative expenses are usually classified in the following way ([1]):

- α initial single expenses: These exist during the closure of the contract. They can be calculated as a percentage from the sum assured or the premium.

- β regular administrative expenses: These are annual renewal expenses during the life time of the policy. They are calculated as a percentage of the sum assured.

- γ premium payment expenses: These are calculated as a percentage of the premium for regular payment policy types.

The expenses increase in time due to inflation. We will consider increase in expenses as well.

1.3.5 Other Assumptions

Risk-free rate (rfr) should be used for all interest rate assumptions ([2]).

The risk-free rate represents the minimum return an investor expects for any investment. It is the theoretical rate of return of an investment with zero risk. Often for such a rate, the return yield of government bonds is used.

Under the fair value approach, the risk-free rate is used for discounting the future cash flow and for annual investment income adjusted with market value margin.

2. Acceleration techniques of calculations

2.1 Analytic function

The main principle of cash flow calculation is the following

$$CF_t = \sum_{j:\forall policies} P_t^{(j)} - C_t^{(j)} - E_t^{(j)} - Dths_t^{(j)} - Surr_t^{(j)} - Matur_t^{(j)}, \qquad (2.1)$$

where

$Dths_t$	is the outflow representing the death benefit assumed to be paid at the end of the policy year t,
$Surr_t$	is the outflow representing surrenders assumed to be paid at the end of the policy year t,
$Matur_t$	the outflow representing the maturity benefits assumed to be paid at the end of year t.

Policy cash flows for Endowment product are shown in the Table 4.3.
Our aim for every scenario is to dervive the formula in the following form:

$$CF_t = \sum_{j:\forall policies} fix\mathrm{CF}_t^{(j)} + \sum_k Coef_t^k \cdot f_t^{(k)}(i_1, i_2, \dots, i_t),$$

where

$fix\mathrm{CF}_t$, $Coef_t$	doesn't depend on interest rate and can be calculated from the one run of the full model,
f_t	is a function of interest rate that is common for all model points,
k	is a number of $Coef_t$ and f_t pair, typically more than 1.

For our derivation let's start from a simple situation. We will calculate the cash flow from one contract during one year. For the type of product in this thesis, the benefit in the case of death or maturity is equal to the sum assured plus the savings part as a value of fund (or capital) at the year of payment t ($SA + CV_t$). In the case of a lapse, the company pays the policyholder the saving part deducted by the surrender fee. So, we have:

$$\begin{aligned}
CF_1 &= l_0(P_1 - C_1 - E_1) - CV_1(1 - fee)w_1 - (CV_1 + SA)d_1 - (CV_1 + SA)m_1 = \\
&= l_0(P_1 - C_1 - E_1) - CV_1[w_1(1 - fee) + d_1 + m_1] - SA(d_1 + m_1) = \\
&= l_0(P_1 - C_1 - E_1) - (CV_0 + P - \alpha - \beta - \gamma - RP_1)(1 + i_1) \\
&* [w_1(1 - fee) + d_1 + m_1] - SA(d_1 + m_1) = \\
&= l_0(P_1 - C_1 - E_1) - SA(d_1 + m_1) - CV_0[w_1(1 - fee) + d_1 + m_1](1 + i_1) - \\
&- SP_1[w_1(1 - fee) + d_1 + m_1](1 + i_1).
\end{aligned}$$

where

l_t	number of policies in-force at the end of year t, $l_t = l_{t-1} - d_t - w_t - m_t$;
d_t	number of deaths at the end of the year t, $d_t = l_{t-1} q_x$;
w_t	is number of lapses during the policy year t, $w_t = l_{t-1}(1-q_x) wthd_t$;
m_t	number of maturities at the end of the year t, $m_t = 0$ for $t < n$, $m_t = l_{t-1} - d_t - w_t$ for $t = n$;
SP_t	$(= P - \alpha - \beta - \gamma - RP_1)$ saving premium;
PR_t	risk part of premium;
CV_t	fund value at the end of year t;
fee	surrender fee applied when the surrender is paid assumed to be at the end of the year.

Using the same logistics, we can continue with the value of cash flow for the second year:

$$\begin{aligned} CF_2 = \cdots = {} & l_1(P_2 - C_2 - E_2) - SA(d_2 + m_2) - \\ & - CV_0[w_2(1 - fee) + d_2 + m_2](1 + i_1)(1 + i_2) - \\ & - SP_1[w_2(1 - fee) + d_2 + m_2](1 + i_1)(1 + i_2) - \\ & - SP_2[w_2(1 - fee) + d_2 + m_2](1 + i_2) \end{aligned}$$

And for the third year we will have:

$$\begin{aligned} CF_3 = \cdots = {} & l_2(P_3 - C_3 - E_3) - SA(d_3 + m_3) - \\ & - CV_0[w_3(1 - fee) + d_3 + m_3](1 + i_1)(1 + i_2)(1 + i_3) - \\ & - SP_1[w_3(1 - fee) + d_3 + m_3](1 + i_1)(1 + i_2)(1 + i_3) - \\ & - SP_2[w_3(1 - fee) + d_3 + m_3](1 + i_2)(1 + i_3) - \\ & - SP_3[w_3(1 - fee) + d_3 + m_3](1 + i_3) \end{aligned}$$

We will denote the fixed part of the cash flow at the year t for the policy j as

$$Fix\mathrm{CF}_t = l_{t-1}(P_t - C_t - E_t) - SA(d_t + m_t)$$

The summurization of all the model points will accumulate the cash flow at the year t for all model points which we are getting the formula for "proxy" or analytic function:

$$\begin{aligned} CF_t = & \sum_{j:\forall policies} Fix\mathrm{CF}_t^{(j)} - \\ & - \sum_{j:\forall policies} [(d_t^{(j)} + m_t^{(j)} + w_t^{(j)} \cdot (1 - fee)) \cdot CV_0^{(j)}] \cdot \prod_{k=1}^{t} (1 + i_k) - \\ & - \sum_{l=1}^{t} \sum_{j:\forall policies} [(d_t^{(j)} + m_t^{(j)} + w_t^{(j)} \cdot (1 - fee)) \cdot SP_l^{(j)}] \cdot \prod_{k=l}^{t} (1 + i_k) \end{aligned} \tag{2.2}$$

The estimation of cash flow by proxy function is a little bit tricky and requires patience and concentration. Yet, the final results of the estimation by proxy function are fast and concluded with no significant errors.

The calculation process by the analytic function can be summarized as the following [2], [3]:

1. Run the full model once. This run doesn't depend on investment return.
2. Derive the coefficients $Fix\mathrm{CF}_t$ and $Coef_t^{(k)}$ and save them (usually about 100 ths or more variables based on the product complexity)
3. Take the selected scenario of interest rates (i_1, i_2, ..., i_n)
4. Calculate $CF_t = \sum_{j:\forall policies} Fix\mathrm{CF}_t^{(j)} + \sum_k Coef_t^{(k)} \cdot f_t^{(k)}(i_1, i_2, \ldots, i_t)$

2.2 Cluster analysis

In this section we will focus on cluster analysis.

The purpose of clustering is to allocate observations of varaibles into homogenous and distinct groups ("clusters"). That means that observations are similar to each other within the group and different from observation in other groups ([4]). Our aim is to group the policies into model points and choose the representant of selected group policies. For each representant we have a scale, depending on how many policies it represents.

We don't have any assumptions about the distibution of the underlying data. Using the cluster analysis we are able to form groups of related observations. Clustering techniques can be used for any data set. All that is needed is a measure of how far one element in the set is from another element, using the function that gives the distance between the elements. The distance function f satisfies the following ([5]):

- $f(e_i, e_i) = 0$,
- $f(e_i, e_j) \geq 0$,
- $f(e_i, e_j) = f(e_j, e_i)$.

where e_i is ith element of the dataset.

We don't have an assumption about the number of clusters in which model points can be grouped. We will use non-hierarchical clustering method, called the k-means method. This method can be described in the following steps ([6],[7]):

1. Specify the number of clusters and the elements of each cluster. It can be choosen arbitarily or deliberatly. One of possible option is $K = \left\lfloor \sqrt{\frac{N}{2}} \right\rfloor$
2. Calculate each cluster's centroid, and the distances between each observation and each centroid. If the observation is nearer the centroid of a cluster other than the one to which it currently belongs, re-assign it to the nearest cluster

3. Repeat step number 2 until all observations are nearest to the centroid of the cluster to which it belongs

4. If the number of clusters cannot be specified with confidence in advance, repeat steps 1 to 3 with a different number of clusters and evaluate the results.

The big disadvantage of such method is that it depends on the order choice, which is used for grouping and this can cause different cluster results each time.

Mathematica software uses the algorithm of k-means. The algorithm for clustering N data points into K disjoint subsets S_j containing N_j data points so as to minimize the sum-of-square criterion [7]

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} |e_n - C_j|^2, \tag{2.3}$$

where e_n is a vector representing the nth data point and C_j is the geometric centroid of the data points in S_j

3. Interest rate scenarios

3.1 Models of interest rates

A zero-coupon bond is a contract promising to pay a certain "face" amount, which we take to be 1, at a fixed maturity date T ([8]). Prior to that, the bond makes no payments. Let's denote the price of the bond at time t as $P(t, T)$ with maturity T. From the definition is clear that $P(T, T) = 1$.

$$P(t, T) = \mathrm{e}^{-R(t,T)\cdot(T-t)},$$

where $R(t, T)$ is the continuisily compounding interest rate between times t and T.

Equivalently we can write

$$R(t, T) = -\frac{\ln P(t, T)}{T - t}, \quad t < T.$$

Instantaneous rate or short rate at time t is called the limit value of $R(t, T)$ for $T \to t$

$$r_t = \lim_{T \to t+} R(t, T) = \lim_{T \to t+} \left(-\frac{\ln P(t, T)}{T - t} \right) = -\frac{\partial}{\partial T} \ln P(t, T).$$

One way of how to model the term structure of interest rates is to model short rate process r_t. We assume that this process follows a stochastic differential equation (SDE). The general form of the model ([9]) is

$$\mathrm{d}r_t = \mu(r_t, t)\mathrm{d}t + \sigma(r_t, t)\mathrm{d}W_t.$$

Here $\mu(r_t, t)$ is called the drift, $\sigma(r_t, t)$ is diffusion and W_t is a Wiener process which is defined as ([10]):

1. $W_0 = 0$;
2. W has continuous paths a.s.;
3. $\forall 0 = t_0 < t_1 < \cdots < t_m$ the increments $W(t1) - W(t_0), \ldots, W(t_m) - W(t_{m-1})$ are independent;
4. $W(t + u) - W(t) \sim N(0, u)$.

In the scope of this thesis, we will use the Cox-Ingersoll-Ross (CIR) for interest rate modelling. It is given with the following stochastic differential equation ([11]):

$$\begin{aligned} \mathrm{d}r_t &= \alpha(\mu - r_t)\mathrm{d}t + \sigma\sqrt{r_t}\mathrm{d}W_t, \\ r(0) &= r. \end{aligned} \tag{3.1}$$

where $\theta = (\alpha, \mu, \sigma)$ are model parameters. The drift function $\mu(r_t, \theta) = \alpha(\mu - r_t)$ is linear and it has a mean reverting property. It means that interest rate r_t moves in the direction of its mean μ with speed α.

The advantage of 3.1 is that the interest rate does not become negative. If r_t reaches zero, the term multiplying $\mathrm{d}W_t$ vanishes and the positive drift term in 3.1 drives the interest rate back into positive territory ([8]).

From [9] and [12] we can write the formulas for conditional expected value and variance:

$$\begin{aligned}
\mathbb{E}[r_t|r_s] &= \mu + \mathrm{e}^{-\alpha(t-s)}(r_s - \mu),\\
Var[r_t|r_s] &= r_s\frac{\sigma^2}{\alpha}\left(\mathrm{e}^{-\alpha(t-s)} - \mathrm{e}^{-2\alpha(t-s)}\right) + \mu\frac{\sigma^2}{2\alpha}\left(1 - \mathrm{e}^{-\alpha(t-s)}\right)^2.
\end{aligned}$$

We will follow the notation given in [11] and [9]. Given r_t at time t the density of $r_{t+\Delta t}$ at time $t + \Delta t$ is

$$p(r_{t+\Delta t}|r_t;\theta,\Delta t) = c\mathrm{e}^{-u-v}\left(\frac{v}{u}\right)^{q/2} I_q(2\sqrt{uv}), \tag{3.2}$$

where

$$\begin{aligned}
c &= \frac{2\alpha}{\sigma^2(1 - \mathrm{e}^{-\alpha\Delta t})},\\
u &= cr_t\mathrm{e}^{-\alpha\Delta t},\\
v &= cr_{t+\Delta t},\\
q &= \frac{2\alpha\mu}{\sigma^2} - 1,
\end{aligned} \tag{3.3}$$

and $I_q(2\sqrt{uv})$ is the modified Bessel function of the first kind of order q and is defined as ([9]):

$$I_q(z) = \left(\frac{z}{2}\right)^q \sum_{k=0}^{\infty} \left(\frac{z}{2}\right)^{2k} \frac{1}{k!\Gamma(q+k+1)},$$

and $\Gamma(x)$ is gamma function.

The distribution function is the non-central $\chi^2(2cr_t; 2q+2, 2u)$ with $2q+2$ degrees of freedom and parameter of noncentrality $2u$ proportional to the current spot rate ([12]).

3.2 Calibration

In this part we will apply our model for certain particular data. We want to estimate parameters (α, μ, σ) of the model CIR 3.1 that it can fit our market historical data.

We will use the methods of ordinary least squares and maximum likelihood estimation ([9]). We will denote the market observations as $r_{t_i}, i = 0, \ldots, N (N \in \mathbb{N})$.

3.2.1 Ordinary Least Squares

We will take the discretization of CIR model 3.1 ([9]):

$$
\begin{aligned}
r_{t_i} &= r_{t_{i-1}} + \alpha(\mu - r_{t_{i-1}})\Delta t_i + \sigma\sqrt{\Delta t_i r_{t_{i-1}}}\epsilon_{t_i},\\
\Delta t_i &= t_i - t_{i-1}, \quad i = 1, \dots, N.
\end{aligned}
$$

We assume that the observations are equidistant and equivalently we can write:

$$
\begin{aligned}
r_{t_i} - r_{t_{i-1}} &= \alpha(\mu - r_{t_{i-1}})\Delta t + \sigma\sqrt{\Delta t r_{t_{i-1}}}\epsilon_{t_i},\\
\frac{r_{t_i} - r_{t_{i-1}}}{\sqrt{r_{t_{i-1}}}} &= \frac{\alpha\mu\Delta t}{\sqrt{r_{t_{i-1}}}} - \alpha\sqrt{r_{t_{i-1}}}\Delta t + \sigma\sqrt{\Delta t}\epsilon_{t_i}
\end{aligned}
$$

where ϵ_t is normally distributed with zero mean and variance Δt ([11]). The initial drift estimates can be found by minimizing the residual sum of squares [9], [11]

$$
(\hat{\alpha}, \hat{\mu}) = \underset{\alpha,\mu}{\operatorname{argmin}} \sum_{i=1}^{N} \left(\frac{r_{t_i} - r_{t_{i-1}}}{\sqrt{r_{t_{i-1}}}} - \frac{\alpha\mu\Delta t}{\sqrt{r_{t_{i-1}}}} - \alpha\sqrt{r_{t_{i-1}}}\Delta t \right)^2. \tag{3.4}
$$

The explicit formula for $\hat{\alpha}$ and $\hat{\mu}$ can be found in [11]. Volatility estimate is the following ([9]):

$$
\hat{\sigma} = \frac{1}{\sqrt{\Delta t}} \sqrt{\frac{\sum_{i=0}^{N} \left(\frac{r_{t_i} - r_{t_{i-1}}}{\sqrt{r_{t_{i-1}}}} - \frac{\hat{\alpha}\hat{\mu}\Delta t}{\sqrt{r_{t_{i-1}}}} - \hat{\alpha}\sqrt{r_{t_{i-1}}}\Delta t \right)^2}{N}}. \tag{3.5}
$$

3.2.2 Maximum Likelihood Estimation

We will define the likelihood function ([11])

$$
L(\theta) = \prod_{i=1}^{N} p(r_{t_{i+1}} | r_{t_i}; \theta, \Delta t),
$$

where p is density defined in 3.2. The corresponding log-likelihood function is:

$$
l(\theta) = \ln L(\theta) = \sum_{i=1}^{N} \ln\, p(r_{t_{i+1}} | r_{t_i}; \theta, \Delta t)
$$

After substitution from 3.2 we can derive the log-likelihood function ot the CIR process:

$$
l(\theta) = N \ln c + \sum_{i=1}^{N} \Big(- u_{t_i} - v_{t_i} + \frac{1}{2} q \ln \left(\frac{v_{t_i}}{u_{t_i}}\right) + \ln\, I_q(2\sqrt{u_{t_i}} v_{t_i}) \Big), \tag{3.6}
$$

where

$$u_{t_i} = cr_{t_{i-1}} \mathrm{e}^{-\alpha \Delta t},$$
$$v_{t_i} = cr_{t_i}.$$

We can find maximum likelihood estimates of $\hat{\theta}$ of vector $\theta = (\alpha, \mu, \sigma)$ by maximizing the log-likelihood function 3.6 ([11]):

$$\hat{\theta} = (\hat{\alpha}, \hat{\mu}, \hat{\sigma}) = \underset{\theta}{\operatorname{argmax}}\, l(\theta).$$

During the estimation we will take into account the positiveness of the parameters.

4. Implementation

In this part we will apply in practice our knowledge from previous chapters.

In the scope of this thesis we are using the UL Endowment product. In case of death or maturity, the insurance company pays the policyholder the agreed amount (sum assured) plus the value of the fund. As the first step we will simulate interest rate scenarios that will be used for investment return and for discounting. Secondly, we will set the assumptions for cash flow calculation. Using the simulated interest rates, we will calculate the present value of cash flows for some set of policy contracts (model points) in three ways. First, by using standard "policy-by-policy" cash-flow, then using the "proxy" function, and finally, using the cluster analysis. In the end we will compare the results and the final time of calculation.

We will run all our calculations in Wolfram Mathematica software. It is used in many scientific, mathematical and computing fields.

4.1 Interest rate models

For risk-free rate assumption, we will use the data of 1Y yield curve for goverment zero-coupon bonds for the Euro area. Data is taken from the 1.1.2016 and till 31.12.2016 and the number observations that we have in our history is 273 ([13]).

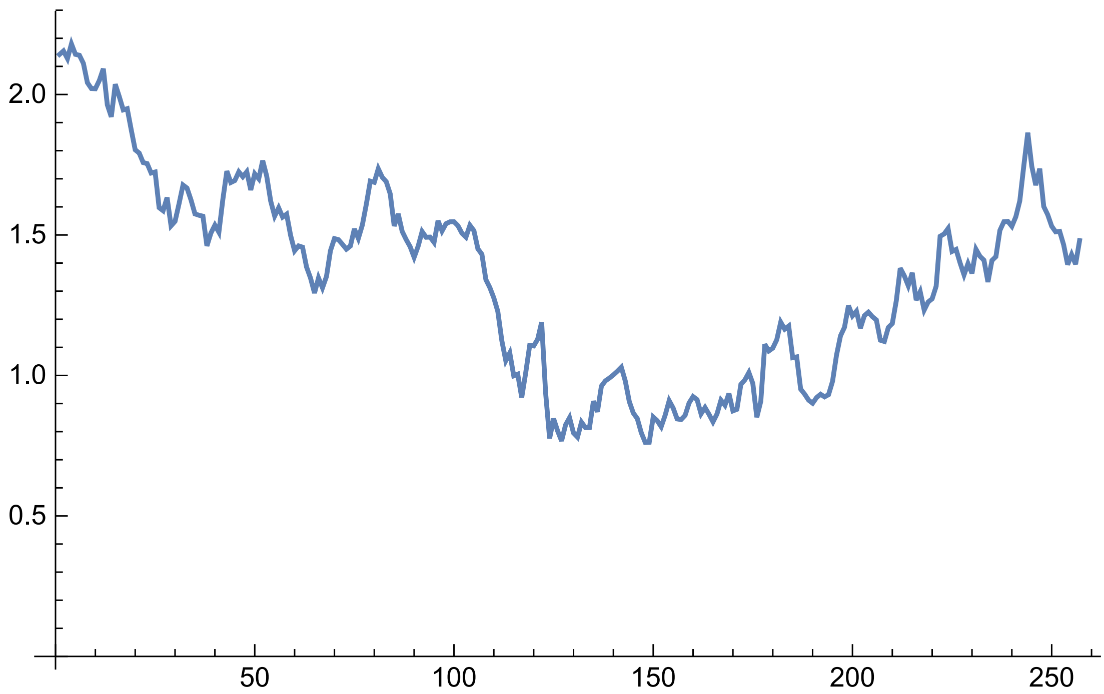

Figure 4.1: Market data of yield curve for goverment bonds for the Euro area

We will start with OLS method and then we will use the initial estimates for MLE method. For simplicity, we will have $\Delta = 1$. By using the formula above 3.4 we will get the estimates

$$(\hat{\alpha}_{\text{OLS}}, \hat{\mu}_{\text{OLS}}) = (0.023, 1.237)$$

And from 3.5 we have:

$$\hat{\sigma}_{\text{OLS}} = 0.051$$

Then we will use these esimations as initial in numerical maximization of the log-likelihood function 3.6 and the we will have the following results:

$$(\hat{\alpha}_{\text{MLE}}, \hat{\mu}_{\text{MLE}}, \hat{\sigma}_{\text{MLE}}) = (0.022, 1.232, 0.052)$$

The value of log-likelihood function is $l(\theta) = 362.3$. We can see that in this case there is no significant differences between the results of OLS and MLE methods. As final estimators of the parameters we will use the MLE estimators.

For the estimation of parameters with the MLE method, we took into account the positivenenss of the parameters. From [11] we have the condition that if α, μ, σ are all positive and holds the inequality $2\alpha\mu \geq \sigma^2$ then the CIR process 3.1 is well-defined and has a steady marginal distribution.

For our parameter estimates the conditions are hold and our model is well-defined. We will simulate 1000 scenarios with our estimated CIR model 3.1:

$$\begin{aligned} \mathrm{d}r_t &= 0.022(1.232 - r_t)\mathrm{d}t + 0.052\sqrt{r_t}\mathrm{d}W_t \\ r(0) &= 2.143 \end{aligned} \tag{4.1}$$

Figure 4.2 is showing the paths and histogram of our 1000 interest rate simulations. The histogram correspondes to non-central χ^2 distribution.

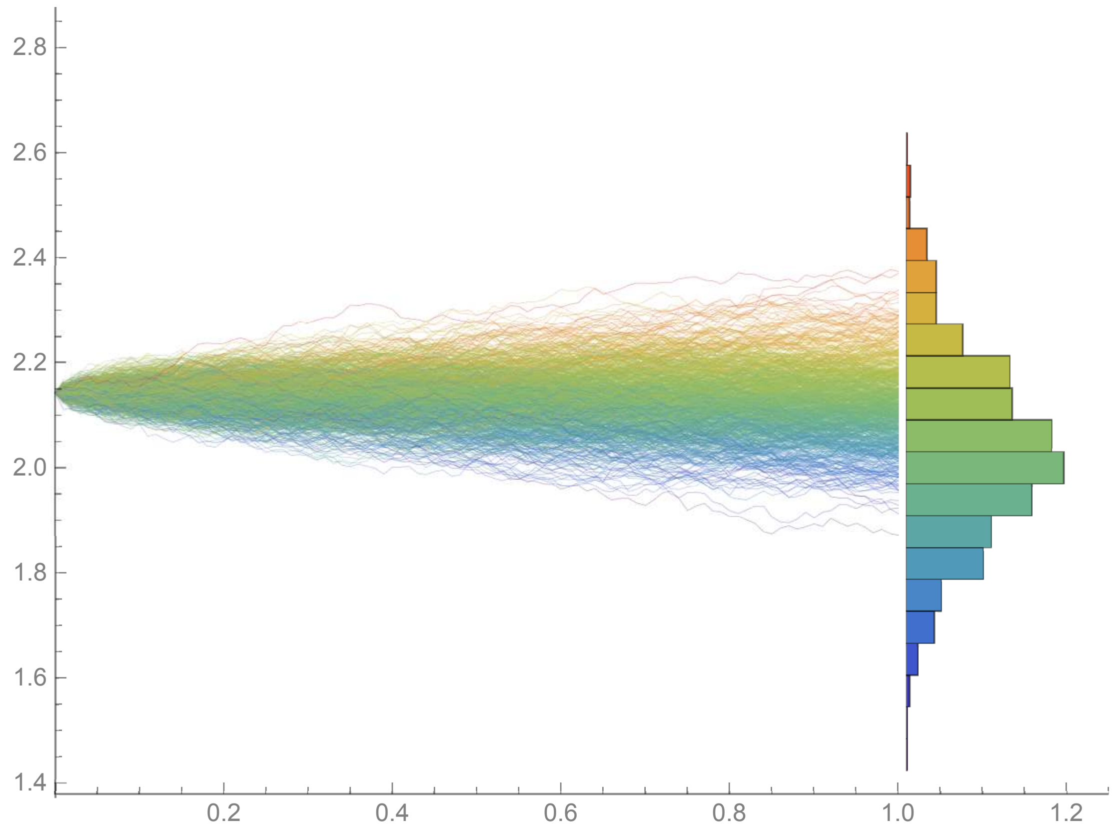

Figure 4.2: Histogram of simulations of interest rates with CIR model 4.1

In our example we will use only 20 simulated economic scenarios. The paths of selected interest rates are shown in the Figure 4.3.

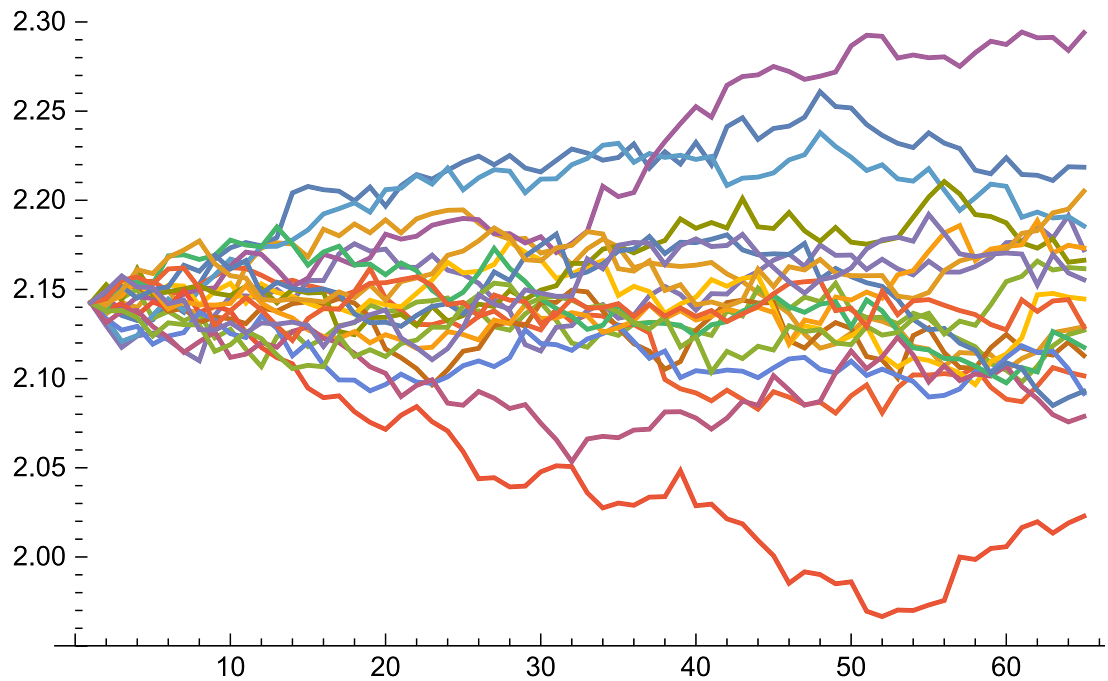

Figure 4.3: Selected interest rates

4.2 Assumptions

We assume that we have only two type of contracts. The first type of contract is with regular (annual) premium payments and the second type are contracts with single premium payment.

For assumptions of mortality we will use the mortality tables from the Czech Statistical Office from year 2015. The mortality data can be found at [14]. We will use the mortality experience coefficients depending on the policy year that is shown in the Table 4.4. The expected mortality in policy year t is $q_x^{exp} = coef_t \cdot q_x$. We assume that the new insurers have lower mortality rates than the individuals who are already insured.

Assumptions of lapses are shown in the Table 4.5. We assume that the probabilities of lapses are higher during the first five policy years.

Further, we will assume that we have some lower (gauranteed) limit of fund value evaluation in time as technical interest rate. We will assume that the technical interest rate is equal to 2,1%. All cash flows are valuated to the date 1.1.2017.

All other assumptions that were used in our example calculation can be found in Table 4.6.

4.3 Results

4.3.1 Analytical function

For better comparison of the results, we will use the datasets with 250, 500 and 1000 model points. Full results of present values of future cash flows can be found in List of Tables in this thesis. Table 4.7, resp 4.8, resp 4.9 shows the result of cash flows for a dataset within the size of 250 MPs, resp. 500 MPs, resp. 1000 MPs.

In this section we will submit the comparison of time needed for calculation by used methods. We can see from the Table 4.1 that analytic function shows better results in each case of our dataset. It's seen that in our example analytic function calculates the present value of cash flow at an average of 10 times faster than the standard policy-by-policy methods. The average difference between the values of present value of cash flows is zero. The insurance companies have much more contracts in their portfolio and with analytic function they can reach the results within a reasonable time. The time difference in calculation for bigger companies can be from hours up to many days according to the size of portfolio.

	Time (in *sec*) Policy-by-policy	Time (in *sec*) Analytic function	Time ratio Pol-by-pol/Analytic	Mean of rel. errors
MP 250	35.771	3.682	9.716	0.000%
MP 500	81.589	5.398	15.116	0.000%
MP 1000	150.198	8.487	17.699	0.000%

Table 4.1: Time comparison of standard cash flow calculation and analytic function

4.3.2 Cluster analysis

The idea of cluster analysis using in acceleration techniques of liability calculation is to speed up the calculation by decreasing the number of model points. This procedure will group the model points into clusers. For each cluster we will have a representer and a scale - count of model points in each cluster. It is possible to choose one of the model points as a group representer or to make a new model point as an average within the group. We will use the second option.

Also we will use three type of datasets (250 MPs, 500 MPs and 1000 MPs). Using the method in Wolfram Mathematica described in 2.3 we will group each dataset into clusters. Mathematica allows to work with points given in an arbitrary number of dimension [5]. The clusters were grouped according to the information of the full model point record, i.e. policy type, sex of policyholder, entry age, inception date, policy period, sum assured, premium and capital value at valuation date. So, for the first dataset of 250 model points we have 161 clusters, for the dataset of 500 model points we have 307 clusters and for 1000 MPs dataset we obtained 423 clusters. Each cluster has the model points with the same policy type (single or regular), the same sex (female or male) and with a minimum difference in other parameter values, i.e. premium, policy period, etc.

We created new model points as representers of the clusters. The representers contain the information of average premium, sum assured, capital value at valuation date, policy period and entry age of policyholder within the cluster, and also the number of policies, which the new model point represents.

We calculated the present value of cash flows using the standard "policy-by-policy" method, but firstly with the full dataset of model points and then with the model points grouped in clusters. The full results of cash flow calculations are summarized in the Tables 4.7, 4.8, 4.9. Table 4.2 contains the time comparison of results using the standard method and the cluster analysis. We can see that for our example the time needed to calculate a present value of future cash flows using the cluster analysis is on average 1.5 times faster than using the standard method. The result difference is up to 1%.

	Time (in *sec*) Policy-by-policy	Time (in *sec*) Cluster Analysis	Time ratio Pol-by-pol/Cluster	Mean of rel. errors
MP 250	35.771	23.634	1.513	0.943%
MP 500	81.589	43.415	1.880	0.404%
MP 1000	150.198	61.934	2.425	0.758%

Table 4.2: Time comparison of standard cash flow calculation for full portfolio and clustered

Conclusion

The purpose of this thesis was to present possible ways about acceleration of valuation of life liabilities. We focused on the UL endowment product. The policyholder will be paid the benefit increased by the value of fund in case of death or maturity.

We introduced the main formulas and principles of calculation of liabilities using the stochastic fair value approach. We also presented the theory for interest rate modelling. We choose a Cox-Ingersoll-Ross (CIR) model and showed the methods for parameter estimation.

We ran all our calculations in Wolfram Mathematica software. We started with the simulation of interest rate scenarios, then we set the assumption for our sample portfolio. In the end we calculated the present value of portfolio cash flows using the standard ”policy-by-policy” method, analytic or ”proxy” function and cluster anlysis that we introduced in this thesis.

We compared the result of our caclulations using three types of portfolio: first with 250 model points, second with 500 model points and finally with 1000 model points. In calculation within this thesis, the analytical function calculates on average about 10 times faster compared to standard method with zero relative errors. Time difference can vary according to the model point samples and selected assumptions. The method described in this thesis can be used with not only strictly mathematical software but also in ordinary available softwares such as MS Excel. The big disadvantage of the method is that initial preparation is highly demanding.

The method of clustering can decrease the number of model points and so decreases the final time of calculation. This method in our example gives on average about 1.5 times faster results than standard ”policy-by-policy” method and the difference in results is up to 1%. The advantage of such a method is that it can be used for every data set. The disadvantage of cluster analysis is that it can produce different cluster results after each usage.

In the case of further analysis and to improve upon these issues, we can use other interest rate models, e.g. Hull-White, Vasicek interest rate models. The results can vary. There is the possibility that a problem could arise regarding calculation with analytic function using more complex benefit payments. For example, if the profit share is paid once a year or if death benefit is paid as a maximum of sum assured and fund value ($\max(SA, CV_t)$).

In this thesis, in cluster analysis implementation, we used the representer of grouped model points as an average within the group. For more development of this work, the usage of weighted average instead of arithmetic average could be attempted.

There are other possibilities to accelerate the calculations that are not discussed in this thesis, for example so-called interpolation approach that uses the grid scenarios. This, however, would be a seperate topic to be researched.

Bibliography

[1] T. Cipra. *Pojistná matematika - teorie a praxe*. Druhé aktualizované vydání. EKOPRESS, 2006.

[2] M. Janeček. *Valuation Techniques of Life Insurance Liabilities*. PhD thesis, Charles university, Faculty of Mathematics and Physics, Prague, 2006.

[3] M. Janeček. Techniques for substantial acceleration of life insurance calculations. 2017. Economic University in Prague.

[4] Marija J. Norušis. *IBM SPSS Statistics 19 Statistical Procedures Companion*.

[5] Wolfram Mathematica. Documentation Center.

[6] P.Tryfos. *Methods for Business Analysis and Forecasting: Text and Cases*. Wiley, 1998.

[7] Wolfram Mathematica. Wolfram MathWorld. `http://mathworld.wolfram.com`.

[8] Steven E. Shreve. *Stochastic Calculus for Finance II*. Springer, 2004.

[9] D. Krycha. Economic scenarios in insurance. Master's thesis, Charles university, Faculty of Mathematics and Physics, Prague, 2012.

[10] T. Cipra. *Finanční Ekonometrie*. Druhé upravené vydání. EKOPRESS, 2013.

[11] K. Kladívko. Maximum Likelihood Estimation of the Cox-Ingersoll-Ross process: the MATLAB Implementation. *Technical Computing Prague*, 2007.

[12] Jr. John C. Cox, Jonathan E. Ingersoll and Stephen A. Ross. A theory of the term structure of interest rates. *Econometrica*, 53:385–407, 1985.

[13] European Central Bank. Statistical Data Warehouse. `https://sdw.ecb.europa.eu/quickview.do?SERIES_KEY=165.YC.B.U2.EUR.4F.G_N_A.SV_C_YM.BETA0`.

[14] Český Statistický Úřad. Úmrtnostní Tabulky. `https://www.czso.cz/csu/czso/umrtnostni_tabulky`.

List of Tables

Policy year			
1	Premium	$+$	$l_0 \cdot P$
	Comm&Exp	$-$	$l_0 \cdot (E_1 + C_1)$
	Sum Assured	$-$	$l_0 q_x \cdot (\mathrm{SA} + \mathrm{CV}_1)$
	Surrender	$-$	$l_0 p_x wthd_1 \cdot \mathrm{CV}_1(1 - fee)$
2	Premium	$+$	$l_0 p_x (1 - wthd_1) \cdot P$
	Comm&Exp	$-$	$l_0 p_x (1 - wthd_1) \cdot (E_2 + C_2)$
	Sum Assured	$-$	$l_0 p_x (1 - wthd_1) q_{x+1} \cdot (\mathrm{SA} + \mathrm{CV}_2)$
	Surrender	$-$	$l_0 p_x (1 - wthd_1)(1 - q_{x+1}) wthd_2 \cdot \mathrm{CV}_2(1 - fee)$
$\vdots$			$\vdots$
$\vdots$			$\vdots$
n	Premium	$+$	$l_0 \prod_{i=0}^{n-2} \Big[p_{x+i}(1 - wthd_{i+1})\Big] \cdot P$
	Comm&Exp	$-$	$l_0 \prod_{i=0}^{n-2} \Big[p_{x+i}(1 - wthd_{i+1})\Big] \cdot (E_n + C_n)$
	Sum Assured	$-$	$l_0 \prod_{i=0}^{n-2} \Big[p_{x+i}(1 - wthd_{i+1})\Big] q_{x+n-1} \cdot (\mathrm{SA} + \mathrm{CV}_n) +$
			$+ l_0 \prod_{i=0}^{n-1} \Big[p_{x+i}(1 - wthd_{i+1})\Big] \cdot (SA + CV_n)$
	Surrender	$-$	$l_0 \prod_{i=0}^{n-2} \Big[p_{x+i}(1 - wthd_{i+1})\Big] (1 - q_{x+n-1}) wthd_n \times$
			$\times \mathrm{CV}_n (1 - fee)$

Table 4.3: Cash flow of Endowment policy during the year

Policy year	Policy type Regular	Single
1	0.30	0.30
2	0.40	0.40
3	0.50	0.50
4	0.60	0.60
≥ 5	0.70	0.70

Table 4.4: Mortality experience

Policy year	Policy type Regular	Single
1	0.20	0.15
2	0.15	0.10
3	0.18	0.13
4	0.15	0.10
5	0.12	0.07
≥ 6	0.08	0.03

Table 4.5: Lapses assumptions

Assumptions	Policy type Regular	Single
α% from SA	3.00%	3.00%
α% from Premium	25.0%	3.00%
β%	0.30%	0.30%
γ%	2.00%	0.00%
Initial Commission % Premium	35.0%	35.0%
Renewal Commission % Premium	4.00%	0.00%
Initial Expenses Fix	2000	0.000
Initial Expenses % Premium	4.00%	1.50%
Renewal Expenses Fix	600.0	0.000
Renewal Expenses % Premium	8.00%	0.50%
Surrender period	2years	2years
Surrender fee	5.00%	5.00%
Inflation rate of fix expenses	2.00%	2.00%

Table 4.6: Other assumptions used in our example

250 model points					
Time of Calculation	(in *sec*)				
Standard method	35.77				
Analytic function	3.68				
Cluster analysis	23.63				
	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Standard method	$-5.4408 \cdot 10^6$	$-5.4459 \cdot 10^6$	$-5.4591 \cdot 10^6$	$-5.4494 \cdot 10^6$	$-5.4570 \cdot 10^6$
Analytic function	$-5.4408 \cdot 10^6$	$-5.4459 \cdot 10^6$	$-5.4591 \cdot 10^6$	$-5.4494 \cdot 10^6$	$-5.4570 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-5.4924 \cdot 10^6$	$-5.4973 \cdot 10^6$	$-5.5103 \cdot 10^6$	$-5.5007 \cdot 10^6$	$-5.5085 \cdot 10^6$
Relative error	0.949%	0.944%	0.939%	0.943%	0.944%
	Scenario 6	Scenario 7	Scenario 8	Scenario 9	Scenario 10
Standard method	$-5.4567 \cdot 10^6$	$-5.4444 \cdot 10^6$	$-5.4509 \cdot 10^6$	$-5.4464 \cdot 10^6$	$-5.4516 \cdot 10^6$
Analytic function	$-5.4567 \cdot 10^6$	$-5.4444 \cdot 10^6$	$-5.4509 \cdot 10^6$	$-5.4464 \cdot 10^6$	$-5.4516 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-5.5080 \cdot 10^6$	$-5.4960 \cdot 10^6$	$-5.5023 \cdot 10^6$	$-5.4979 \cdot 10^6$	$-5.5030 \cdot 10^6$
Relative error	0.941%	0.948%	0,943%	0.945%	0.943%
	Scenario 11	Scenario 12	Scenario 13	Scenario 14	Scenario 15
Standard method	$-5.4667 \cdot 10^6$	$-5.4659 \cdot 10^6$	$-5.4561 \cdot 10^6$	$-5.4666 \cdot 10^6$	$-5.4427 \cdot 10^6$
Analytic function	$-5.4667 \cdot 10^6$	$-5.4659 \cdot 10^6$	$-5.4561 \cdot 10^6$	$-5.4666 \cdot 10^6$	$-5.4427 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-5.5180 \cdot 10^6$	$-5.5174 \cdot 10^6$	$-5.5075 \cdot 10^6$	$-5.5179 \cdot 10^6$	$-5.4941 \cdot 10^6$
Relative error	0.937%	0.942%	0.942%	0.939%	0.944%
	Scenario 16	Scenario 17	Scenario 18	Scenario 19	Scenario 20
Standard method	$-5.4490 \cdot 10^6$	$-5.4466 \cdot 10^6$	$-5.4637 \cdot 10^6$	$-5.4518 \cdot 10^6$	$-5.4567 \cdot 10^6$
Analytic function	$-5.4490 \cdot 10^6$	$-5.4466 \cdot 10^6$	$-5.4637 \cdot 10^6$	$-5.4518 \cdot 10^6$	$-5.4567 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-5.5005 \cdot 10^6$	$-5.4979 \cdot 10^6$	$-5.5152 \cdot 10^6$	$-5.5029 \cdot 10^6$	$-5.5080 \cdot 10^6$
Relative error	0.945%	0.943%	0.942%	0.939%	0.941%

Table 4.7: Comparison of used methods. Present Values of future cash flows and calculation time for dataset with 250 model points and for selected interest scenarios

500 model points	
Time of Calculation	(in *sec*)
Standard method	81.59
Analytic function	5.40
Cluster analysis	43.42

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Standard method	$-7.4645 \cdot 10^6$	$-7.4693 \cdot 10^6$	$-7.4873 \cdot 10^6$	$-7.4738 \cdot 10^6$	$-7.4846 \cdot 10^6$
Analytic function	$-7.4645 \cdot 10^6$	$-7.4693 \cdot 10^6$	$-7.4873 \cdot 10^6$	$-7.4738 \cdot 10^6$	$-7.4846 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-7.4928 \cdot 10^6$	$7.4995 \cdot 10^6$	$-7.5175 \cdot 10^6$	$-7.5039 \cdot 10^6$	$7.5148 \cdot 10^6$
Relative error	0.406%	0.404%	0.403%	0.403%	
	Scenario 6	Scenario 7	Scenario 8	Scenario 9	Scenario 10
Standard method	$-7.4839 \cdot 10^6$	$-7.4674 \cdot 10^6$	$-7.4760 \cdot 10^6$	$-7.4700 \cdot 10^6$	$-7.4769 \cdot 10^6$
Analytic function	$-7.4839 \cdot 10^6$	$-7.4674 \cdot 10^6$	$-7.4760 \cdot 10^6$	$-7.4700 \cdot 10^6$	$-7.4769 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-7.5140 \cdot 10^6$	$-7.4977 \cdot 10^6$	$-7.5062 \cdot 10^6$	$-7.5002 \cdot 10^6$	$-7.5071 \cdot 10^6$
Relative error	0.402%	0.406%	0.405%	0.405%	0.404%
	Scenario 11	Scenario 12	Scenario 13	Scenario 14	Scenario 15
Standard method	$-7.4974 \cdot 10^6$	$-7.4966 \cdot 10^6$	$-7.4830 \cdot 10^6$	$-7.4975 \cdot 10^6$	$-7.4646 \cdot 10^6$
Analytic function	$-7.4974 \cdot 10^6$	$-7.4966 \cdot 10^6$	$-7.4830 \cdot 10^6$	$-7.4975 \cdot 10^6$	$-7.4646 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-7.5275 \cdot 10^6$	$-7.5270 \cdot 10^6$	$-7.5132 \cdot 10^6$	$-7.5276 \cdot 10^6$	$-7.4947 \cdot 10^6$
Relative error	0.401%	0.405%	0.404%	0.402%	0.402%
	Scenario 16	Scenario 17	Scenario 18	Scenario 19	Scenario 20
Standard method	$-7.4733 \cdot 10^6$	$-7.4699 \cdot 10^6$	$-7.4936 \cdot 10^6$	$-7.4771 \cdot 10^6$	$-7.4839 \cdot 10^6$
Analytic function	$-7.4733 \cdot 10^6$	$-7.4699 \cdot 10^6$	$-7.4936 \cdot 10^6$	$-7.4771 \cdot 10^6$	$-7.4839 \cdot 10^6$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-7.5035 \cdot 10^6$	$-7.5001 \cdot 10^6$	$-7.5240 \cdot 10^6$	$-7.5071 \cdot 10^6$	$-7.5140 \cdot 10^6$
Relative error	0.404%	0.405%	0.405%	0.402%	0.402%

Table 4.8: Comparison of used methods. Present Values of future cash flows and calculation time for dataset with 500 model points and for selected interest scenarios

1000 model points	
Time of Calculation	(in *sec*)
Standard method	150.20
Analytic function	8.49
Cluster analysis	61.93

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Standard method	$-1.1878 \cdot 10^7$	$-1.1887 \cdot 10^7$	$-1.1913 \cdot 10^7$	$-1.1893 \cdot 10^7$	$-1.1910 \cdot 10^7$
Analytic function	$-1.1878 \cdot 10^7$	$-1.1887 \cdot 10^7$	$-1.1913 \cdot 10^7$	$-1.1893 \cdot 10^7$	$-1.1910 \cdot 10^7$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-1.1968 \cdot 10^7$	$-1.1978 \cdot 10^7$	$-1.2003 \cdot 10^7$	$-1.1983 \cdot 10^7$	$1.2001 \cdot 10^7$
Relative error	0.763%	0.760%	0.756%	0.758%	0.759%
	Scenario 6	Scenario 7	Scenario 8	Scenario 9	Scenario 10
Standard method	$-1.19075 \cdot 10^7$	$-1.1885 \cdot 10^7$	$-1.1896 \cdot 10^7$	$-1.1888 \cdot 10^7$	$-1.1897 \cdot 10^7$
Analytic function	$-1.19075 \cdot 10^7$	$-1.1885 \cdot 10^7$	$-1.1896 \cdot 10^7$	$-1.1888 \cdot 10^7$	$-1.1897 \cdot 10^7$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-1.1998 \cdot 10^7$	$-1.1976 \cdot 10^7$	$-1.1986 \cdot 10^7$	$-1.1979 \cdot 10^7$	$-1.1988 \cdot 10^7$
Relative error	0.757%	0.764%	0.759%	0.761%	0.759%
	Scenario 11	Scenario 12	Scenario 13	Scenario 14	Scenario 15
Standard method	$-1.1927 \cdot 10^7$	$-1.1926 \cdot 10^7$	$-1.1906 \cdot 10^7$	$-1.1928 \cdot 10^7$	$-1.1880 \cdot 10^7$
Analytic function	$-1.1927 \cdot 10^7$	$-1.1926 \cdot 10^7$	$-1.1906 \cdot 10^7$	$-1.1928 \cdot 10^7$	$-1.1880 \cdot 10^7$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-1.2016 \cdot 10^7$	$-1.2016 \cdot 10^7$	$-1.1996 \cdot 10^7$	$-1.2017 \cdot 10^7$	$-1.1970 \cdot 10^7$
Relative error	0.752%	0.757%	0.757%	0.754%	0.760%
	Scenario 16	Scenario 17	Scenario 18	Scenario 19	Scenario 20
Standard method	$-1.1892 \cdot 10^7$	$-1.1887 \cdot 10^7$	$-1.1922 \cdot 10^7$	$-1.1898 \cdot 10^7$	$-1.1907 \cdot 10^7$
Analytic function	$-1.1892 \cdot 10^7$	$-1.1887 \cdot 10^7$	$-1.1922 \cdot 10^7$	$-1.1898 \cdot 10^7$	$-1.1907 \cdot 10^7$
Relative error	0.00%	0.00%	0.00%	0.00%	0.00%
Cluster analysis	$-1.1982 \cdot 10^7$	$-1.1977 \cdot 10^7$	$-1.2012 \cdot 10^7$	$-1.1987 \cdot 10^7$	$-1.1997 \cdot 10^7$
Relative error	0.760%	0.760%	0.757%	0.758%	0.757%

Table 4.9: Comparison of used methods. Present Values of future cash flows and calculation time for dataset with 500 model points and for selected interest scenarios

List of Abbreviations

x	age of insured person
n	policy period in years
t	policy year, $0 \leq t \leq n$
q_x	probability that a person who is alive at the age of x will die before the age $x + 1$
p_x	probability that a person who is alive at the age of x will be alive at the age of $x + 1$
${}_tp_x$	probability that a person who is alive at the age of x will be alive at the age of $x + t$
ℓ_t	expected number of policies in-force at the end of the year t
d_x	expected number of deaths during the year t
w_t	expected number of of lapses during the year t
$wthd_t$	probability of lapse at the policy year t
P_t	premium income assume to be happen at the beginning of the year t
C_t	commissions paid to agents assumed to be paid at the beginnning of the year t
E_t	expenses outflow assumed to be paid at the beginning of the policy year t
SA	sum assured paid in case of death or maturity in the policy year t
CV_t	fund (capital) value at the end of the year t
RP_t	risk part of the premium at the end of year t
SP_t	saving part of the premium at the end of year t
rfr_t	risk free rate related to the year t
i_t	interest rate related to the year t
TIR	technical interest rate
fee	surrender charge applied when the surrender is paid assumed to be at the end of the year t
CF_t	cash flow at the end of the policy year t
FV_t	Fair Value of liabilities at the end of the policy year t
MP	model point

Attachments

Source code in Mathematica

1. Standard method of cash-flow calculation (Policy-by-policy)

```
In[1]:= PVofCF[typ_, pocdat_, vstvek_, pohl_, n_, valDate_,
        scen_, PC_, Prem_, f0_, Count_] :=
        Module[{polYear, age, polYearVect, periodnew,
                MortExpVect, q, qexp, LapseVect, invIncome,
                discfact, riskpr, AlphaVect, BetaVect,
                GammaVect, konst, distrInvIncome, rate,
                SurrPaid, CommVector, ExpensesVector,
                inforceBoY, inforceEoY, numDeath, numLapses,
                numLapsesPom, numMatur, FondBoYReal,
                PremiumReal, AlphaReal, BetaReal,
                GammaReal, PCReal, riskprReal,
                distrInvIncomeReal, FondEoYReal, SurrPaidReal,
                DeathPaidReal, MaturPaidReal, CommReal,
                ExpensesReal, CFvector, CFdisc, fundBoY,
                fundEoY, discrate, PremPom, PremVector},
        polYear = QuantityMagnitude[DateDifference[pocdat,
                valDate, "Year"]] + 1;
        age = vstvek + QuantityMagnitude[DateDifference[pocdat,
                valDate, "Year"]];
        polYearVect = Table[polYear + i, {i, 0, n - polYear}];
        periodnew = n - polYear + 1;
        MortExpVect = ConstantArray[{}, periodnew];
        For[i = 1, i <= periodnew, i++,
                If[polYearVect[[i]] < Length[mortexp[[typ]]],
                MortExpVect[[i]] =
                        mortexp[[typ,polYearVect[[i]]]],
                MortExpVect[[i]] = Last[mortexp[[typ]]]]];
        If[pohl == 0, q = Table[qmale[[i]], {i, age + 1,
                        age + periodnew}],
                q = Table[qfemale[[i]], {i, age + 1,
                        age + periodnew}]];
        qexp = q*MortExpVect;
        LapseVect = ConstantArray[{}, periodnew];
        For[i = 1, i <= periodnew, i++,
                If[polYearVect[[i]] < Length[lapsevect[[typ]]],
                LapseVect[[i]] =
                        lapsevect[[typ,polYearVect[[i]]]],
                LapseVect[[i]] = Last[lapsevect[[typ]]]]];
        invIncome = Scenarios[[scen]];
        discrate = DiscountScenarios[[scen]];
        discfact = Table[Product[1/(1 + discrate[[i]]),
                {i, 1, k}], {k, 1, periodnew}];
```

```
PremPom = ConstantArray[{}, periodnew];
For[i = 1, i <= periodnew, i++,
        If[typ == 2,
                If[polYearVect[[i]] == 1,
                        PremPom[[i]] = 1,
                                PremPom[[i]] = 0],
                        PremPom[[i]] = 1]];
PremVector = Prem*PremPom;
riskpr = Table[(PC*q[[i]])/(1 + TIR[[typ]]),
        {i, 1, Length[q]}];
AlphaVect = ConstantArray[{}, periodnew];
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] == 1,
                AlphaVect[[i]] = AlphaP[[typ]]*Prem +
                AlphaSA[[typ]]*PC, AlphaVect[[i]] = 0]];
BetaVect = ConstantArray[Beta[[typ]]*PC, periodnew];
GammaVect = ConstantArray[Gamma[[typ]]*Prem, periodnew];
konst = PremVector - AlphaVect - BetaVect - GammaVect -
                riskpr;
rate = Map[Max[TIR[[typ]], #] &, invIncome -
                InvestmentMargin];
fundBoY = Join[{f0}, ConstantArray[{}, periodnew - 1]];
For[i = 1, i <= periodnew - 1, i++,
        fundBoY[[i + 1]] = (fundBoY[[i]] + konst[[i]])*
                        (rate[[i]] + 1)];
fundEoY = Join[Rest[fundBoY], {(fundBoY[[periodnew]] +
        konst[[periodnew]])*(rate[[periodnew]] + 1)}];
distrInvIncome = (fundBoY + konst)*Table[rate[[i]],
                {i, 1, periodnew}];
SurrPaid = (1 - SurrFee[[typ]])*fundEoY;
CommVector = ConstantArray[{}, periodnew];
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] == 1,
        CommVector[[i]] = InitCommP[[typ]]*Prem +
                        InitCommSA[[typ]]*PC,
        CommVector[[i]] = RenCommP[[typ]]*Prem +
                        RenCommSA[[typ]]*PC]];
ExpensesVector = ConstantArray[{}, periodnew];
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] == 1,
        ExpensesVector[[i]] = InitExpFix[[typ]] +
                InitExpP[[typ]]*Prem +
                RenExpFix[[typ]]*
                (1 + InflRateFixExp)^(i - 1) +
                RenExpP[[typ]]*Prem,
        ExpensesVector[[i]] = RenExpFix[[typ]]*
                (1 + InflRateFixExp)^(i - 1) +
                RenExpP[[typ]]*Prem]];
```

```
inforceBoY = Join[{Count}, ConstantArray[{},
                periodnew - 1]];
For[i = 1, i <= periodnew - 1, i++,
        inforceBoY[[i + 1]] = inforceBoY[[i]]*
                (1 - qexp[[i]])*
                (1 - If[polYearVect[[i]] > 2,
                        LapseVect[[i]], 0])];
numDeath = inforceBoY*qexp;
numLapsesPom = (inforceBoY - numDeath)*LapseVect;
numLapses = numLapsesPom;
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] > 2, numLapses[[i]] =
                numLapsesPom[[i]], numLapses[[i]] = 0]];
numMatur = ConstantArray[0, periodnew];
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] == n,
                numMatur[[i]] = inforceBoY[[i]] -
                        numDeath[[i]] - numLapses[[i]],
                        numMatur[[i]] = 0]];
inforceEoY = inforceBoY - numDeath -
                numLapses - numMatur;
FondBoYReal = inforceBoY*fundBoY;
PremiumReal = PremVector*inforceBoY;
AlphaReal = AlphaVect*inforceBoY;
BetaReal = BetaVect*inforceBoY;
GammaReal = GammaVect*inforceBoY;
PCReal = PC*inforceBoY;
riskprReal = riskpr*inforceBoY;
distrInvIncomeReal = distrInvIncome*inforceBoY;
FondEoYReal = fundEoY*inforceEoY;
SurrPaidReal = SurrPaid*numLapses;
DeathPaidReal = (fundEoY + PC)*numDeath;
MaturPaidReal = (fundEoY + PC)*numMatur;
CommReal = CommVector*inforceBoY;
ExpensesReal = ExpensesVector*inforceBoY;
CFvector = PremiumReal - SurrPaidReal - DeathPaidReal -
        MaturPaidReal - CommReal - ExpensesReal;
CFdisc = CFvector*discfact;
Total[CFdisc]]
```

```
In[2]:= cftable = ConstantArray[0, {Length[PolicyType],
                Length[Scenarios]}]
```

```
In[3]:= cfallscenallMP = ConstantArray[0, Length[Scenarios]]
```

```
In[4]:= For[m = 1, m <= Length[PolicyType], m++,
        For[s = 1, s <= Length[Scenarios], s++,
        cftable[[m,s]] = PVofCF[PolicyType[[m]],
        IncDates[[m]], EntryAge[[m]], Sex[[m]],
```

```
            PolicyPeriod[[m]], ValuationDate, s,
            SumAssured[[m]], Premiums[[m]],
            CVatValDate[[m]], CountPol[[m]]]]]
```

```
In[5]:= For[w = 1, w <= Length[Scenarios], w++,
            cfallscenal1MP[[w]] =
                    Total[Transpose[cftable][[w]]]]
```

2. Analytic function for cash calculation

```
In[6]:= fcepro1MP[typ_, pocdat_, vstvek_, pohl_, n_, valDate_,
    PC_, Prem_, f0_, Count_] :=
    Module[{polYear, age, polYearVect, periodnew,
    MortExpVect, q, qexp, LapseVect, PremPom, PremVector,
    riskpr, AlphaVect, BetaVect, GammaVect, inforceBoY,
    numDeath, numLapsesPom, numLapses, numMatur, inforceEoY,
    CommVector, ExpensesVector, PremiumReal, CommReal,
    ExpensesReal, NetPremium, PayoutSA, Decrems, PayoutCV,
    SavPrem, PayoutSP, NetPremEq, PayoutSAEq, PayoutCVEq,
    PayoutSPEq},
    polYear = QuantityMagnitude[DateDifference[pocdat,
                valDate, "Year"]] + 1;
    age = vstvek + QuantityMagnitude[DateDifference[pocdat,
                valDate, "Year"]];
    polYearVect = Table[polYear + i, {i, 0, n - polYear}];
    periodnew = n - polYear + 1;
    MortExpVect = ConstantArray[{}, periodnew];
    For[i = 1, i <= periodnew, i++,
            If[polYearVect[[i]] < Length[mortexp[[typ]]],
            MortExpVect[[i]] =
                            mortexp[[typ,polYearVect[[i]]]],
            MortExpVect[[i]] = Last[mortexp[[typ]]]]];
    If[pohl == 0, q = Table[qmale[[i]],
            {i, age + 1, age + periodnew}],
                q = Table[qfemale[[i]],
                {i, age + 1, age + periodnew}]];
    qexp = q*MortExpVect;
    LapseVect = ConstantArray[{}, periodnew];
    For[i = 1, i <= periodnew, i++,
            If[polYearVect[[i]] < Length[lapsevect[[typ]]],
            LapseVect[[i]] =
                    lapsevect[[typ,polYearVect[[i]]]],
            LapseVect[[i]] = Last[lapsevect[[typ]]]]];
    PremPom = ConstantArray[{}, periodnew];
    For[i = 1, i <= periodnew, i++,
            If[typ == 2, If[polYearVect[[i]] == 1,
            PremPom[[i]] = 1, PremPom[[i]] = 0],
            PremPom[[i]] = 1]];
    PremVector = Prem*PremPom;
```

```
riskpr = Table[(PC*q[[i]])/(1 + TIR[[typ]]),
               {i, 1, Length[q]}];
AlphaVect = ConstantArray[{}, periodnew];
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] == 1,
        AlphaVect[[i]] = AlphaP[[typ]]*Prem +
                AlphaSA[[typ]]*PC, AlphaVect[[i]] = 0]];
BetaVect = ConstantArray[Beta[[typ]]*PC, periodnew];
GammaVect = ConstantArray[Gamma[[typ]]*Prem, periodnew];
inforceBoY = Join[{Count}, ConstantArray[{},
                        periodnew - 1]];
For[i = 1, i <= periodnew - 1, i++,
        inforceBoY[[i + 1]] = inforceBoY[[i]]*
        (1 - qexp[[i]])*(1 - If[polYearVect[[i]] > 2,
                        LapseVect[[i]], 0])];
numDeath = inforceBoY*qexp;
numLapsesPom = (inforceBoY - numDeath)*LapseVect;
numLapses = numLapsesPom;
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] > 2,
        numLapses[[i]] = numLapsesPom[[i]],
        numLapses[[i]] = 0]];
numMatur = ConstantArray[0, periodnew];
For[i = 1, i <= periodnew, i++,
                If[polYearVect[[i]] == n,
                numMatur[[i]] = inforceBoY[[i]] -
                                numDeath[[i]] -
                                numLapses[[i]],
                                numMatur[[i]] = 0]];
inforceEoY = inforceBoY - numDeath -
                        numLapses - numMatur;
CommVector = ConstantArray[{}, periodnew];
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] == 1,
        CommVector[[i]] = InitCommP[[typ]]*Prem +
                InitCommSA[[typ]]*PC,
        CommVector[[i]] = RenCommP[[typ]]*Prem +
                RenCommSA[[typ]]*PC]];
ExpensesVector = ConstantArray[{}, periodnew];
For[i = 1, i <= periodnew, i++,
        If[polYearVect[[i]] == 1,
        ExpensesVector[[i]] = InitExpFix[[typ]] +
        InitExpP[[typ]]*Prem + RenExpFix[[typ]]*
        (1 + InflRateFixExp)^(i - 1) +
        RenExpP[[typ]]*Prem,
        ExpensesVector[[i]] = RenExpFix[[typ]]*
        (1 + InflRateFixExp)^(i - 1) +
        RenExpP[[typ]]*Prem]];
```

```
PremiumReal = PremVector*inforceBoY;
CommReal = CommVector*inforceBoY;
ExpensesReal = ExpensesVector*inforceBoY;
NetPremium = PremiumReal - CommReal - ExpensesReal;
NetPremEq = PadRight[NetPremium, 65];
PayoutSA = (numDeath + numMatur)*PC;
PayoutSAEq = PadRight[PayoutSA, 65];
Decrems = numDeath + numMatur +
                numLapses*(1 - SurrFee[[typ]]);
PayoutCV = f0*Decrems;
PayoutCVEq = PadRight[PayoutCV, 65];
SavPrem = PremVector - AlphaVect - BetaVect -
                GammaVect - riskpr;
PayoutSP = LowerTriangularize[Table[Decrems[[i]]*
                SavPrem[[j]], {i, 1, Length[Decrems]},
                {j, 1, Length[Decrems]}]];
PayoutSPEq = PadRight[PayoutSP, {65, 65}];
{NetPremEq, PayoutSAEq, PayoutCVEq, PayoutSPEq}]
```

```
In[7]:= fceScen[scen_, cc_, dd_] :=
Module[{invIncome, incomerate, discrate, discfact,
Tdiscfact, itriangle, PayoutSPint, sumPayoutSPint},
invIncome = Scenarios[[scen]];
incomerate = 1 + Map[Max[TIR[[1]], #] &,
                invIncome - InvestmentMargin];
discrate = DiscountScenarios[[scen]];
discfact = Table[Product[1/(1 + discrate[[i]]),
                {i, 1, k}], {k, 1, Length[discrate]}];
Tdiscfact = Table[Product[incomerate[[i]], {i, 1, j}],
                {j, 1, Length[discrate]}]*cc;
itriangle =
LowerTriangularize[Table[Product[incomerate[[i]],
        {i, l, j}], {j, 1, Length[discrate]},
        {l, 1, Length[discrate]}]];
PayoutSPint = itriangle*dd;
sumPayoutSPint = ConstantArray[0, Length[discrate]];
For[w = 1, w <= Length[discrate], w++,
        sumPayoutSPint[[w]] = Total[PayoutSPint[[w]]]];
{Tdiscfact, sumPayoutSPint, discfact}]
```

```
In[8]:= VektCF := Module[{AOld, BOld, COld, DOld, ANew, APol,
BNew, BPol, CNew, CPol, DNew, DPol, allscen, C2, D2,
discF},
AOld = ConstantArray[0, 65];
BOld = ConstantArray[0, 65];
COld = ConstantArray[0, 65];
DOld = ConstantArray[0, 65];
For[u = 1, u <= Length[PolicyType], u++,
{APol, BPol, CPol, DPol} = fcepro1MP[PolicyType[[u]],
```

```
IncDates[[u]], EntryAge[[u]], Sex[[u]],
PolicyPeriod[[u]], ValuationDate,
SumAssured[[u]], Premiums[[u]],
CVatValDate[[u]], CountPol[[u]]];
ANew = APol + AOld; AOld = ANew;
BNew = BPol + BOld; BOld = BNew;
CNew = CPol + COld; COld = CNew;
DNew = DPol + DOld; DOld = DNew];
allscen = ConstantArray[0, Length[Scenarios]];
For[s = 1, s <= Length[Scenarios], s++,
{C2, D2, discF} = fceScen[s, CNew, DNew];
allscen[[s]] =
        Total[(ANew - BNew - C2 - D2)*discF]];
allscen]
```