Universität Ulm
Fakultät für Mathematik und
Wirtschaftswissenschaften

The Impact of Stochastic Volatility on Pricing, Hedging and Hedge Efficiency of Guaranteed Minimum Withdrawal Benefits for Life Contracts

Diplomarbeit

in Wirtschaftsmathematik

vorgelegt von
Frederik Ruez
September 2009

Gutachter

Prof. Dr. Hans-Joachim Zwiesler
Prof. Dr. Rüdiger Kiesel

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Motivation of the thesis

Variable Annuities are fund-linked annuities where typically the policyholder pays a single premium that is then – after deduction of acquisition fees – invested in one or several mutual funds. Usually, the policyholder may choose from a variety of different mutual funds, which in most cases consist of positions in bonds and equities.
Such products were introduced in the 1970s in the United States. Two decades later, in the 1990s, insurers started to offer certain guarantee riders on top of the basic structure of the Variable Annuity policies, including so-called guaranteed minimum death benefits (GMDB), as well as guaranteed minimum living benefits, which can be classified into three main subcategories: guaranteed minimum accumulation benefits (GMAB), guaranteed minimum income benefits (GMIB) and guaranteed minimum withdrawal benefits (GMWB). The GMAB type of guarantee provides the policyholder with some guaranteed value at the maturity of the contract, while the GMIB type provides a guaranteed annuity benefit, starting after a certain deferment period. However, the currently most popular type of guaranteed minimum living benefits is the third one, the GMWB rider.
Under certain conditions, the policyholder may withdraw money from their account, even if the value of the account has dropped to zero since policy inception. Such withdrawals are guaranteed as long as both, the amount that is withdrawn within each policy year and the total amount that has been withdrawn over the term of the policy stay within certain limits.
Recently, insurers started to include additional features in GMWB type of products. The most prominent is called "GMWB for Life" (also known as guaranteed living

withdrawal benefits, GLWB): guaranteed lifelong withdrawals. Within this guarantee type, the total amount of withdrawals is unlimited. However, the annual amount that may be withdrawn by the policyholder while the insured is still alive must not exceed some maximum value, or otherwise the guarantee will be affected.
The withdrawals made by the policyholder are deducted from their account value – as long as it has not been depleted. Afterwards, the insurer has to compensate for the guaranteed withdrawals for the rest of the insured's life.
In contrast to a conventional annuity, in which the single premium paid by the policyholder is annuitized, the fund assets of the contract remain accessible to the policyholder within a "GMWB-for-Life" or "GLWB" type of policy. The policyholder may access the remaining fund assets at any time by (partially) surrendering the contract.
In case of death of the insured, any remaining fund value is paid out to the insured's beneficiaries.

In return for this guarantee, the insurer receives guarantee fees that are usually deducted as a fixed annual percentage from the policyholder's fund assets (but of course only as long as there are any assets left).

Therefore, from an insurer's point of view, these products contain a combination of several risks from policyholder behavior, financial markets, and longevity that makes these kind of guarantees difficult to hedge.

Owing to the significant financial risk that is inherent within the insurance contracts sold, in general risk management strategies such as dynamic hedging are applied. However, during the recent financial crisis, insurers have suffered from inefficient hedging strategies within their books[1].

Among other effects, the financial crises led to a significant increase in actual and implied equity volatility, and thus to a tremendous increase in the value of most standard and non-standard equity-linked options, including the value of the sold and offered Variable Annuity guarantee riders. In particular for insurers with no or no sufficient hedging concept against the risk of changing volatilities (in particular increasing volatility levels), the hedge portfolio did not compensate for the increase in the option's value, leading to a loss for existing business and less attractive conditions for new contracts, i.e. the same guarantees come at higher guarantee fees.

[1]Cf. e.g. different articles and papers in "Life and Pensions": "A challenging environment" (June 2008), "Variable Annuities - Flawed product design costs Old Mutual 150m" (September 2008), "Variable annuities - Milliman denies culpability for clients' hedging losses" (October 2008), "Variable Annuities - Axa injects $3bn into US arm" (January 2009).

There already exists some literature on the pricing of different guaranteed minimum benefits and in particular on the pricing of GMWB rider options: Valuation methods have been proposed by e.g., Milevsky and Posner (2001, [26]) for the GMDB option, Milevsky and Salisbury (2006, [27]) for the GMWB option, Bacinello et al. (2009, [3]) for life insurance contracts with surrender guarantees, and Holz et al. (2007, [17]) for GMWB for Life riders.
Bauer et al. (2008, [2]) introduced a general model framework that allows for the simultaneous and consistent pricing and analysis of various Variable Annuity guarantees. We also refer to their paper for a comprehensive analysis of non-pricing related literature on Variable Annuities.

To our knowledge, however, there is only little literature on the efficiency of different strategies for hedging against the market risk inherent in Variable Annuity guarantees. Coleman et al. (2005, [7], and 2007, [8]) provide such analyses for death benefit guarantees under different hedging and data-generating models. Again to our knowledge, the performance of different hedging strategies for GLWB contracts with varied product designs has not been analyzed under stochastic equity volatility yet, nor have the accompanying inherent model risks been evaluated.

The aim of this thesis is to fill this gap.

1.2 Outline of the thesis

The remainder of the thesis is organized as follows.

First, we give a high-level description of the Guaranteed Living Withdrawal Benefits (GLWB) rider options and explain their general functionality in chapter 2, where we also present the product designs of the GLWB rider options that will be analyzed in the numerical section of the thesis. The considered product designs differ in the ratchet (or step-up) and bonus feature that is implemented in the respective design. We also describe the model framework for insurance liabilities that is used for our analyses and which is akin to the one presented by Bauer et al. (2008, [2]).
In the last part of chapter 2, we describe the general pricing framework that is used for the evaluation of the GLWB rider option, which is necessary in order to find the "fair" guaranteed withdrawal rate. We also present the assumptions that we used regarding the policyholder behavior and the mortality of the insured.

In chapter 3, we present the models of the financial market that we will use for different purposes within our analyses. We describe in detail both equity models that we

use for comparison, the well-known Black-Scholes-Merton model (with deterministic equity volatility) as reference model and the Heston model as a model that allows for stochastic equity volatility. We show how for both models an equivalent martingale measure can be derived, which is necessary for the evaluation of the GLWB options within the general pricing framework of chapter 2.
In the second part of chapter 3 we present and explain the numerical methods that we use later for the analyses within the numerical part of this thesis, including the numerical valuation of European standard options via Fourier inversion techniques under the Heston model.
Further, we introduce the notion of *Black-Scholes-Merton Implied Volatility* and analyze the implied volatility surfaces generated by the Heston model for different parameter sets.

The hedging strategies whose hedge efficiency we will analyze later in the numerical part of the thesis are presented in chapter 4, where we introduce different types of dynamic hedging strategies that may be applied by the insurance company in order to reduce the risk that originates from selling GLWB rider options. The strategies differ mainly in the model used for hedging and in the hedge instruments used within the strategies. Finally, for each hedging strategy, we give a clear summary of the used hedge ratios in table form.

Chapter 5 deals with the design, architecture and implementation of the software solution that we used to conduct the analyses of the numerical part of this thesis. We also give details on how we use Microsoft Excel as an user interface to our software solution and we list and discuss the open-source libraries that we made use of. We conclude the chapter with a detailed usage example of the final software solution.

In the first chapter of the numerical part of the thesis, chapter 6, we present the first set of analyses regarding the pricing of the GLWB rider options, i.e. the determination of the "fair" guaranteed withdrawal rate under different product designs and under different assumptions regarding the financial market and policyholder behavior.
We also analyze the characteristics of the considered product designs that were introduced in chapter 2. To this end, we analyze the risk-return profiles of the considered contracts from the viewpoint of the policyholder, and analyze the respective distribution of the GLWB guarantee's "trigger time", i.e. the specific point in time, when, for the first time, guarantee payments from the insurer to the policyholder are due. Furthermore, we present our results regarding the development of the sensitivity of a pool of identical policies with respect to changes in the underlying fund's price (the

so-called “Delta”) over time and for each product design.

In the second chapter of the numerical part of this thesis, chapter 7, we analyze the performance and behavior of the different hedging strategies that were presented in chapter 4 under different stochastic scenarios. The results include analyses of the distribution of the hedge portfolio’s value, the insurer’s cumulative profit/loss and certain risk measures hereof. Additionally, we quantify the model risk that is inherent in the use of the considered hedging strategies. To this end, we use different models of the financial market for calculation of the hedge positions and for data generation within the simulation. A particular focus lies on examining the effects if the model used for hedging differs from the data-generating model.

We conclude with a summary of the presented work in the final chapter 8, where we also give an outlook of possible future research work.

Chapter 2

Liability Framework

In Bauer et al. (2008), a general framework for modeling and a valuation of variable annuity contracts was introduced. Within this framework, any contract with one or several living benefit guarantees and/or a guaranteed minimum death benefit can be represented. In their numerical analysis however, only contracts with a rather short finite time horizon were considered. Within the same model framework, Holz et al. (2008) describe how GMWB for Life products can be included in this model. In what follows, we introduce this model framework focusing on the peculiarities of the contracts considered within our numerical analyses. We refer to Bauer et al. (2008) as well as Holz et al. (2008) for the explanation of other living benefit guarantees and more details on the model.

2.1 High-level description of the considered insurance contracts

Variable Annuities are fund-linked products. At inception of the contract, the policyholder pays a single premium P, which is then – after deduction of acquisition fees – invested in one or several mutual funds. We call the value of the insured's individual portfolio the *account value* and denote it by AV_t. During the runtime of the contract, all running fees are taken from the account value by cancellation of fund units. Furthermore, the policyholder has the possibility to surrender the contract, which is the same as withdrawing the whole account value, or, of course, to withdraw just a portion of the account value. Products with a GMWB ("Guaranteed Minimum Withdrawal Benefits") option give the policyholder the possibility of guaranteed withdrawals during the lifespan of the contract, i.e. the policyholder may withdraw a guaranteed amount at prespecified points in time, even if the value of the portfolio has dropped to zero in the meantime. The initially guaranteed withdrawal amount is usually (or may be expressed as) a certain percentage x_{WL} of the single premium P.

For our analyses, we focus on the case where such withdrawals are guaranteed lifelong ("GMWB for life" or "Guaranteed Lifetime Withdrawal Benefits", GLWB), which means that, if the account value of the policy drops to zero while the insured is still alive, the insured can still continue to withdraw the guaranteed amount until death. The insurer charges a fee for this guarantee which is usually a prespecified annual percentage of the account value. In case the insured dies before the account value was depleted and/or the contract was surrendered, the remaining account value is paid to the beneficiary as death benefit.

Depending on the type of the GLWB option, the amount guaranteed for withdrawal may increase during the policy lifespan if the fund's assets perform well, allowing the policyholder to withdraw a higher amount than the initially guaranteed amount. This increase may either be permanent (withdrawal "step-up" or "ratchet") or be effective just for the single withdrawal ("surplus distribution" or "performance bonus"). In our numerical analyses in sections 6 and 7, we have a closer look on four different product designs that can be observed in the market:

- **No Ratchet:** The first and simplest alternative is one where no ratchets or surplus exists at all. In this case, the guaranteed withdrawal amount is constant and does not depend on market movements.
- **Lookback Ratchet:** The second alternative is a ratchet mechanism where a withdrawal benefit base at outset is given by the single premium paid. During the contract term, on each policy calculation date the withdrawal benefit base is increased to the account value if the account value exceeds the previous withdrawal benefit base. The guaranteed withdrawal is increased accordingly to x_{WL} multiplied by the new withdrawal benefit base. This effectively means that the fund performance needs to compensate for policy charges and annual withdrawals in order to increase the guaranteed withdrawals. With this product design, increases in the guaranteed withdrawal amount are permanent, i.e. over time, the guaranteed withdrawal amount may only increase, never decrease.

- **Remaining WBB Ratchet:** With the third ratchet mechanism, the withdrawal benefit base at outset is also given by the single premium paid. The withdrawal benefit base is however reduced by every guaranteed withdrawal. On each policy calculation date where the current account value exceeds the current withdrawal benefit base, the withdrawal benefit base is increased to the account value. The guaranteed annual withdrawal is increased by x_{WL} multiplied by the difference between the account value and the previous withdrawal benefit base. This effectively means that the fund performance needs to compensate for policy charges only but not for withdrawals in order to increase guaranteed withdrawals. This ratchet mechanism is therefore c.p. somewhat "richer" than the Lookback Ratchet. As with the Lookback Ratchet design, increases in the guaranteed amount are permanent.

- **Performance Bonus:** For this alternative the withdrawal benefit base is defined similarly as in the Remaining WBB ratchet, but with the difference, that in this design the withdrawal benefit base is never increased. Instead of permanently increasing the guaranteed withdrawal amount, on each policy calculation date where the current account value is greater than the current withdrawal benefit base, 50% of the difference is paid out immediately as a so-called "Performance Bonus", additionally to the base guaranteed amount of $x_{WL} \cdot P$. In contrast to the previous two designs, the minimum amount of the forthcoming guaranteed withdrawals remains unchanged in the Performance Bonus design. For the calculation of the withdrawal benefit base, only the base guaranteed amount $x_{WL} \cdot P$ is subtracted from the benefit base and not the performance bonus payments.

2.2 Model of the liabilities

For the sake of simplicity, we assume that all payments made to and by the policyholder, including fee payments, withdrawals, surrender benefits and death benefits, may only occur at prespecified points in time $\left\{\tilde{t}_i\right\}_{i\geq 0}$, expressed in terms of years from contract inception. We will refer to these dates as *contract (or policy) calculation dates*. Let the filtration $(\mathcal{F}_t)_{t\in\left\{\tilde{t}_i\right\}_{i\geq 0}}$ represent the corresponding information structure, where $\mathcal{F}_t$ is the information available at time t .

As a further simplification, we restrict the options that the policyholders have to two choices: At a contract calculation date $\tilde{t}_k$, they may either withdraw the then guaranteed withdrawal amount $W_{\tilde{t}_k}^{guar}$, or they may fully surrender the contract and withdraw all of the remaining fund assets.

At outset, the percentage φ^{acq} is deducted from the single premium P as acquisition fee and the remaining amount is invested in the selected funds, hence it holds for the initial account value $AV_0 = P \cdot (1 - \varphi^{acq})$.

Let AV_t^- denote the account value at time t after deduction of fees, but before withdrawals. Accordingly, AV_t^+ denotes the value of the account at time t after both, deduction of fees and withdrawals. We assume that at each end of a policy period administration and guarantee fees, in terms of percentages φ^{adm} and φ^{guar} , are deducted from the policyholder's account value. With S_t denoting the spot price of the portfolio's underlying fund at time t , the transition of the account value between two calculation dates $\tilde{t}_k$ and $\tilde{t}_{k+1}$ is given by

$$AV_{\tilde{t}_{k+1}}^- = AV_{\tilde{t}_k}^+ \cdot \frac{S_{\tilde{t}_{k+1}}}{S_{\tilde{t}_k}} \cdot e^{-\varphi^{adm} - \varphi^{guar}} \tag{2.1}$$

While being still alive, the policyholder withdraws an amount W_t from the account at each of the contract calculation dates following the outset of the contract. The random variable W_t is $\mathcal{F}_t$-measurable. Note that, although (depending on the guarantee of the contract) withdrawals may exceed the current account value, the account value itself may not drop below zero and is therefore given by

$$AV_t^+ = \max\left(AV_t^- - W_t, 0\right). \tag{2.2}$$

At each contract calculation date t , the maximum amount allowed to be withdrawn by the policyholder is the maximum out of the account value AV_t^- and the guaranteed withdrawal amount W_t^{guar} , which are both $\mathcal{F}_t$-measurable random variables. Therefore, a (full) surrender of the policyholder at a given contract calculation date t may be expressed as $W_t > W_t^{guar} \wedge W_t = AV_t^-$, with all of the following withdrawals being set to zero. The guarantee payments G_t due to be made by the insurer at time t are then given by the shortage resulting from the difference between guaranteed withdrawal amount and the remaining account value:

$$G_t := \left(W_t^{guar} - AV_t^-\right)^+ = \max\left(W_t^{guar} - AV_t^-, 0\right). \tag{2.3}$$

2.2.1 Transition at a policy calculation date

At a policy calculation date $\tilde{t}_k$, we have to distinguish the following cases:

The insured has died within the previous period $(\tilde{t}_{k-1}, \tilde{t}_k]$: If the insured has died within the previous policy period, the account value is paid out to the beneficiary as death benefit. With the payment of the death benefit, the insurance contract matures. Thus, the account value and the guaranteed withdrawal amount are set to zero.

The insured has survived the previous policy period and withdraws an amount within the limits of the withdrawal guarantee at time $\tilde{t}_k$: If the insured has survived the previous period, no death benefit is paid. For the withdrawal amount $W_{\tilde{t}_k}$, it holds $0 \leq W_{\tilde{t}_k} \leq W_{\tilde{t}_k}^{guar}$. The amount withdrawn is deducted from the account value as far as the funds suffice. After the account value has dropped to zero, the insurer bears the exceeding part of the guaranteed withdrawal amount. Thus, $AV_{\tilde{t}_k}^{+} = \max(AV_{\tilde{t}_k}^{-} - W_{\tilde{t}_k}, 0)$.

The insured has survived the previous policy year and at the policy anniversary withdraws an amount exceeding the limits of the withdrawal guarantee : In this case again, no death benefits are paid. For the sake of brevity, we only give the formulae for the case of full surrender, i.e. the case in which $W_{\tilde{t}_k} = AV_{\tilde{t}_k}^{-} > W_{\tilde{t}_k}^{guar}$, since partial surrender is not analyzed in what follows. In case of full surrender, the complete account value is withdrawn, therefore we set $AV_{\tilde{t}_k}^{+} = 0$ and the contract terminates.

2.2.2 Computation of the guaranteed withdrawal amount W^{guar}

For all three product designs that make use of the withdrawal benefit base as an auxiliary variable for computing the guaranteed withdrawal amount, we have two variables at time t: WBB_t^{-}, the withdrawal benefit base before the withdrawal of W_t takes place, and the variable WBB_t^{-}, which denotes the withdrawal benefit base right after the deduction of W_t. At outset, the withdrawal benefit base is set to the single premium P paid by the policyholder, i.e. $WBB_0^{+} = P$, and the initial

guaranteed withdrawal amount is set to $W_0^{guar} = x_{WL} \cdot P$.
At a contract calculation date $\tilde{t}_k > 0$, the computation of the guaranteed withdrawal amount $W_{\tilde{t}_k}^{guar}$ works as described below for each ratchet mechanism:

- **No Ratchet:**

$$W_{\tilde{t}_k}^{guar} = x_{WL} \cdot P \tag{2.4}$$

- **Lookback Ratchet:**

$$WBB_{\tilde{t}_k}^{-} = \max(WBB_{\tilde{t}_{k-1}}^{+}, AV_{\tilde{t}_k}^{-}) \tag{2.5}$$

$$W_{\tilde{t}_k}^{guar} = x_{WL} \cdot WBB_{\tilde{t}_k}^{-} \tag{2.6}$$

$$WBB_{\tilde{t}_k}^{+} = \max(WBB_{\tilde{t}_k}^{-} - W_{\tilde{t}_k}, 0) \tag{2.7}$$

- **Remaining WBB Ratchet:**

$$WBB_{\tilde{t}_k}^{-} = \max(WBB_{\tilde{t}_{k-1}}^{+}, AV_{\tilde{t}_k}^{-}) \tag{2.8}$$

$$W_{\tilde{t}_k}^{guar} = W_{\tilde{t}_{k-1}}^{guar} + x_{WL} \cdot \max(AV_{\tilde{t}_k}^{-} - WBB_{\tilde{t}_{k-1}}^{+}, 0) \tag{2.9}$$

$$WBB_{\tilde{t}_k}^{+} = \max(WBB_{\tilde{t}_k}^{-} - W_{\tilde{t}_k}, 0) \tag{2.10}$$

- **Performance Bonus:**

$$WBB_{\tilde{t}_k}^{-} = WBB_{\tilde{t}_{k-1}}^{+} \tag{2.11}$$

$$W_{\tilde{t}_k}^{guar} = x_{WL} \cdot P + \frac{1}{2} \max(AV_{\tilde{t}_k}^{-} - WBB_{\tilde{t}_{k-1}}^{+}, 0) \tag{2.12}$$

$$WBB_{\tilde{t}_k}^{+} = \max(WBB_{\tilde{t}_k}^{-} - x_{WL} \cdot P, 0) \tag{2.13}$$

2.3 Contract valuation framework

The valuation framework in this section follows in some parts the one used in Bacinello, Biffis and Millossovich (2009, [3]) and in others Bauer, et. al. (2008, [2]). We take as given a filtered probability space $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$, in which $\mathbb{P}$ is the real-world (or physical measure) and $\mathbb{F} = (\mathcal{F}_t)_{t \geq 0}$ is a filtration with $\mathcal{F}_0 = \{\emptyset, \Omega\}$ and $\mathcal{F}_t \subset \mathcal{F} \; \forall \, t \geq 0$. We assume that trading takes place continuously over time and without any transaction costs or spreads. Furthermore we assume that the price processes of the traded assets in the market are adapted and of bounded variation. With the assumed absence of arbitrage, a probability measure $\mathbb{Q}^*$ exists which is equivalent to $\mathbb{P}$ and under which the gain from holding a traded asset is a $\mathbb{Q}^*$-martingale after discounting by the chosen numéraire process, the money-market account. We call $\mathbb{Q}^*$ the equivalent martingale measure (EMM). Details on the derivation and the existence of the EMM $\mathbb{Q}^*$ subject to the used financial market model are given in chapter 3.

Let τ_X denote the date of the insured's death, expressed in terms of years since policy inception. Similarly, let i_X denote the index of the contract calculation date immediately following the death of the insured. Let $\mathbb{H}$ denote the filtration generated by the process $N_t = 1_{\tau_X \leq t}$, which is zero as long as the insured is alive and which jumps to one in the moment the insured dies. With $\mathbb{G}$ being the enlargement of the filtration $\mathbb{F}$ in order to include $\mathbb{H}$, i.e. $\mathbb{G} = \mathbb{F} \vee \mathbb{H}$, it follows that τ_X is a $\mathbb{G}$-stopping time, since

$$\{\omega \in \Omega : \tau_X(\omega) < t\} \subset \mathcal{G}_t \tag{2.14}$$

holds. Assuming independence between financial markets and mortality as well as risk-neutrality of the insurer with respect to mortality risk, we are able to use the product measure of the risk-neutral measure of the financial market and the mortality measure. In what follows, we denote this product measure by $\mathbb{Q}$ and use the enlargement $(\Omega, \mathcal{G}, \mathbb{G}, \mathbb{Q})$ of the filtered probability space $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{Q}^*)$.

We denote by x_0 the insured's age (in full years) at the time of the start of the contract, and with $X = \tau_X + x_0$ the age of the insured at the time of their death. Further denote ${}_tp_{x_0}$ the probability under $\mathbb{Q}$ for a x_0-year old to survive the next t years, ${}_sq_{x_0+t}$ the probability under $\mathbb{Q}$ for a x_0+t-year old to die within s years, and let ω be the limiting age of the mortality table, i.e. the age beyond which survival is assumed to be impossible. The probability that an insured aged x_0 at inception passes away within the period $(t, s]$ is thus given by the product ${}_tp_{x_0} \cdot {}_sq_{x_0+t}$. The limiting age ω allows for a finite time horizon $T = \omega - x_0$. The policy calculation dates are therefore given by $0 = \tilde{t}_0 < \tilde{t}_1 < \ldots < \tilde{t}_M = T$, with $M \in \mathbb{N}$.

We restrict the policyholder behavior to the simple case where the policyholder (while being still alive) has two possibilities to chose from at each policy calculation date t:

- either they withdraw exactly the guaranteed amount W_t^{guar}
- or they surrender the contract, i.e. they withdraw $AV_t^- > W_t^{guar}$.

This means we do not allow for partial surrender nor a withdrawal amount lower than the guaranteed amount. We denote the policy calculation date at which the policyholder surrenders with τ_S , where $\tau_S = \infty$ is to be interpreted as though the policyholder did not chose to surrender the contract while the contract was still in force. Let i_S denote the corresponding index of the policy calculation date τ_S , such that $\tau_S = \tilde{t}_{i_S}$ holds, given $\tau_S < \infty$.

The contract calculation date at which the account value of the policyholder falls below the guaranteed withdrawal amount for the first time and thus the date at which the insurer has to compensate for the guaranteed withdrawals for the first time, may then be expressed via the random variable τ_G , which is given by the formula

$$\tau_G := \inf\left\{ t \in \{\tilde{t}_i\}_{i=0}^{M} \,\middle|\, t < \tau_X,\ t < \tau_S,\ AV_t^- < W_t^{guar} \right\}. \tag{2.15}$$

As usually, the infimum of the empty set is defined as $+\infty$, which in that case implies that the guarantee of the contract does not trigger while the insured is still alive or before they chose to surrender. Let i_G denote the corresponding index of the contract calculation date τ_G , such that $\tau_G = \tilde{t}_{i_G}$ holds, given $\tau_G < \infty$. In the following, we will sometimes refer to τ_G as the "trigger time" of the guarantee. After the guarantee has triggered, for the policyholder there is no reason to surrender the contract, hence it holds $\tau_S < \infty \Rightarrow \tau_S < \tau_G$. However, for the sake of simplicity, we allow for (theoretical) surrender after the guarantee has triggered and will interpret this as if the policyholder did not surrender. Both, τ_S and τ_G , are stopping times with respect to the filtration $\mathbb{G} = (\mathcal{G}_t)_{t \in \{\tilde{t}_i\}_{i \geq 0}}$, i.e. it holds $\forall\ i \in \mathbb{N}$

$$\left\{\omega \in \Omega : \tau_G(\omega) < \tilde{t}_i\right\} \subset \mathcal{G}_{\tilde{t}_i} \tag{2.16}$$

$$\left\{\omega \in \Omega : \tau_S(\omega) < \tilde{t}_i\right\} \subset \mathcal{G}_{\tilde{t}_i}\ . \tag{2.17}$$

With B_t denoting the value at time t of the cash-bond (the chosen numéraire), the time- t value G_t^P of the deflated future guarantee payments G_t (see formula 2.3) of the insurer is given by

$$G_t^P = \begin{cases} 0 & ,\tau_G = \infty \\ B_t \cdot \sum_{i=i_G}^{i_X} \frac{G_{\tilde{t}_i}}{B_{\tilde{t}_i}} \cdot 1_{\{\tilde{t}_i > t\}} & ,\tau_G < \infty \end{cases} \tag{2.18}$$

$$= \begin{cases} 0 & ,\tau_G = \infty \\ B_t \cdot \left(\frac{(W_{\tau_G}^{guar} - AV_{\tau_G}^-)}{B_{\tau_G}} 1_{\{\tau_G > t\}} + \sum_{i=i_G+1}^{i_X} \frac{W_{\tilde{t}_i}^{guar}}{B_{\tilde{t}_i}} 1_{\{\tilde{t}_i > t\}} \right) & ,\tau_G < \infty \end{cases} . \tag{2.19}$$

The time- t value G_t^F of all future guarantee fees deducted from the account value can be computed similarly. We assume that the fraction of the total amount of running fees (for administration and guarantee) that is assigned for covering the costs of the guarantee is given by $\frac{\varphi^{guar}}{\varphi^{adm}+\varphi^{guar}}$ [1]. Thus, G_t^F takes the form

$$G_t^F = B_t \cdot \frac{\varphi^{guar}}{\varphi^{adm}+\varphi^{guar}} \cdot \left(1 - e^{-(\varphi^{adm}+\varphi^{guar})}\right) \cdot \sum_{i=1}^{\min(i_G, i_X, i_S)} \frac{1}{B_{\tilde{t}_i}} \cdot AV_{\tilde{t}_{i-1}}^+ \cdot \frac{S_{\tilde{t}_i}}{S_{\tilde{t}_{i-1}}} \cdot 1_{\{\tilde{t}_i > t\}} \tag{2.20}$$

$$= B_t \cdot \frac{\varphi^{guar}}{\varphi^{adm}+\varphi^{guar}} \cdot \left(1 - e^{-(\varphi^{adm}+\varphi^{guar})}\right) \cdot \sum_{i=1}^{\min(i_G, i_X, i_S)} \frac{1}{B_{\tilde{t}_i}} \cdot AV_{\tilde{t}_i}^- \cdot e^{\varphi^{adm}+\varphi^{guar}} \cdot 1_{\{\tilde{t}_i > t\}} \tag{2.21}$$

$$= B_t \cdot \frac{\varphi^{guar}}{\varphi^{adm}+\varphi^{guar}} \cdot \left(e^{\varphi^{adm}+\varphi^{guar}} - 1\right) \cdot \sum_{i=1}^{\min(i_G, i_X, i_S)} \frac{1}{B_{\tilde{t}_i}} \cdot AV_{\tilde{t}_i}^- \cdot 1_{\{\tilde{t}_i > t\}} . \tag{2.22}$$

We define the time- t value V_t^G of the GLWB option as the value of all (deflated) future guarantee payments occurring after time t, less the value of all (deflated) future guarantee fees deducted from the policyholder's account value, again only those occurring after time t. In formulas,

$$V_t^G = \mathbb{E}_{\mathbb{Q}}\left[G_t^P \middle| \mathcal{G}_t\right] - \mathbb{E}_{\mathbb{Q}}\left[G_t^F \middle| \mathcal{G}_t\right] . \tag{2.23}$$

At outset, we call the contract (in an actuarial sense) *fair* from an insurer's point

[1] This is motivated by the fact that $\int_0^T \varphi^{guar} \cdot \exp(-(\varphi^{adm} + \varphi^{guar})t)dt = \frac{\varphi^{guar}}{\varphi^{adm}+\varphi^{guar}} \cdot \left(1 - \exp(-(\varphi^{adm} + \varphi^{guar})T)\right)$.

of view, if the GLWB option's value V_0^G is zero. That is, to make the contract fair, the following equation must hold at inception of the contract:

$$\mathbb{E}_{\mathbb{Q}}\left[G_0^P\right] \stackrel{!}{=} \mathbb{E}_{\mathbb{Q}}\left[G_0^F\right] \ . \tag{2.24}$$

Finally, we assume that the policyholder surrenders the contract at each contract calculation date t with a certain probability p_t^S conditional to the insured being alive, the contract being still in force and an account value above zero. The probability that the policyholder surrenders at the contract calculation date $\tilde{t}_i$ (again under the assumption of an untriggered guarantee and the insured being still alive) is therefore given by $\left(\prod_{j=0}^{i-1}(1-p_{\tilde{t}_j}^S)\right) p_{\tilde{t}_i}^S$. We will denote this probability by $\tilde{p}_i^S$.

Then the computation of the time-0 value of the GLWB option, V_0^G, allows for the following conditionings of the expected value under $\mathbb{Q}$:

$$\begin{aligned}
V_0^G &= \mathbb{E}_{\mathbb{Q}}\left[G_0^P - G_0^F\right] && (2.25)\\
&= \mathbb{E}_{\mathbb{Q}}\left[\mathbb{E}_{\mathbb{Q}}\left[G_0^P - G_0^F \middle| \tau_X\right]\right] && (2.26)\\
&= \mathbb{E}_{\mathbb{Q}}\left[\mathbb{E}_{\mathbb{Q}}\left[\mathbb{E}_{\mathbb{Q}}\left[G_0^P - G_0^F \middle| \tau_S\right] \middle| \tau_X\right]\right] && (2.27)\\
&= \mathbb{E}_{\mathbb{Q}}\left[\mathbb{E}_{\mathbb{Q}}\left[\mathbb{E}_{\mathbb{Q}}\left[G_0^P - G_0^F \middle| \tau_S\right] \middle| i_X\right]\right] && (2.28)\\
&= \sum_{i=1}^{M} {}_{\tilde{t}_{i-1}}p_{x_0} \cdot {}_{\tilde{t}_i-\tilde{t}_{i-1}}q_{x_0+\tilde{t}_{i-1}} \cdot \mathbb{E}_{\mathbb{Q}}\left[\mathbb{E}_{\mathbb{Q}}\left[G_0^P - G_0^F \middle| \tau_S\right] \middle| i_X = i_x\right] && (2.29)\\
&= \sum_{i=1}^{M} {}_{\tilde{t}_{i-1}}p_{x_0} \cdot {}_{\tilde{t}_i-\tilde{t}_{i-1}}q_{x_0+\tilde{t}_{i-1}} \cdot \left[\sum_{i_s=0}^{i_x-1} \tilde{p}_{i_s}^S \cdot \mathbb{E}_{\mathbb{Q}}\left[G_0^P - G_0^F \middle| i_S = i_s, i_X = i_x\right]\right.\\
&\quad \left. + \left(1 - \sum_{i_s=0}^{i_x-1} \tilde{p}_{i_s}^S\right) \cdot \mathbb{E}_{\mathbb{Q}}\left[G_0^P - G_0^F \middle| \tau_S = \infty, i_X = i_x\right]\right] \ . && (2.30)
\end{aligned}$$

Chapter 3

Financial Market and Numerical Analysis Framework

3.1 Models of the financial market

For our analyses we assume two primary traded assets: the fund's underlying, whose spot price at time t we denote by S_t, and the money-market account, whose value at time t is denoted by B_t. The price processes $B = (B_t)_{0 \le t \le T}$ and $S = (S_t)_{0 \le t \le T}$ are assumed to be adapted, right-continuous with left-limits ("càdlàg"[1]) and strictly positive semi-martingales on a filtered probability space $(\Omega, \mathcal{F}, \mathbb{F}, \mathbb{P})$ with $\mathbb{F} = (\mathcal{F})_{0 \le t \le T}$ and a fixed time horizon $T \in (0, \infty)$. Furthermore, we assume the market to be free of arbitrage, in the sense of being equivalent (cf. Bingham and Kiesel, 2004, [5]) to the existence of an equivalent martingale measure (EMM). However, we do not assume the market to be complete in all cases, i.e. there may be contingent claims that are non-attainable by self-financing strategies.

We assume the interest rate spread to be zero and the money-market account to evolve at a continuously compounded risk-free rate r, which remains constant over time. Thus, the dynamics of the money-market account are given by

$$dB_t = rB_t dt \tag{3.1}$$

and thus

$$B_t = B_0 \exp(rt) \ . \tag{3.2}$$

[1] French "continue à droite, limitée à gauche".

For the dynamics of the process of the underlying spot price, $S = (S_t)_{0\leq t\leq T}$, we use two different models: first we assume the equity volatility to be deterministic and constant over time, and hence use the Black-Scholes-Merton (Black and Scholes, 1973, [6], Merton, 1973) model. Second, to allow for a stochastic equity volatility modeling, we use the Heston (1993, [16]) model, in which both, the underlying itself and its instantaneous (or *local*) variance, are modeled by stochastic processes.

3.1.1 Black-Scholes-Merton model

In the Black-Scholes-Merton framework (Black and Scholes, 1973, [6], Merton, 1973), the underlying's spot price S follows a geometric Brownian motion whose dynamics under the real-world measure $\mathbb{P}$ are given by the following stochastic differential equation (SDE)

$$dS_t = \mu S_t dt + \sigma_{BS} S_t dW_t, \quad S_0 \geq 0 \tag{3.3}$$

where μ is the (constant) drift rate of the underlying, $\sigma_{BS} \geq 0$ its constant volatility and $W = (W_t)_{0\leq t\leq T}$ denotes a $\mathbb{P}$-Wiener process. By Itō's lemma, S has the solution

$$S_t = S_0 \exp\left(\left(\mu - \frac{\sigma_{BS}^2}{2}\right) t + \sigma_{BS} W_t\right), \quad S_0 \geq 0. \tag{3.4}$$

3.1.2 Heston model

There are various extensions to the Black-Scholes-Merton model that aim at a more realistic modeling of the underlying's volatility. We use the Heston (1993, [16]) model in our analyses where the instantaneous (or local) variance of the asset is modeled stochastic. Under the Heston model, the market is assumed to be driven by two stochastic processes: the underlying's price $S = (S_t)_{0\leq t\leq T}$, and its instantaneous (local) variance $V = (V_t)_{0\leq t\leq T}$, which is assumed to follow a one-factor mean-reverting square-root process identical to the one used in the Cox-Ingersoll-Ross (1985, [10]) interest rate model. The dynamics of the two processes under the real-world measure $\mathbb{P}$ are given by the following system of stochastic differential equations,

$$dS_t = \mu S_t dt + \sqrt{V_t} S_t \left(\rho dW_t^1 + \sqrt{1-\rho^2} dW_t^2\right), \qquad S_0 \geq 0 \tag{3.5}$$

$$dV_t = \kappa (\theta - V_t) dt + \sigma_V \sqrt{V_t} dW_t^1, \qquad V_0 \geq 0, \tag{3.6}$$

where μ again is the drift of the underlying, V_t is the *instantaneous* (or *local*) variance of the underlying's spot price process at time t, κ is the speed of mean reversion, θ is the long-term average variance, σ_V is the so-called vol of vol, or (more precisely) the volatility of the variance, ρ denotes the correlation between the processes of the underlying and the variance, and $W^{1/2} = \left(W_t^{1/2}\right)_{0\leq t\leq T}$ are two independent $\mathbb{P}$-Wiener processes.

The condition $2\kappa\theta \geq \sigma_V^2$ ensures that the variance process will remain strictly positive almost surely (cf. Cox, Ingersoll, Ross, 1985 [10]).

To our knowledge, there is no analytical solution for S available, thus numerical methods are used in the simulation, which are presented in section 3.3.1.

3.2 Change of measure: the equivalent martingale measure

In order to determine the values (i.e. the risk-neutral expectations) of the assets in our model, we need to transform the real-world measure $\mathbb{P}$ into its risk-neutral counterpart $\mathbb{Q}^*$, i.e. into an equivalent measure (that is, both measures share the same null sets, i.e. same things possible and same things impossible) under which the process of the deflated (by the natural numeréraire B) underlying's spot price is a (local) martingale. While the transformation to such a measure is unique under the Black-Scholes-Merton model, it is not under the Heston model.

3.2.1 Black-Scholes-Merton model

If no dividends are paid on the underlying, by Itō's formula, the dynamics of the deflated underlying's price $\tilde{S} = \left(\tilde{S}_t\right)_{0\leq t\leq T}$, $\tilde{S}_t := S_t/B_t$, under the Black-Scholes-Merton model is given by the following SDE (cf. Bingham and Kiesel, 2004, [5]):

$$d\tilde{S}_t = (\mu - r)\tilde{S}_t dt + \sigma_{BS}\tilde{S}_t dW_t,\ S_0 \geq 0 \tag{3.7}$$

where r denotes the risk-free rate of return, μ the return of the underlying and $W = (W_t)_{0\leq t\leq T}$ is a Wiener process under the real-world measure $\mathbb{P}$.

Let $\gamma = (\gamma(t))_{0\leq t\leq T}$ be a measurable, adapted process with $\int_0^T \gamma(t)^2 dt < \infty$ almost surely. We then define the process $L = (L(t))_{0 \le t \le T}$ via

$$L(t) = \exp\left\{ -\int_0^t \gamma(s)' dW_s - \frac{1}{2}\int_0^t \gamma(s)^2 ds \right\}. \tag{3.8}$$

Now assume γ fulfills *Novikov's condition* (again cf. Bingham and Kiesel, 2004, [5]), i.e.

$$E\left(\exp\left\{ \frac{1}{2}\int_0^t \gamma(s)^2 ds \right\} \right) < \infty \tag{3.9}$$

holds, which implies that L is a (continuous) martingale. Girsanov's theorem then shows that the process $\tilde{W}_t := W_t + \int_0^t \gamma(s) ds,\ 0 \le t \le T$, is a Wiener process under the equivalent probability measure $\tilde{\mathbb{P}}$ (defined on $(\Omega, \mathcal{F}_T)$) with the Radon-Nikodým derivative

$$\frac{\tilde{\mathbb{P}}}{\mathbb{P}} = L(T). \tag{3.10}$$

If $\mathbb{F} = (\mathcal{F}_t)_{0 \le t \le T}$ is the Brownian filtration (the filtration generated by the Wiener process), any pair of equivalent probability measures $\mathbb{Q}^* \sim \mathbb{P}$ on $\mathcal{F}_T$ is a *Girsanvov pair*, i.e.

$$\left.\frac{\mathbb{Q}^*}{\mathbb{P}}\right|_{\mathcal{F}_t} = L(t). \tag{3.11}$$

With that, the dynamics for $\tilde{S}$ under the equivalent measure $\mathbb{Q}^*$ are

$$\tilde{S}_t = (\mu - r - \sigma_{BS}\gamma(t))\, \tilde{S}_t dt + \sigma_{BS}\tilde{S}_t d\tilde{W}_t,\ S_0 > 0. \tag{3.12}$$

Since $\tilde{S}$ has to be a (local) martingale under $\mathbb{Q}^*$, the drift part of the SDE must be zero, i.e.

$$(\mu - r - \sigma_{BS}\gamma(t)) \overset{!}{=} 0, \quad 0 \le t \le T, \tag{3.13}$$

which implies that $\gamma(t)$ is a constant and given by the following formula (also known as the "market price of risk"

$$\gamma(t) \equiv \frac{\mu - r}{\sigma_{BS}}. \tag{3.14}$$

We then arrive at the following $\mathbb{Q}^*$-dynamics for the undiscounted price of the underlying, S:

$$S_t = rS_t dt + \sigma_{BS} S_t d\tilde{W}_t,\ S_0 > 0, \tag{3.15}$$

with $\tilde{W}$ being a $\mathbb{Q}^*$-Wiener process.

3.2.2 Heston model

Under the Heston model, as there are two sources of risk, W^1 and W^2, there are also two "market price of risk" processes, denoted by γ_1 and γ_2 (corresponding to W^1 and W^2).

It is known (for all of the following we refer to Wong and Heyde, 2006, [31]) that the (class of) equivalent martingale measures $\mathbb{Q}^*$, if existent, can be considered in terms of the Radon-Nikodým derivative

$$\frac{\mathbb{Q}^*}{\mathbb{P}}\Big|_{\mathcal{F}_T} = L(T) \tag{3.16}$$

$$= \exp\Bigg\{ - \left(\int_0^T \gamma_1(u) dW_u^1 + \int_0^T \gamma_2(u) dW_u^2 \right) \tag{3.17}$$

$$- \frac{1}{2} \left(\int_0^T \gamma_1^2(u) du + \int_0^T \gamma_2^2(u) du \right) \Bigg\} . \tag{3.18}$$

A necessary condition to ensure that the discounted underlying price is a local martingale and therefore a necessary condition for an equivalent local martingale measure to exist, is given by the equation

$$\mu - r = \sqrt{V_t} \left(\rho\gamma_1(t) + \sqrt{1-\rho^2}\gamma_2(t) \right) . \tag{3.19}$$

Heston (1993, [16]) proposed the following additional restriction on the *market-price-of-volatility-risk* process, assuming it to be linear in local volatility,

$$\gamma_1(t) = \lambda\sqrt{V_t} , \tag{3.20}$$

where $\lambda \in \mathbb{R}$ is a real-valued constant.

With this, we know that the variance process under $\mathbb{Q}^*$ has the dynamics

$$
\begin{aligned}
dV_t &= \kappa\theta dt - (\kappa + \lambda\sigma_V) V_t dt + \sigma_V \sqrt{V_t} d\tilde{W}_t^1, V_0 > 0, && (3.21)\\
&= \kappa^* (\theta^* - V_t) dt + \sigma_V \sqrt{V_t} d\tilde{W}_t^1, \qquad V_0 > 0, && (3.22)
\end{aligned}
$$

with $\tilde{W}^1$ being a $\mathbb{Q}^*$-Wiener process.

Hence we obtain the same functional form as under $\mathbb{P}$, but with transformed parameters for the speed of mean reversion and long-term average variance,

$$
\begin{aligned}
\kappa^* &= \kappa + \lambda\sigma_V && (3.23)\\
\theta^* &= \frac{\kappa\theta}{\kappa + \lambda\sigma_V} \; . && (3.24)
\end{aligned}
$$

For the second market-price-of-risk process γ_2, we obtain by substituting 3.20 into equation (3.19) and under the aforementioned parameter restriction $2\kappa\theta \geq \sigma_V^2$, the equation

$$
\gamma_2(t) = \frac{1}{\sqrt{1-\rho^2}} \left(\frac{\mu - r}{\sqrt{V_t}} - \lambda\rho\sqrt{V_t} \right) \; , \qquad (3.25)
$$

and thus, if λ is given, both market-price-of-risk processes are fully defined.

Wong and Heyde (2006, [31]) also show that the equivalent local martingale measure that corresponds to the market price of volatility risk $\lambda\sqrt{V_t}$, exists if the inequality

$$
-\frac{\kappa}{\sigma_V} \leq \lambda < \infty \qquad (3.26)
$$

is fulfilled. They further show that, if an equivalent local martingale measure $\mathbb{Q}^*$ exists and

$$
\kappa + \lambda\sigma_V \geq \sigma_V \rho \qquad (3.27)
$$

holds, the discounted stock price $\tilde{S} = \left(\tilde{S}_t\right)_{0 \leq t \leq T}$, $\tilde{S}_t := S_t / B_t$ is a $\mathbb{Q}^*$-martingale.

Finally, provided both measures, $\mathbb{P}$ and $\mathbb{Q}^*$, exist, the $\mathbb{Q}^*$-dynamics of S and V, again under the assumption that no dividends are paid, are given by (cf. Wong and Heyde, 2006, [31])

$$
\begin{aligned}
dS_t &= rS_t dt + \sqrt{V_t} S_t \left(\rho d\tilde{W}_t^1 + \sqrt{1-\rho^2} d\tilde{W}_t^2 \right), \qquad S_0 \geq 0 && (3.28)\\
dV_t &= \kappa^* (\theta^* - V_t) dt + \sigma_V \sqrt{V_t} d\tilde{W}_t^1, \qquad V_0 \geq 0, && (3.29)
\end{aligned}
$$

where $\tilde{W}^{1/2}$ are two independent $\mathbb{Q}^*$-Wiener processes and where κ^* and θ^* are the risk-neutral counterparts to κ and θ as defined in (3.23) and (3.24) respectively.

3.3 Numerical analysis framework

3.3.1 Monte-Carlo simulations

As, to our knowledge, there exist no closed form solutions for the in section 2.3 defined value of the GLWB rider option V_0^G, we have to rely on numerical methods to determine the option's value, i.e. the value of the difference between expected discounted future guarantee payments made by the insurer and the expected discounted future guarantee fees deducted from the policyholder's fund assets. We call the contract fair, if this difference is zero at inception of the contract. To this end, for both equity models, we use Monte-Carlo simulations to compute the value of the GLWB option as defined in section 2.3, that is to compute the expectation $V_0^G = E_{\mathbb{Q}^*}\left[G_0^P - G_0^F\right]$ we use the Monte-Carlo approximation / estimator for V_0^G (cf. Glasserman, 2003, [14]),

$$\hat{V}_0^G = \frac{1}{N}\sum_{k=1}^{N}\left(G_0^P\left(\hat{S}^k\right) - G_0^F\left(\hat{S}^k\right)\right) \tag{3.30}$$

where each $\hat{S}^k = \left(\hat{S}_i^k\right)_{0\leq i\leq M}$ is one path of the fund's underlying price process at the policy calculation dates $\left(\tilde{t}_i\right)_{0\leq i\leq M}$.

a) Path generation

As there is a closed form solution for S available under the Black-Scholes-Merton model, simulation of the financial market, i.e. path generation of S is rather simple: Let $z_i^k,\ 0\leq i\leq M,\ 0\leq k\leq N$ be a realization of independent standard-normal distributed random variables $Z_i^k \sim N(0,1),\ 0\leq i\leq M,\ 0\leq k\leq N$. Then the path generation for one path $\hat{S}^k = \left(\hat{S}_i^k\right)_{0\leq i\leq M}$ in the Black-Scholes-Merton case works as follows (cf. Glasserman, 2003, [14]),

$$\hat{S}_0^k = S_0 \tag{3.31}$$

and

$$\hat{S}_{i+1}^k = \hat{S}_i^k \exp\left(\left(\mu - \frac{1}{2}\sigma_{BS}^2\right)\left(\tilde{t}_{i+1} - \tilde{t}_i\right) + \sigma_{BS}\sqrt{\tilde{t}_{i+1} - \tilde{t}_i}\; Z_i^k\right) . \tag{3.32}$$

In the Heston model however, as there is no closed-form solution for S, numerical methods are needed for the path generation of $\hat{S}^k$. We follow Lord, Koekkoek and van Dijk (2008, [25]) in the remainder of this section.

A method to simulate paths for the Heston model without bias, using non-central chi-squared random variables and a numerical Fourier inversion of the characteristic function, has recently been derived by Broadie and Kaya (2004, 2006). However, the part of the numerical Fourier inversion makes the algorithm highly time-consuming in the case if many observation dates of the underlying's price are needed, i.e. if many intermediate steps need to be computed, as, for instance, it is the case with highly path-dependent products. Since there are indeed many intermediate steps needed for our simulations, for instance when we analyze and compare different hedging strategies, and as performance is an important issue with this kind of simulations, we do not use this unbiased algorithm for our simulations, but, instead, use methods that allow for a (comparably) fast, however biased generation of simulation paths of S.

To this end, we use Euler discretizations of the instantaneous variance process V, i.e. we use (very) small (typically equidistant) steps in time to approximate the dynamics of V given in its corresponding SDE. Let Δt be the chosen step size of our Euler discretization and let $(t_j)_{0\le j\le \tilde{N}}$ be the resulting time grid and $\left(\hat{V}_j\right)_{0\le j\le \tilde{N}}$ the corresponding simulated path[2] of V, then a naïve Euler discretization for V would read

$$\hat{V}_0 = V_0 \tag{3.33}$$

and

$$\hat{V}_{j+1} = \hat{V}_j + \kappa\left(\theta - \hat{V}_j\right)\Delta t + \sigma_V \sqrt{\hat{V}_j}\,\sqrt{\Delta t}\; z_j, \qquad 0 \le j \le \tilde{N} - 1 , \tag{3.34}$$

where the $(z_j)_{0\le j\le \tilde{N}}$ again are realizations of independent standard-normal distributed random variables.

The problem with the simulation scheme in (3.34) is, that, given $\tilde{V}_j > 0$, the proba-

[2]For ease of notation and better readability, we omitted the indication of the path number, $()^k$.

bility that $\tilde{V}_{j+1}$ will be negative is given by (cf. Lord, Koekkoek and van Dijk, 2008, [25]):

$$\mathbb{P}\left(\tilde{V}_{j+1} < 0\right) = \mathrm{N}\left(\frac{-\left(1-\kappa\Delta t\right)\tilde{V}_j - \kappa\theta\Delta t}{\sigma_V\sqrt{\tilde{V}_j}\sqrt{\Delta t}}\right) , \tag{3.35}$$

where N is the cumulative distribution function of the standard normal distribution. That is, the probability that we end up with a negative value for the local variance within our simulations is strictly positive within the proposed scheme in (3.34).

Lord, Koekkoek and van Dijk (2008, [25]) introduce a general framework for all of the common fixes for this problem. All of the considered Euler schemes can be unified within the scheme (with the auxiliary values $\left(\tilde{V}_j\right)_{0\leq j\leq \tilde{N}}$)

$$\hat{V}_0 = V_0 \tag{3.36}$$

$$\tilde{V}_0 = V_0 \tag{3.37}$$

and

$$\tilde{V}_{j+1} = f_1\left(\tilde{V}_j\right) - \kappa\Delta t\cdot\left(f_2\left(\tilde{V}_j\right) - \theta\right) + \sigma_V\sqrt{f_3\left(\tilde{V}_j\right)}\,\sqrt{\Delta t}\; z_j \tag{3.38}$$

$$\hat{V}_{j+1} = f_3\left(\tilde{V}_{j+1}\right) , \qquad 0 \leq j \leq \tilde{N}-1 . \tag{3.39}$$

Now, the considered schemes can be summarized as presented in table 3.1, where $|x|$ is the absolute value and $x^+ = \max(x, 0)$. For details and references, see Lord, Koekkoek and van Dijk (2008, [25]).

Scheme	$f_1(x)$	$f_2(x)$	$f_3(x)$
Absorption	x^+	x^+	x^+
Reflection	$\|x\|$	$\|x\|$	$\|x\|$
Higham and Mao	x	x	$\|x\|$
Partial truncation	x	x	x^+
Full truncation	x	x^+	x^+

Table 3.1: Overview of Euler schemes for simulation of the local volatility process under the Heston model.

Lord, Koekkoek and van Dijk (2008, [25]) demonstrate that using the correct fix at the boundary is important, and significantly impacts the magnitude of the bias. In their examples, the numerical results indicate that the full truncation scheme produces the smallest bias, being far superior to the (in practice widely used) absorption scheme. The most biased scheme according to their comparison is the reflection scheme with almost double the bias of the absorption scheme and about 100 times the bias of the full truncation scheme. Therefore, we use the proposed full truncation scheme in our simulations.

For the simulation of the underlying's price $\left(\hat{S}_j\right)_{0\leq j\leq\tilde{N}}$ along the chosen time grid $(t_j)_{0\leq j\leq\tilde{N}}$, we switch to logarithms, resulting in

$$\hat{S}_0 = S_0 \tag{3.40}$$

and (with $0 \leq j \leq \tilde{N}-1$),

$$\ln\hat{S}_{j+1} = \ln\hat{S}_j + \left(\mu - \frac{1}{2}\hat{V}_j\right)\Delta t + \sqrt{\hat{V}_j}\,\sqrt{\Delta t}\,\left(\rho z_j + \sqrt{1-\rho^2}\tilde{z}_j\right), \tag{3.41}$$

where z_j are the values used in the simulation of the local variance, and $\tilde{z}_j$ are another set of realizations of independent standard-normal distributed random variables.

We refer to Andersen (2007, [1]) and Van Haastrecht et. al. (2008, [29]) for more elaborate simulation schemes of the Heston model, that aim at finding the best mix between exact simulation and computational efficiency.

b) Variance reduction techniques

In order to reduce the variance of the Monte-Carlo estimator of an option value, one could use a higher number N of paths in the simulation, but this comes at the cost of a higher computational effort and, as the central feature of the Monte Carlo method is a standard deviation of the form $\sigma_{MC}/\sqrt{N}$ (cf. Glasserman, 2003, [14]), where four times the number of paths only result in a bisection of the standard deviation, this may prove inefficient. There are several alternative methods known how the variance of the Monte-Carlo estimator can be reduced, well known is for instance the use of the control variate technique, where (additionally to the option value of interest) the value of an option for which a closed-form solution exists is also priced within the Monte-Carlo simulation. The computed Monte-Carlo price of this option is then compared to its real value, given by the closed-form solution. The according deviation is then used to improve the estimator for the value of the option which is of interest.

Another technique is the use of so-called *antithetic variates* (cf. Glasserman, 2003, [14]), where the simulation of the paths is done in form of pairs in case of the Black-Scholes-Merton model and in form of quadruples in case of the Heston model. This method uses the fact, that a Wiener process $W = (W_t)_{0 \le t \le T}$ multiplied by -1 (i.e. reflected with respect to the time axis) again is a Wiener process, and therefore, if we have simulated one path of the Wiener process, we can use it in fact twice as the pair $(W, -W)$, resulting in a symmetrical distribution of the simulated paths, as the outcome of the simulation generated by the first path will be balanced by the outcome of the simulation result of the second path.

When the financial market in our model is driven by two Wiener processes, as it is the case with the Heston model, we can use the same approach to generate four scenarios out of one realization of each Wiener process, resulting in the quadruple $(\{W^1, W^2\}, \{W^1, -W^2\}, \{-W^1, W^2\}, \{-W^1, -W^2\})$.

An additional benefit of the antithetic variates method is a reduced computational effort, as in order to perform a Monte Carlo simulation with the same number of paths, the computational costs to produce realizations of the standard-normal distribution is reduced, because flipping the sign normally needs considerably less effort than calculating an "original" draw from the standard-normal distribution.

3.3.2 Computation of sensitivities (Greeks)

Where no analytical solutions for the sensitivity of the option's or guarantee's value to changes in model parameters (the so-called Greeks, see for instance Joshi (2003, [19]) for a detailed explanation) exist, we use Monte-Carlo methods to compute the respective sensitivities numerically. We use finite differences (cf. Glasserman, 2003, [14]) as approximations of the partial derivatives, where the direction of the shift is chosen accordingly to the direction of the risk, i.e. for the delta we shift the stock downwards in order to compute the *backward* finite difference, and for the vega we shift the current volatility upwards, this time to compute a *forward* finite difference.

3.3.3 Valuation of European standard options via Fourier inversion

In some of the hedging strategies considered in section 4, European standard or "*plain vanilla*" options are used. Under the Black-Scholes-Merton model, there are closed form solutions for the price of European call and put options (cf. e.g. Bingham and Kiesel, 2004, [5]). For strike price K and maturity T, the fair value of a call option at time t is given by the Black (1976) formula

$$C^{BS}\left(S_t, K, \sigma_{BS}, t, T\right) = P(t,T)\left[F(t,T) \cdot N(d_1) - K \cdot N(d_2)\right], \tag{3.42}$$

where $N(\cdot)$ denotes the cumulative distribution function of the standard normal distribution and d_1 and d_2 are given by

$$d_1 := \frac{\ln(F(t,T)/K) + (\sigma_{BS}^2/2)(T-t)}{\sigma_{BS}\sqrt{T-t}} \tag{3.43}$$

and

$$d_2 := d_1 - \sigma_{BS}\sqrt{T-t}\ . \tag{3.44}$$

Further, $F(t,T)$ denotes the forward price and $P(t,T)$ denotes the discount factor to the option expiry date. With μ being the underlying's drift under the risk-neutral measure, and r the risk-free rate of return, we have

$$F(t,T) := S_t \exp\left(\mu(T-t)\right) \tag{3.45}$$

and

$$P(t,T) := \exp\left(-r(T-t)\right)\ . \tag{3.46}$$

Similarly, the price of a European put option is given by

$$P^{BS}\left(S_t, K, \sigma_{BS}, t, T\right) = P(t,T)\left[K \cdot N(-d_2) - F(t,T) \cdot N(-d_1)\right]. \tag{3.47}$$

Pricing via Fourier inversion under the Heston model

For the Heston stochastic volatility model, Heston (1993, [16]) found a semi-analytical solution for pricing European call and put options using Fourier inversion techniques. The time-t prices for European call and put options with strike price K and maturity T can be expressed very similar to the Black-Scholes ones, namely (for the following cf. Kahl and Jackel, 2005, [21])

$$C^{\text{Heston}}(S_t, K, V_t, t, T)\ = P(t,T)\left[F(t,T) \cdot P_1 - K \cdot P_2\right] \tag{3.48}$$

and

$$P^{\text{Heston}}(S_t, K, V_t, t, T) = P(t,T)\left[F(t,T)\cdot(P_1 - 1) - K\cdot(P_2 - 1)\right], \tag{3.49}$$

with $F(t,T)$ and $P(t,T)$ as defined in (3.45) and (3.46) respectively, and with

$$P_{1/2} := \frac{1}{2} + \frac{1}{\pi}\int_0^\infty f_{1/2}(u)du\ . \tag{3.50}$$

The functions f_1 and f_2 are

$$f_1(u) := \text{Re}\left(\frac{e^{-iu\ln K}\varphi(u-i)}{iuF(t,T)}\right) \tag{3.51}$$

and

$$f_2(u) := \text{Re}\left(\frac{e^{-iu\ln K}\varphi(u)}{iu}\right)\ , \tag{3.52}$$

with $\varphi(\cdot)$ being defined as the log-characteristic function of the underlying's price S_T at expiry under the risk-neutral measure $\mathbb{Q}^*$, that is

$$\varphi(u) := E_{\mathbb{Q}^*}\left[e^{iu\ln S_T}\middle|S_t\right]\ . \tag{3.53}$$

The equations for the price of a call or put option given the log-characteristic function $\varphi(\cdot)$ of the underlying's price at expiry are generic and apply to any model. Specifically for the Heston model, with $\tau = T - t$, we have the log-characteristic function

$$\varphi(u) = e^{C(\tau,u)+D(\tau,u)V_t+iu\ln F(t,T)}\ , \tag{3.54}$$

where the coefficients C and D are solutions of a two-dimensional system of ordinary differential equations (ODE) of Riccati-type, and are given by

$$C(\tau,u) = \frac{\kappa\theta}{\sigma_V^2}\left(\left(\kappa - \rho\sigma_V ui + d(u)\right)\tau - 2\ln\left(\frac{c(u)e^{d(u)\tau}-1}{c(u)-1}\right)\right), \tag{3.55}$$

and

$$D(\tau,u) = \frac{\kappa - \rho\sigma_V ui + d(u)}{\sigma_V^2}\left(\frac{e^{d(u)\tau}-1}{c(u)e^{d(u)\tau}-1}\right), \tag{3.56}$$

with the auxiliary functions

$$c(u) = \frac{\kappa - \rho\sigma_V ui + d(u)}{\kappa - \rho\sigma_V ui - d(u)} \tag{3.57}$$

and

$$d(u) = \sqrt{(\rho\sigma_V ui - \kappa)^2 + iu\sigma_V^2 + \sigma_V^2 u^2}\ . \tag{3.58}$$

As Kahl and Jackel (2005, [21]) point out, the computation of the terms $P_{1/2}$ in (3.50) includes the evaluation of the logarithm with complex arguments in (3.55), which may lead to numerical instabilities for certain sets of parameters and/or long-dated options. For that reason, we use the scheme proposed in their paper, which is supposed to allow for a robust computation of the fair values of European call and put options for (practically) all levels of parameters and – as they state – even for maturities of many decades.

Kahl and Jackel also point out in their paper that the choice of the right integration scheme is another crucial point for a robust implementation of the semi-analytical Heston solution, since the integrands $f_{1/2}$ can vary in their shape from simply exponentially decaying to highly oscillatory depending on the choice of parameters. As in the proposed scheme, we use the adaptive Gauss-Lobatto quadrature method as described in Gander and Gautschi (2000, [13]) for the numerical integration of P_1 and P_2.

3.4 Black-Scholes-Merton Implied Volatility surface

The Black-Scholes-Merton *Implied Volatility* of an European standard option is the specific volatility σ_{BS} for which the result of the Black-Scholes-Merton formulas in (3.42) for a call option or (3.47) for a put option respectively coincide with the observed market prices of these options, i.e. the volatility implied by the market price of the option. For instance in the case of a call option with maturity T, strike set at K and a current level of the underlying S_t, this means that for a given time-t market price $C^{\text{market}}(S_t, K, t, T)$ of this option, the Black-Scholes-Merton implied volatility $\sigma_{BS}(S_t, K, t, T, r)$, with the (assumed to be observable) risk-free rate of return r, is defined as the unique[3] value for which the following equation holds:

$$C^{\text{market}}(S_t, K, t, T) = C^{\text{BS}}(S_t, K, \sigma_{BS}, t, T). \tag{3.59}$$

If we have a matrix of option prices of call and put options on the same underlying but for different maturities and different strikes, we call the corresponding set of Black-Scholes-Merton (BSM) implied volatilities, the *implied volatility surface*, i.e. the surface of the BSM volatilities implied by the observed market prices. For instance, if the market prices are computed with the Black-Scholes-Merton model using only one constant volatility σ_{BS} for all strikes and maturities, the resulting implied volatility surface would be flat.

In reality however, the implied surfaces observed in the market are not flat but show different kinds of dynamic "deformations" (cf. e.g Cont and Da Fonseca, 2002, [9], and Joshi, 2003, [19]) with the most prominent being the following:

- *skewness* or *smile*: typically put options with a low *moneyness* (i.e. the strike is significantly below the current level of the underlying) have a (much) higher implied volatility than options struck at-the-money, i.e. the strike price coincides with the current spot price of the underlying, or options with a strike above the current level of the underlying (in-the-money puts or out-of-the-money calls). Depending on the underlying, the implied volatilities may rise again with an increasing strike level. The emerging pattern is called the *smile* of the implied volatility surface.
- *term structure*: the implied volatilities observed in the market also usually change with the time to maturity of the options. Typically, the smile of the surface flattens and the volatilies around the current level of the underlying (or the forward value $F(t,T) = E_{\mathbb{Q}^*}\left[S_T|\mathcal{F}_t\right]$ hereof) rise or fall.

It is of interest to know the characteristics and dynamics of the implied volatility surface that a certain financial market model generates, i.e. the volatility surface which is implied by the prices of a given set of standard options computed within the model. Ideally, if the model is used for market-consistent pricing, these computed prices should coincide (within an error margin) with the observed market prices.

[3] In the Black-Scholes-Merton model, the Vega of a call option is given by $\frac{\partial C^{BS}}{\partial \sigma_{BS}} = S\sqrt{T-t}N'(d_1)$, which - by means of put-call parity and because the vega of a forward is zero - coincides with the Vega of the put option. Note that the Vega is always positive and therefore the map from volatilities to prices is injective (cf. Joshi, 2003, [19]).

Therefore, when confronted with the task of calibrating an equity model for pricing purposes, there are usually two main approaches on how to determine the parameters (cf. Wilmott, 2006, [30]): The statistical approach first looks at the statistics of the underlying and estimates the real-world model parameters from historical data of the underlying, then tries to estimate the market price of risk processes (in case of the Heston model, the parameter λ) from option price data in order to get the risk-neutral parameters. The second approach aims at the market consistency of the model and directly calibrates the risk-neutral parameters to option prices observable in the market, such that the model replicates these prices as accurately as possible. The model is then used to determine the values of options that are not observable in the market.

Figure 3.1 shows the implied volatility surface that is implied by the option prices generated by the Heston model with parameters as given in table 6.4 that we use later on for our analyses in chapters 6 and 7.

The two implied volatility surfaces pictured in figure 3.2 illustrate the dynamics of the implied volatility surface generated by the Heston model, as they show how the generated surface changes if the start value for the local variance, V_0, is changed to a higher or lower value than the long-term variance θ respectively.

The option prices were calculated via the Fourier inversion technique presented in section 3.3.3 and the Newton-Raphson iteration procedure (cf. Bingham and Kiesel, 2004, [5]) was used for calculation of the implied volatilities.

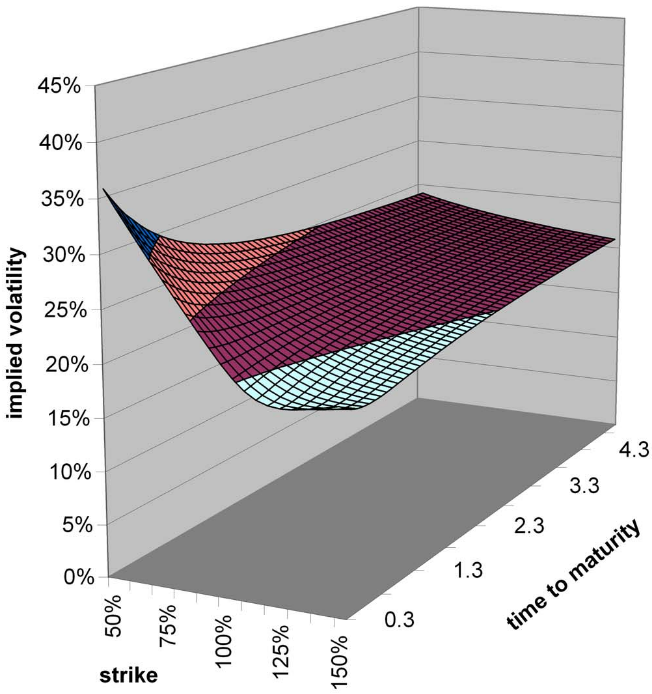
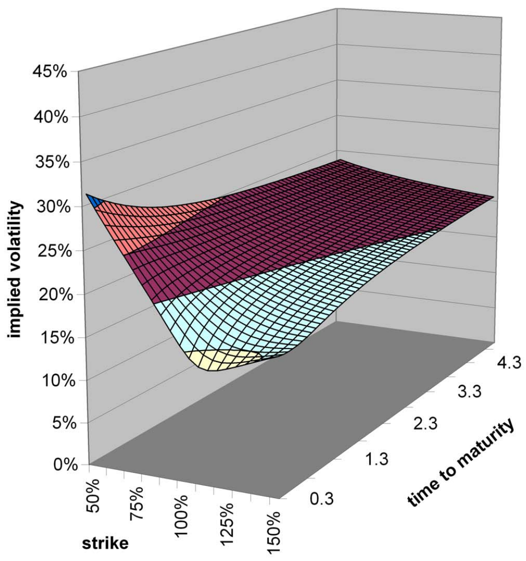
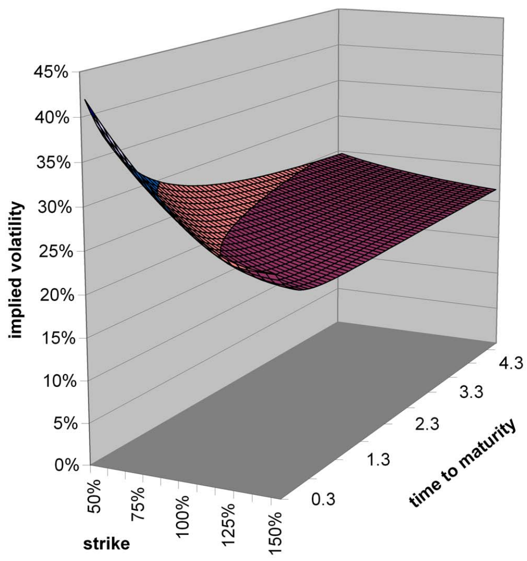


Figure 3.1: Implied volatility surface generated by the Heston model with the option's strike stated as percentage of the underlying's current spot price, and with model parameters $r = 0.04$, $\theta = 0.220^2$, $\kappa = 4.75$, $\sigma_V = 0.55$, $\rho = -0.569$ and $V_0 = \theta$.



(i) Lower start value for the local variance: $V_0 = 0.10^2$



(ii) Higher start value for the local variance: $V_0 = 0.35^2$

Figure 3.2: Implied volatility surfaces generated by the Heston model with the option's strike stated as percentage of the underlying's current spot price, with model parameters $r = 0.04$, $\theta = 0.220^2$, $\kappa = 4.75$, $\sigma_V = 0.55$, $\rho = -0.569$, and with $V_0 = 0.10^2$ used in (i) and $V_0 = 0.35^2$ used in (ii) respectively.

Chapter 4

Hedging

In this Section, we describe the different (dynamic) hedging strategies that we use for the hedging simulations whose results are presented in chapter 7. The presented strategies may be applied by the insurer in order to reduce the financial risk of the guarantees (and thereby reducing the required economic risk capital). We first describe the assumed structure of the (hedge) portfolio of the insurer before we describe the considered hedging strategies for both equity models presented in chapter 3, Black-Scholes-Merton and Heston, and for different assumptions regarding the used hedge instruments.

4.1 Hedge portfolio

We assume that the insurer has sold a pool of policies with GLWB guarantees. We denote by Ψ_t the cumulative (fair) option value for that pool at time t, i.e. Ψ_t is the sum of the individual (fair) option values at time t of all policies in that pool, where the (fair) option value is defined and to be calculated as presented in section 2.3.
Additionally, we assume that the insurer cannot influence the value Ψ_t of the policy pool by changes in the underlying fund, like, for instance, changing the fund's exposure to risky assets or forcing the policyholder to switch to a different, e.g. less volatile, fund[1].

[1]In practice, however, this is a frequently seen feature in products, which allows the insurer to hedge against volatility risk by means of product design, and effectively transfer volatility risk back to the policyholder. In our analyses, however, we concentrate on hedging strategies that aim at compensating the (externally generated) changes of an option's value.

We further assume that the insurer invests the guarantee fees in a hedge portfolio Π_t^{Hedge} and implements certain hedging strategies within this hedging portfolio. Also, if the guarantee of a contract is triggered, all of the subsequent guaranteed payments are made from this hedge portfolio. Thus, the insurer's cumulative profit/loss Π_t (in what follows sometimes just denoted as the insurer's profit) at time t that originates from the pool of policies and the implemented hedging strategy is given by

$$\Pi_t = -\Psi_t + \Pi_t^{\text{Hedge}} , \tag{4.1}$$

where Ψ_t is the the time-t value of the implied guarantees of the pool of policies and Π_t^{Hedge} is the time-t value of the corresponding hedge portfolio implemented by the insurer.

The following hedging strategies aim at reducing the insurer's risk by implementing certain investment strategies within the hedge portfolio Π_t^{Hedge}. Note that the value Ψ_t of the pool of policies at time t does not only depend on the number of the written contracts and their respective face value, but also on retrospective and prospective factors, such as - due to the path-dependent nature of the GLWB contracts - historical prices of the fund at previous withdrawal dates, and several model assumptions concerning mortality, lapsation, and the financial market.

The insurer's choice of the financial market model and the corresponding choice of parameters has a significant impact on the hedging strategies. Therefore, we will differentiate in the following between the hedging model that is chosen and used by the insurer, and the data-generating model that we use to simulate the development of the underlying and the market prices of European call and put options. This allows us, e.g., to analyze the impact on the insurer's risk situation if the insurer bases pricing and hedging on the Black-Scholes-Merton model (hedging model) with deterministic equity volatility, whereas in reality (data-generating model) equity volatility is stochastic. We assume the value of the guarantee to be marked-to-model, where the same model is used for valuation as the insurer uses for hedging. All other assets in the insurer's portfolio are marked-to-market, i.e. their prices are determined by the (external) data-generating model.

We assume that, additional to the fund's underlying and the money-market account, a market for European "plain vanilla" options on the underlying (i.e. simple European put and call options) exists. However, we assume that only options with limited time to maturity are liquidly traded. As well as the underlying and the money-market account, we assume the option prices (i.e. the implied volatilities of the options) to be driven by the data-generating model, and presume risk-neutrality with respect to volatility risk, i.e. the market price of volatility is set to zero in case the Heston model is used as data-generating model. Additionally, we assume the spread between bid and ask prices/volatilities to be zero.

For all considered hedging strategies we assume the hedging portfolio to consist of positions in three assets, whose quantities are rebalanced at the beginning of each hedging period: a position of quantity Δ_t^S in the underlying, a position of quantity Δ_t^B in the money-market account and a position of Δ_t^X in a 1-year ATMF straddle option (that is, an option consisting of long positions in one call and one put, both with one year to maturity and struck at-the-money with respect to the maturity's forward, hereafter referred to by the acronym "ATMF"). We assume the insurer to hold the position in the straddle for one hedging period, then sell the options at the then-current price, and set up a new position in a then 1-year ATMF straddle. For each hedging period, the price of the currently used straddle option is denoted by X_t. We assume that the portion of the hedge portfolio that is not invested in either the underlying or the straddle options is invested in (or borrowed from) the money market. Thus, the hedge portfolio at time t has the form

$$\Pi_t^{\text{Hedge}} = \Delta_t^S S_t + \Delta_t^B B_t + \Delta_t^X X_t, \tag{4.2}$$

where

$$\Delta_t^B = \frac{\Pi_t^{\text{Hedge}} - \Delta_t^S S_t - \Delta_t^X X_t}{B_t}. \tag{4.3}$$

4.2 Dynamic hedging strategies

For both considered hedging models, Black-Scholes-Merton and Heston, we analyze three different types of (dynamic) hedging strategies.

4.2.1 No active hedging

The first strategy simply invests all guarantee fees in the money-market account and does no further hedging, i.e. $\Delta_t^S = 0$ and $\Delta_t^X = 0 \quad \forall\, t$. The strategy is obviously identical for both models.

4.2.2 Delta hedging

The second type of hedging strategy uses a position in the underlying in order to hedge the portfolio against small changes in the underlying's price, i.e. to achieve so-called *delta-neutrality* of the portfolio.

Using the Black-Scholes-Merton model as hedging model, the position Δ_t^S is chosen as the delta of Ψ_t, i.e. the partial derivative of Ψ_t with respect to the underlying's spot price, because of the following reasoning: The portfolio of the insurer at time t consists of the option value Ψ_t of the pool of policies and the hedge portfolio Π_t^{Hedge}. The hedge portfolio only consists of positions in the underlying and the money-market account. Therefore we have

$$\Pi_t = -\Psi_t + \Delta_t^S S_t + \Delta_t^B B_t \ . \tag{4.4}$$

A natural question to ask is how the value of the portfolio Π_t changes from time t to time $t + dt$. The change in the portfolio value originates in changes in all three asset positions:

$$d\Pi_t = -d\Psi_t + \Delta_t^S dS_t + \Delta_t^B dB_t \ . \tag{4.5}$$

Assuming the underlying follows a geometric Brownian motion as described in section 3.1.1 with the SDE

$$dS_t = \mu S_t dt + \sigma_{BS} S_t dW_t, \quad S_0 \geq 0 \ , \tag{4.6}$$

then, from Itō we have (cf. Wilmott, 2006, [30])

$$d\Psi_t = \frac{\partial \Psi_t}{\partial t} dt + \frac{\partial \Psi_t}{\partial S_t} dS_t + \frac{1}{2} \sigma_{BS}^2 S_t^2 \frac{\partial^2 \Psi_t}{\partial S_t^2} dt \ . \tag{4.7}$$

Thus the portfolio changes by

$$d\Pi_t = -\frac{\partial \Psi_t}{\partial t} dt - \frac{\partial \Psi_t}{\partial S_t} dS_t - \frac{1}{2} \sigma_{BS}^2 S_t^2 \frac{\partial^2 \Psi_t}{\partial S_t^2} dt + \Delta_t^S dS_t + \Delta_t^B r B_t dt. \tag{4.8}$$

The random parts in (4.8) are

$$\left(-\frac{\partial \Psi_t}{\partial S_t} + \Delta_t^S \right) dS_t, \tag{4.9}$$

and therefore, if we choose

$$\Delta_t^S = \frac{\partial \Psi_t}{\partial S_t} \tag{4.10}$$

the randomness is reduced to zero.

In the Black-Scholes-Merton framework with time-continuous trading and with no transaction costs, such a position is sufficient to perform a perfect hedge. In reality however, time-discrete trading and transaction costs cause imperfections.

While delta hedging under the Black-Scholes-Merton model (given the typical assumptions), constitutes a theoretically perfect hedge, it does not under the Heston model. Delta hedging with $\Delta_t^S = \frac{\partial \Psi_t}{\partial S_t}$ could still be used, arguing that this strategy constitutes a first-order approximation (via Taylor's theorem) to the real dynamics of Ψ_t. However, as it is clear that in most cases the hedge cannot be perfect, the question arises if this imperfection can be effectively minimized.

This question leads to (locally) risk-minimizing strategies, that aim at minimizing the variance of the instantaneous change of the portfolio. Under the Heston model[2], the problem

$$\mathrm{Var}\left(d\Pi_t\right) \to \min, \qquad \Delta_t^S \in \mathbb{R}, \quad \Delta_t^X \equiv 0 \tag{4.11}$$

has the solution (cf. e.g. Ewald et al., 2007, [12])

$$\Delta_t^S = \frac{\partial \Psi_t^{\text{Heston}}}{\partial S_t} + \frac{\rho \sigma_V}{S_t} \frac{\partial \Psi_t^{\text{Heston}}}{\partial V_t}, \tag{4.12}$$

where the superscript "`Heston`" indicates that the time-t value of the pool of policies, Ψ_t, is calculated within the Heston model.

To keep notation simple, this (locally) risk-minimizing strategy under the Heston model is sometimes also referred to as "delta" hedge.

4.2.3 Delta and Vega hedging

The third type of hedging strategies incorporates the use of the straddle option X_t, exploiting its sensitivity to changes in volatility for the purpose of neutralizing (or

[2] Note that a (time-continuously) delta-hedged portfolio under the Black-Scholes-Merton model is already risk-free. Therefore for the Black-Scholes-Merton model, the delta-hedging strategy coincides with the locally risk minimizing strategy.

at least limiting) the portfolio's exposure to changes in volatility.

Under the Black-Scholes-Merton model, volatility is assumed to be constant; therefore using the model to hedge against a changing volatility may appear counterintuitive. Nevertheless, following Taleb (1997, [28]), we analyze some kind of *ad-hoc* vega hedge in our simulations, that aims at compensating the deficiencies of the Black-Scholes-Merton model: In order to perform the vega hedge, we do not compute the Black-Scholes-Merton vega $\frac{\partial \Psi_t^{\mathrm{BS}}}{\partial \sigma_{BS}}$ of the guarantee's value Ψ_t and compare it to the corresponding Black-Scholes-Merton vega $\frac{\partial X_t^{\mathrm{BS}}}{\partial \sigma_{BS}}$ of the straddle option's value X_t, but, instead, we use the so-called *modified vega* of Ψ_t for comparison.

Since the expected future cash flows of all maturities of the pool of policies cannot be expected to react the same way to changes in today's volatility, the modified vega applies a different weighting to the respective vega of each maturity. We use the inverse of the square root of time as simple weighting method and use the maturity of the hedging instrument X_t, i.e. one year, as the benchmark maturity (cf. Taleb 1997, [28]).

The *modified vega* of Ψ_t at a policy calculation date $\tilde{t}_i$ then has the form

$$\mathrm{ModVega}(\tilde{t}_i) = \sum_{k=i+1}^{M} \upsilon_{\tilde{t}_k}(\tilde{t}_i) \frac{1}{\sqrt{\tilde{t}_k - \tilde{t}_i}} \tag{4.13}$$

where the $\upsilon_{\tilde{t}_k}(\tilde{t}_i)$ denote the respective Black-Scholes-Merton vega, computed at time $\tilde{t}_i$, of the expected discounted cash flow at time $\tilde{t}_k$ of the pool of policies. The ratio between the modified vega and the vega of the straddle then is used to determine the portfolio's position in the straddles (i.e. the quantity of the straddles bought).

Under the Heston model, we compare the two derivatives of Ψ_t and X_t with respect to the current local variance V_t in the "vega" sense and then analogously determine the option position required to offset the changes of the portfolio with respect to changes in the local variance in first order, while ignoring terms of higher-order and *Cross-Greek*[3] terms.

Of course, under both hedging models, the position in the underlying must be adjusted for the delta of the option position $\Delta_t^X X_t$.

[3]Second-order and higher partial derivatives of the option price with respect to several different model parameters, for instance the sensitivity of the option delta with respect to change in volatility, the so-called "Vanna" (also known as "DvegaDspot" or "DdeltaDvol"), which is defined as the second-order derivative of the option value, once to the underlying spot price and once to the Black-Scholes volatility, in formulas: Vanna $= \frac{\partial^2 \Pi_t}{\partial S_t \partial \sigma_{BS}}$.

The hedge ratios for all three strategies used in our simulations are summarized in table 4.1 for the Black-Scholes-Merton model, and in table 4.2 for the Heston model, where the supercripts "`BS`" and "`Heston`" of the option prices indicate which model was used for computation of the option prices.

When using the Black-Scholes-Merton model for hedging, we use the assumed long-term volatility σ_{BS} for computation of the Greeks of the portfolio value Ψ_t^{BS}. For the computation of the sensitivities of the option price X_t^{BS}, however, we use the Black-Scholes-Merton implied volatility $\sigma_{BS}^X(t)$ implied by the market price at time t of the straddle option. This means for the simulation, that, at a rebalancing date of the hedge portfolio, we first retrieve the market price of the straddle option from the (unknown) data-generating model, use this price to calculate the Black-Scholes-Merton implied volatility of the option via a root finding algorithm (see section 3.4) and then put this implied volatility in the Black-Scholes-Merton model in order to calculate the sensitivities (delta and vega) of the straddle option.

	Δ^S	Δ^X
(NH)	–	–
(D-BS)	$\frac{\partial \Psi_t^{\mathrm{BS}}}{\partial S_t}$	–
(DV-BS)	$\frac{\partial \Psi_t^{\mathrm{BS}}}{\partial S_t} - \Delta_t^X \frac{\partial X_t^{\mathrm{BS}}}{\partial S_t}$	$\mathrm{ModVega}(t) \Big/ \frac{\partial X_t^{\mathrm{BS}}}{\partial \sigma_{BS}^X}$

Table 4.1: Hedge ratios for the different strategies if the Black-Scholes-Merton model is used as hedging model.

	Δ^S	Δ^X
(NH)	–	–
(D-H)	$\frac{\partial \Psi_t^{\mathrm{Heston}}}{\partial S_t} + \frac{\rho\sigma_V}{S_t}\frac{\partial \Psi_t^{\mathrm{Heston}}}{\partial V_t}$	–
(DV-H)	$\frac{\partial \Psi_t^{\mathrm{Heston}}}{\partial S_t} - \Delta_t^X \frac{\partial X_t^{\mathrm{Heston}}}{\partial S_t} + \frac{\rho\sigma_V}{S_t}\left(\frac{\partial \Psi_t^{\mathrm{Heston}}}{\partial V_t} - \Delta_t^X \frac{\partial X_t^{\mathrm{Heston}}}{\partial V_t}\right)$	$\frac{\partial \Psi_t^{\mathrm{Heston}}}{\partial V_t} \Big/ \frac{\partial X_t^{\mathrm{Heston}}}{\partial V_t}$

Table 4.2: Hedge ratios for the different strategies if the Heston model is used as hedging model.

4.2.4 Additional semi-static hedge

Additionally, for all "active" dynamic hedging strategies (Delta and Delta-Vega), we assume that the hedger buys European put options at each policy calculation date such that the possible guarantee payments for the next calculation date are fully hedged by the put options (assuming surrender and mortality rates are deterministic and known by the insurer).

This strategy aims at avoiding having to hedge an option with short time to maturity and hence having to deal with a potentially rapidly alternating delta (due to a high gamma of the option) if the option is near the strike (cf. Taleb, 1997, [28]).

This is possible for all four considered ratchet mechanisms, since the minimum guaranteed withdrawal amount at a policy calculation date is always known at the previous calculation date [4].

[4]Actually, for the ratchet mechanisms "No Ratchet", "Lookback Ratchet" and "Remaining WBB Ratchet" it holds that $E\left[W^{guar}_{\tilde{t}_{i+1}} W \middle| \mathcal{G}_{\tilde{t}_i}\right] = W^{guar}_{\tilde{t}_i}$, i.e. at a policy calculation date, the minimum guaranteed withdrawal amount of the next calculation date is the exact value of the current guaranteed withdrawal amount, due to the monotonically increasing property of these ratchet mechanisms. For the "Performance Bonus" product design, the minimum guaranteed withdrawal amount is always equal to the initial guaranteed withdrawal amount, i.e. the guaranteed withdrawal amount without any performance bonus.

Chapter 5

Software Solution

In this chapter we present the main aspects of the design and the architecture of the software solution that we used to conduct the different numerical analyses of the GLWB option presented in the forthcoming chapters 6 and 7. The two main requirements on the design and the implementation of the software solution were flexibility and reusability. That is, a single implementation of an insurance product or of a financial instrument should be sufficient for all of the desired simulations and calculations. Similarly, once implemented numerical algorithms (like the Monte Carlo routine for instance) should be working for different products, even those that will be implemented at a later point in time. Generally, in order to allow for a potentially growth and extension of the software solution for forthcoming analyses, it should be easy to add new instruments and new models of the financial market without needing to change much of the already existing code. All of this also aims at keeping redundancy at a minimum, and can therefore also be expected to be beneficial for software quality assurance.

First, we give a list of the main specifications of the software solution that we defined, discuss the issues and questions arising from the implementation and present our solution afterwards.

5.1 Functional specification

In this section, we specify the core functionality and the requested behavior of the final software solution and the requested properties of the input and output data.

- **core functionality**

  - The software solution should allow the user to conduct Monte-Carlo analyses of various financial instruments (including the GLWB products presented in chapter 2) and different models of the financial market (at least the Black-Scholes-Merton and the Heston model must be implemented).
  - The result of the Monte-Carlo simulations should be generic in the sense that while the structure of the output may be pre-specified, its content may change depending on the analyzed instrument.
  - A toolkit for the (statistical) analysis of the Monte-Carlo results should be provided to the user, including tools and functions to compute percentiles, compute simple statistics (empirical moments like mean, standard deviation, skew and kurtosis) and risk measures (value at risk, conditional tail expectation, lower moments, ...).
  - The software solution should also provide functions for the pricing of European standard options, i.e. functions should be provided that implement the closed-form formulas in case of the Black-Scholes-Merton model and the inverse Fourier transformation in the case of the Heston model.

- **input / output**
  - The required input data should be easily transferable out of spreadsheet files, i.e. file types that are used within spreadsheet calculation programs (`.xls` files for instance).
  - The result of the analyses should also be easily transferable to spreadsheet files or should be spreadsheet files in the first place.

- **user interface**
  - The user interface should be easy to use, should allow for a clear but flexible input structure and should assist the user in avoiding mistakes.
  - The user interface should also be supportive of task automatization, i.e. it should allow for a kind of scripting language (e.g. a simple batch file, Visual Basic for Applications, Perl or Python) in order to automate the use of the core functionality of the software solution via the use of macros.

- **general requirements**
  - As the numerical analyses are expected to be rather time-consuming, the software solution should aim at maximizing efficiency and speed of computation.

– The software solution should be reusable for subsequent analyses and tasks and should therefore provide a flexible structure in both, the construction of the code itself and the user interface.

– The implementation should allow for an easy way to add new financial instruments for analysis without having to change (too much) of the already existing code.

– The programming language used should be widely-used amongst academics and practitioners, and should also have a strong (online) community supporting it.

– The implementation should be easy to test and debug, i.e. the programmer should be able to locate and correct mistakes in the code with as little effort as possible.

– It must be possible for the programmer to expose functions directly to the user (in the sense that the user is able to use the function directly) that have standard signatures, i.e. functions whose input and output parameters are of standard types such as `double`, `integer` and `string`, as well as arrays and matrices of the aforementioned types.

– It must be clearly definable which functions are visible to and usable by the user and which are not. The changes in the code that are needed for exposing a function to the user should be kept minimal.

5.2 Architecture and implementation

The software solution that we build for our analyses is designed and implemented for the use under Microsoft Windows. All of the core functionality of our software solution was implemented using the C++ programming language and compiled with the Visual C++ 9.0 compiler provided by Microsoft.

We used an object oriented approach for the software solution and hence also make use of the accompanying terminology and concepts, particularly the concept of (abstract) classes as representation of an object type and as container for the features and traits of the object. We also use the concept of information hiding in this context, which says that the (implementational) details of the methods of an object as well as the data structure should be hidden from the user of this object and be encapsulated within the object. The concept also says that it is only of interest *what* is implemented by the object and not *how*, i.e. for the user is only of interest which interface (as a collection of provided functionality) is implemented by the object. This concept is called *polymorphism*, as the same interface can be implemented by several object types (classes) and therefore, if a function requires an object as argument that implements a specific interface, objects of all of these implementing classes may be passed to the function, since only the provided functionality matters and not the true type of the object.

In our software solution there are two main interfaces (or as they are known in C++: *abstract classes*, i.e. classes that have so-called *pure virtual* methods, which declare only the method's signature but do not define the method's body), which are the parent classes of all models and instruments that we implemented:

- `cProduct`: This is the main interface for all financial products and instruments. `cMCProduct` is derived from this class and represents all financial products and instruments that support the simulation within a Monte Carlo routine, and hence implement the needed functionality. Every derived class of `cProduct` and `cMCProduct` also contains the calculation dates of the product, i.e. for instance for an European call option the start and end date of the option and for a GLWB-type product all of the policy calculation dates.
- `cModel`: This is the abstract class which all of the financial market model classes are derived from. Every class that represents a model of the financial market also contains all of the corresponding model parameters and additional data needed within this model. For each model, there has to be a corresponding scenario generator, i.e. a class that produces an arbitrary number of path realizations for all of the considered assets in this market (typically at least one stock and a money-market account).

Our software solution includes a central storage facility implemented in the class `cStorage`, in which objects of the types `cProduct`, `cModel`, as well as matrices of `doubles` and models for lapsation (`cLapsation`) and mortality (`cMortality`) can be stored and retrieved from within all parts of the code. For storage and retrieval of the objects an identifier in form of an arbitrary string is used, i.e. when storing an object, apart from the object itself, a string has to be delivered by the user. The string must be unique within the category (model, product, . . .), otherwise the already stored object with the same identifier is replaced by the new object.

This storage class is designed to allow only one single instance of itself, i.e. there may never exist two parallel instances (objects) of this class. There are at least two ways of implementing such a behavior, with one being the so-called *Singleton* pattern (cf. for instance Joshi, 2004, [20]), where the user has to address all of their calls to the static method usually called `instance()` or `getInstance()` which returns a reference or a pointer to the single object of the class, which was created at the time of the first call of `instance()`. The other way of implementing this "single object pattern" makes use of `static` methods and variables, i.e. variables whose lifetime extend across the entire run of the program and methods that can be called without an existing instance of the class. We chose the latter for our implementation, mainly because of its superior simplicity.

The core design principle for the implementation of the Monte-Carlo functionality of the software solution is as follows: there is the function `MonteCarlo`, which expects information about the financial market model to be used in the simulation and about the product that is to be simulated. The information about the model is passed in form of an object of a class derived from the abstract base class called `cModel`. Every class that represents a model of the financial market used within the Monte Carlo simulation must be derived from this class. Each derived model class then is supposed to contain the model type (in form of an enumeration) and the corresponding parameters of the model. To each model, there is a corresponding scenario generator, which will generate random scenarios according to the given financial market model and for a given set of calculation dates. The base class of all scenario generators is called `cScenarioGenerator` and has the following structure:

```
class cScenarioGenerator{
public:
    virtual void initScenario(cScenario& Scenario) = 0;
    virtual void generateScenario(cScenario& Scenario) = 0;
};
```

where the class `cScenario` is a container for the financial market data (especially prices of the stocks and the money-market account at the calculation dates) in one specific scenario. The function `MonteCarlo` creates an instance of `cScenario` and passes it to the scenario generator for initialization and for each new scenario. The purpose of the function `initScenario` is that in some financial market models not all of the simulated market data is stochastic, but deterministic, i.e. the values remain the same in each scenario. In order to achieve computational savings, in this case `initScenario` writes these deterministic values in the scenario object at the beginning of the simulation and later on `generateScenario` only updates the values for the assets that are (pseudo-) randomly generated.

5.3 User interface

For the user interface we decided to follow a direct approach and integrate the solution into the spreadsheet environment provided by the Microsoft Office Excel software. The library containing the C++ implementation of our software solution is compiled as a *native Dynamic-link library* [1] and marked via the file extension `.xll` for the use within Microsoft Office Excel as an add-in. To achieve the interoperability of C++ functions and Microsoft Office Excel, we made use of version 3.0.0 of the "`xlw` C++ wrapper for Excel" library[2] which makes use of the C application programming interface (API) of Excel[3], and allows the programmer to easily export C++ functions to Excel, where these "hand-made" functions are embedded just like the built-in functions as, for instance, `=SUM(...)` are. To give an example: If a function is to be exported to Excel, which is called `MyTest`, takes a `double` as argument and returns also a `double`, then the following code must be added to the project files:

```
#include <xlw/xlw.h>
using namespace xlw;
extern "C" {
LPXLFOPER EXCEL_EXPORT xlMyTest( XlfOper xlValue ){
EXCEL_BEGIN;
    ...
    double inputValue = xlValue.AsDouble();
    double ret = Test(inputValue);
    return XlfOper(ret);
EXCEL_END;
}
XLRegistration::Arg MyTestArgs[] = {
    { "Value", "The input value", "XLF_OPER" }
};
XLRegistration::XLFunctionRegistrationHelper registerMyTest(
    "xlMyTest", "MyTest",
    "Tests the xlw C++ wrapper for Excel.",
```

[1] See for instance Microsoft's documentation of Dynamic-link libraries at `http://msdn.microsoft.com/en-us/library/ms682589(VS.85).aspx`.

[2] Project website to be found at `http://xlw.sourceforge.net`.

[3] This C API of Excel is documented in: Baarns Consulting Group. Microsoft Excel 97 Developer's kit, Microsoft Press, 1997.

```
    MyTestArgs, 1);
}
```

In order to automate this procedure of adding the `xlw`-specific code to the project, we implemented a script using the Ruby programming language[4] which parses all of the header files of the project (i.e. all files in the project directory ending on `.h`) and searches for specific annotations to function declarations and produces the export code for the matched function automatically. The annotation to the function declaration which marks the function to be exported to Excel has the following pattern:

```
//# EXPTOXLL(MyTest, Tests the xlw C++ wrapper for Excel.)
double MyTest
(
    const double Value // The input value
);
```

The first argument after `EXPTOXLL` indicates the definable name of the function within Excel (in the sense of using it in a spreadsheet cell via `=MYTEST(...)`), and the second argument, as well as the comment part after the input argument (after the double-slash), are used for the description of the fields of the Excel Function Wizard, as can be seen in the screenshot shown in figure 5.1 (note that the automatization script always adds the prefix `FR` to the function name in order to make the function names distinctable from the built-in functions).

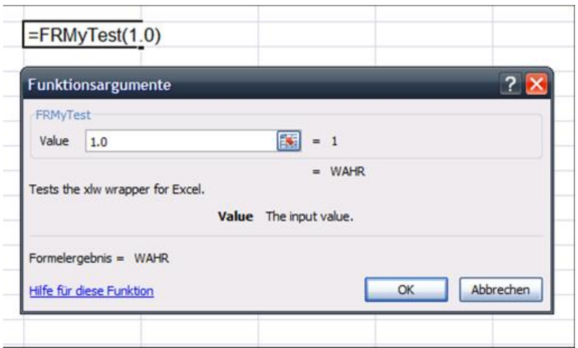


Figure 5.1: Screenshot showing the Microsoft Office Excel Function Wizard with the exemplary C++ function that was exposed to Excel via the use of xlw.

[4] See `http://www.ruby-lang.org`.

5.4 Used libraries

Our software solution makes use of several open-source libraries for C++, for the reason that we wanted to avoid writing code that has already been written by others and – more importantly – has been tested and used by many programmers. The following list gives an overview of the used libraries and their purpose (cf. Joshi, 2004, [20]):

- **Boost**: The `boost` project is an open source library designed to extend the C++ standard library (stl). It can be found at `http://www.boost.org`. The intention is that the libraries incorporated in `boost` will become part of the C++ Standard in the future, and hence, the license is very unrestrictive and allows the user basically to do whatever they want with the code (as opposed to projects using the GNU license[5]). This is important if it is intended to use the library within a product that is used or sold commercially. From the `boost` website[6] some of the requirements met by the license:

  - Must grant permission without fee to copy, use and modify the software for any use (commercial and non-commercial).
  - Must require that the license appear with all copies [including redistributions] of the software source code.
  - Must not require that the license appear with executables or other binary uses of the library.
  - Must not require that the source code be available for execution or other binary uses of the library.

  Of the many libraries of the `boost` project we use the following within our software solution (in alphabetical order, with descriptions taken from the `boost` website):

  - `Bind`: *boost::bind is a generalization of the standard functions std::bind1st and std::bind2nd. It supports arbitrary function objects, functions, function pointers, and member function pointers, and is able to bind any argument to a specific value or route input arguments into arbitrary positions.* We use this in the context of numerical quadrature, where arbitrary

[5] See the website of the Free Software Foundation, `http://www.fsf.org/licensing/`, for details.

[6] `http://www.boost.org/users/license.html`, retrieved August 20, 2009.

unary functions (i.e. functions that take a double as argument and returning a double) can be passed to the quadrature function in order to have them numerically integrated.

  - `Conversion`: *Polymorphic and lexical casts.* Used in our software solution for lexical casts from strings to doubles.
  - `Date Time`: *A set of date-time libraries based on generic programming concepts.* We use this for the calculation dates of the analyzed financial instruments.
  - `Math/Statistical Distributions`: *A wide selection of univariate statistical distributions and functions that operate on them.*
  - `Random`: *A complete system for random number generation.* We use this for (pseudo-) random number generation, especially the implementation of the Mersenne Twister 19937 algorithm[7].
  - `Smart Ptr`: *Smart pointer class templates.* We use smart pointers to manage all of the `cModel` and `cProduct` objects, i.e. when an object of this type is to be passed to a function, a smart pointer referencing to the object is passed instead. This way we avoid potential memory leakage when working with many dynamically generated objects. This is especially important because leaked memory would accumulate with every function call from Excel and would not be freed again until the Excel application is closed.
  - `uBLAS`: *uBLAS provides matrix and vector classes as well as basic linear algebra routines. Several dense, packed and sparse storage schemes are supported.* We use this for all numerical calculations and in high-performance areas of the code, where fast parsing of arrays is needed. In those parts of the code where performance is not an issue, we often use the `std::vector` template provided by the standard library instead.

- **QuantLib**: The QuantLib project is aimed at "providing a comprehensive software framework for quantitative finance. QuantLib is a free / open-source library for modeling, trading, and risk management in real-life." The project website can be found at `http://www.quantlib.org`. Similarly to `boost`, the license is very unrestrictive, allowing free use in commercial software. We took parts of this library and used it for several numerical algorithms in our software

[7]cf. Matsumoto, M., Nishimura, T. (1998), "Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator".

solution, including the Gauss-Lobatto quadrature, which is used in the context of the Heston model for the pricing of European "plain vanilla" options via Fourier inversion (cf. section 3.3.3).

- **xlw**: We already mentioned the use of this library for generating Excel plug-ins, known as `xlls`. The code for `xlw` can be obtained from `http://xlw.sourceforge.net`.

5.5 Usage example

In this section, we give a short example to illustrate the usage of our software solution via the Microsoft Office Excel spreadsheet application. In this example we will perform a 10-year Monte Carlo simulation of a portfolio of identical GLWB contracts under the Heston stochastic volatility model and we will compute simple statistics of the simulation result.

(i) =FRSchedule(Start; Step; NbSteps)
In order to generate a financial instrument, we need calculation dates, which are often of the form "each month for x years". For this purpose =FRSchedule provides the functionality to define a start date, a step size (e.g. `1w` for a step size of one week, or `2m` for the step size of two months), and the total number of steps. The function will return a vector of dates according to the specification. In our example we have annual calculation dates for a period of 10 years, therefore the function call is =FRSchedule(TODAY(); ‘‘1y’’; 10).

(ii) =FRSetPGMxBProduct(ID; LumpSum; GWRate; DeathBenefitMethod; StepUpMethod;[...]; AcquisitionFee ;[...]; CalcDates)
This function generates an object of the type `cGMxBProduct`, the first argument, `ID`, will be the string identifier of the object that is needed to pass it to other functions (e.g. `LookbackRatchet#01` would be a possible choice). The last argument is a vector of the calculation dates, i.e. here we would pass the dates that we created in the first step. The rest of the arguments are the parameters of the product (note that we omitted some of the parameters for the sake of simplicity of presentation). All of the functions that begin with `FRSet...` will first check the parameters for correctness, will then create an object of their specific type, will store the created object via the `cStorage` class, and will finally return the identifier of the object back to the cell in Excel from where the function was called.

(iii) =FRSetMortality(ID; Ages; DeathProbabilities)
For the portfolio of GLWB contracts, we need an underlying model of the mortality of the insured. In this case, we only use a simple mortality table, matching an annualized probability to each age. The first argument is the identifier of the created mortality table, e.g. `Mort#male#1965` would be a possible choice, whereas `Ages` and `DeathProbabilities` are vectors representing the mortality probability for each age of the insured.

(iv) =FRSetLapsation(ID; ElapsedTime; LapsationProbabilities)
For the portfolio of GLWB contracts, we also need an underlying model of the lapsation of the insured. Again, we use a simple model where in each contract year there is a certain annualized probability that the insured will surrender. The first argument is the identifier of the created surrender probabilities, for instance `Lapse#01`.

(v) =FRSetPGMxBPortfolio(ID; GMxBProductID; MortalityID; LapsationID; Age)
In order to generate the product portfolio we have to pass to the function =FRSetPGMxBPortfolio its chosen identifier, the identifier of the underlying product, the identifier of the mortality object, the identifier of the lapsation object, and the age of the insured at outset of the contract, i.e. in our example the call for a 65-year old would look like this:
=FRSetPGMxBPortfolio(‘‘Portfolio#01’’; ‘‘LookbackRatchet#01’’;
‘‘ Mort#male#1965’’; ‘‘Lapse#01’’; 65).

(vi) =FRSetMHeston(ID; Spot; ShortVol; LongVol; MeanReversion; VolOfVol; Correlation; UndYC; CurrYC)
Now we have to create an object for the model of the financial market. The function =FRSetMHeston will create an object of the model class `cHestonModel`, which is derived of the abstract class `cModel`. `Spot` is the spot price of the stock at inception, the following parameters are model-specific (cf. section 3.1), `UndYC` is the yield curve of the drift of the stock, and `CurrYC` is the yield curve of the currency in which the money-market account is denoted.

(vii) =FRMonteCarlo(ProductID; ModelID; NbPaths; Seed)
To perform the actual Monte-Carlo simulation, we have to pass the identifier of the product that is to be analyzed and the identifier of the model that the model that is to be used for scenario generation. Additionally, the number of paths and the seed for the random number generation is needed as argument. The function will then try to fetch the model and the product from `cStorage`,

and will then perform the Monte-Carlo simulation. All single path results are then stored in `cStorage`, and the mean of the results is returned as a cell matrix to Excel.

(viii) =FRStatistics(Statistic ; Percentile ; Dimension; RowNbr)
This function is used to calculate statistics from a simulation result which is stored in `cStorage`. The following statistics are supported: minimum, maximum, mean, standard deviation, variance, skewness, kurtosis, value at risk, expected shortfall, and arbitrary percentiles between 0 and 1. The arguments `Dimension` and `RowNbr` are used to select the value of the result matrix from which the statistic is to be computed.

Chapter 6

Contract Analysis

In this chapter, we analyze the GLWB rider options that we presented and defined in chapter 2. We first compare the pricing results for each product design under different model assumptions for the financial market and the policyholder behavior. Afterwards we analyze the distribution of the guaranteed withdrawal amount generated by the different product designs, and also analyze and compare the different distributions of the point in time when the guarantee of the GLWB option triggers. We conclude the chapter with an analysis of the deltas (i.e. the sensitivity of the option's value to changes in the underlying's spot price) of the considered GLWB product designs.

6.1 Determination of the fair guaranteed withdrawal rate

In this section, we first calculate the guaranteed withdrawal rate x_{WL} that makes a contract fair (in the sense discussed in section 2.3), assumed all other policy parameters are given. In order to calculate x_{WL}, we perform a root search with x_{WL} as argument and the value of the GLWB option V_0^G (as defined in section 2.3) as function value. For all of our analyses we assume a fee structure of the GLWB contract as given in Table 6.1.

Policy calculation dates are assumed to be annually, i.e. withdrawals and step-ups are made only once per year at a prespecified point in time. The number of policyholders in the pool is assumed to be large enough to allow for the use of the Law of Large Numbers with respect to mortality and lapsation within the pool, and

Acquisition fee	4.00% of lump sum
Management fees	1.50% p.a. of NAV
Guarantee fees	1.50% p.a. of NAV
Withdrawal fees	0.00% of withdrawal amount

Table 6.1: Assumed fee structure for all considered contracts.

thus deterministic mortality and surrender rates may be applied for projection of the cohort of policyholders. We further assume the policyholder to be a 65 years old male. For pricing purposes, we use best-estimate mortality probabilities given in the DAV 2004R table published by the German Actuarial Society (DAV).

Regarding the policyholder behavior, we assume that policyholders - as long as their contracts are still in force - only chose between two options at a policy calculation date: either to withdraw exactly the guaranteed annual withdrawal amount or to perform a full surrender and withdraw all of the remaining assets[1].

All results in this chapter have been calculated assuming an annual risk-free rate of interest of $r = 4\%$.

6.1.1 Results for the Black-Scholes-Merton model

Table 6.3 displays the results for the fair guaranteed withdrawal rate x_{WL} for all four ratchet mechanisms that were defined in chapter 2 and for different assumptions regarding the policyholder behavior, where the pricing was done using the Black-Scholes-Merton model with different assumed values σ_{BS} for the equity volatility. In regard to the policyholder behavior, we consider three different scenarios: no policyholder ever surrenders (no surr), surrender happens according to Table 6.2 (surr 1) and surrender with twice the probabilities given in Table 6.2 (surr 2).

A comparison of the different product designs shows that, obviously, the highest annual guarantee can be provided if no ratchet or performance bonus is provided at all. If no surrender and a volatility of $\sigma_{BS} = 20\%$ is assumed, the guarantee is similar to a "5 for life" product (4.98%). Including a Lookback Ratchet would need a reduction of the initial annual guarantee by 66 basis points to 4.32%. If a richer ratchet mechanism is provided such as the Remaining WBB Ratchet, the guarantee

[1]Please note that if the guaranteed withdrawal amount coincides with the remaining fund assets, the withdrawal of all remaining assets is not considered to be a full surrender and the guarantee does not end.

Year	Surrender rate
1	6 %
2	5 %
3	4 %
4	3 %
5	2 %
≥6	1 %

Table 6.2: Assumed deterministic surrender rates.

needs to be reduced to 4.01%. About the same annual guarantee (4.00%) can be provided if no ratchet is provided but a Performance Bonus is paid out annually.

Throughout our analyses, the Remaining WBB Ratchet and the Performance Bonus allow for a similar annual guarantee. However, for lower volatilities, the Remaining WBB Ratchet seems to be less valuable than the Performance Bonus and therefore allows for a higher withdrawal rate while for higher volatilities the Performance Bonus allows for higher withdrawal rates. Thus, the relative impact of volatility on the price of a GLWB depends on the chosen product design and appears to be particularly high for ratchet type products (II and III). This can also be observed when comparing the No Ratchet case with the Lookback Ratchet. While - when volatility is increased from 15% to 25% - for the No Ratchet case, the fair withdrawal rate decreases by just over half a percentage point from 5.26% to 4.70%, it decreases by almost a full percentage point from 4.80% to 3.85% in the Lookback Ratchet case (all values for the no surrender case). The reason for this is that for the products with ratchet, high volatility leads to a possible lock in of high positive returns of the underlying fund in some years (as some kind of option on an option) and thus is a rather valuable feature if volatilities are high.

If the insurance company assumes the deterministic surrender probabilities given in table 6.2 when pricing the GLWB option, the fair withdrawal rates increase for all considered model points. The extent of the increase of the withdrawal rate is rather similar over all product types and assumed values of the Black-Scholes-Merton volatility σ_{BS}. The annual guarantee increases by around 15-20 basis points if the surrender assumption from table 6.2 is made and increases by another 20 basis points if this surrender assumption is doubled.

		Product			
		No Ratchet	Lookback Ratchet	Rem. WBB Ratchet	Performance Bonus
Volatility	Surrender	I	II	III	IV
$\sigma_{BS} = 15\%$	No surr	5.26	4.80	4.43	4.37
	Surr 1	5.45	5.00	4.62	4.57
	Surr 2	5.66	5.22	4.83	4.79
$\sigma_{BS} = 20\%$	No surr	4.98	4.32	4.01	4.00
	Surr 1	5.16	4.50	4.18	4.19
	Surr 2	5.35	4.71	4.38	4.40
$\sigma_{BS} = 22\%$	No surr	4.87	4.13	3.85	3.85
	Surr 1	5.04	4.30	4.01	4.03
	Surr 2	5.23	4.50	4.20	4.24
$\sigma_{BS} = 25\%$	No surr	4.70	3.85	3.61	3.62
	Surr 1	4.86	4.01	3.76	3.81
	Surr 2	5.04	4.20	3.94	4.01

Table 6.3: Fair guaranteed withdrawal rates x_{WL} in % for different ratchet mechanisms, different policyholder behavior assumptions and for different values of the Black-Scholes-Merton volatility σ_{BS}.

6.1.2 Results for the Heston model

For the pricing of the GLWB rider option under the Heston model we use the calibration given in table 6.4, where the Heston parameters (long-term local variance θ, speed of mean reversion κ, "vol of vol" σ_V, equity correlation ρ, and starting value for the local variance process V_0) are those derived by Eraker (2004, [11]), which have also been stated in annualized form for instance by Ewald et al. (2007, [12]).

Parameter	Value
r	0.04
θ	$(0.220)^2$
κ	4.75
σ_V	0.55
ρ	-0.569
V_0	θ

Table 6.4: Benchmark parameters for the Heston model.

One of the key parameters in the Heston model is the market price of volatility risk λ (cf. section 3.2). Since absolute λ-values are hard to interpret, table 6.5 shows the corresponding values for the long-term local variance θ and the speed of mean reversion κ for different choices of λ.

Market price of volatility risk	Speed of mean reversion κ	Long-run local variance θ
$\lambda = 3$	6.40	$(0.190)^2$
$\lambda = 2$	5.85	$(0.198)^2$
$\lambda = 1$	5.30	$(0.208)^2$
$\lambda = 0$	4.75	$(0.220)^2$
$\lambda = -1$	4.20	$(0.234)^2$
$\lambda = -2$	3.65	$(0.251)^2$
$\lambda = -3$	3.10	$(0.272)^2$

Table 6.5: $\mathbb{Q}$-parameters for different choices of the market price of volatility risk factor λ.

Higher values of λ correspond to a lower long-term variance and a higher speed of mean reversion, while negative values of λ correspond to a higher long-term variance and a lower speed of mean reversion. Thus, a negative value of λ leads to a higher variation and a higher long-term mean of the local variance process, meaning that if the insurer is "short volatility" (i.e. the insurer potentially suffers from volatility increases) as it is the case if the insurer has a portfolio of sold GLWB rider options, then presumably a negative value of λ would be chosen for pricing purposes. $\lambda = 2$ implies a long-term volatility of $\theta = (19.8\%)^2$ and $\lambda = -2$ implies a long-term variance of $\theta = (25.1\%)^2$.

In table 6.6, we show the pricing results for the fair annual withdrawal rates under the Heston model for all different product designs, the same scenarios for the policyholder behavior as in the Black-Scholes-Merton case, and integer values of λ between -2 and 2.

Under the Heston model, the fair annual guaranteed withdrawal appears to be the same as under the Black-Scholes model with a comparable constant volatility. For instance for $\lambda = 0$, which corresponds to a long-term volatility of 22%, the fair annual guaranteed withdrawal rate for a contract without ratchet is given by 4.87%, exactly the same number as under the Black-Scholes-Merton model. In the Lookback Ratchet case, the Heston model leads to a fair guaranteed withdrawal rate of 4.17%, the Black-Scholes-Merton model of 4.13%. For the other two product designs, again, both asset models almost exactly lead to the same withdrawal rates. Also the impact

		Product			
Market price of volatility risk factor	Surrender	No Ratchet I	Lookback Ratchet II	Rem. WBB Ratchet III	Performance Bonus IV
$\lambda = \;\;2$	No surr	4.99	4.36	4.03	4.00
	Surr 1	5.18	4.56	4.21	4.22
	Surr 2	5.38	4.76	4.40	4.43
$\lambda = \;\;1$	No surr	4.93	4.27	3.95	3.93
	Surr 1	5.12	4.46	4.13	4.14
	Surr 2	5.31	4.66	4.32	4.35
$\lambda = \;\;0$	No surr	4.87	4.17	3.86	3.84
	Surr 1	5.05	4.35	4.03	4.06
	Surr 2	5.24	4.55	4.22	4.27
$\lambda = -1$	No surr	4.79	4.05	3.75	3.74
	Surr 1	4.97	4.23	3.92	3.95
	Surr 2	5.16	4.42	4.10	4.16
$\lambda = -2$	No surr	4.70	3.90	3.62	3.62
	Surr 1	4.87	4.08	3.79	3.82
	Surr 2	5.05	4.26	3.97	4.04

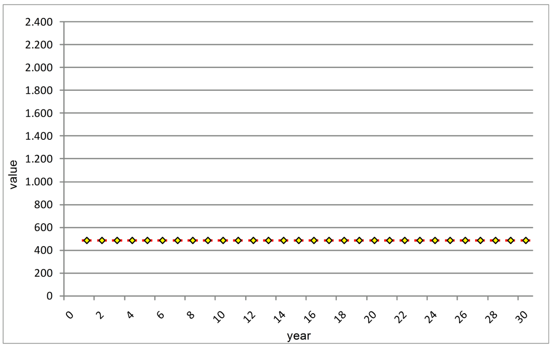
Table 6.6: Fair guaranteed withdrawal rates x_{WL} in % under the Heston model for different ratchet mechanisms, different assumptions regarding the policyholder behavior and for different values of the market price of volatility risk parameter λ.

that the different assumed policyholder behavior scenarios have on the pricing outcome seem to be of the same magnitude as in the Black-Scholes-Merton case, with an increase of around 20 basis points between the scenarios.

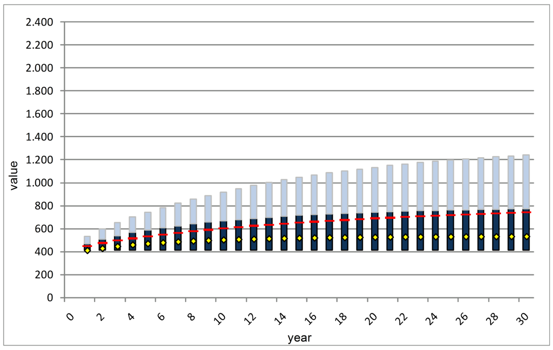
Thus, for the pricing (as opposed to hedging, see chapter 7) of GLWB rider options, the long-term volatility assumption is much more crucial than the question whether volatility should be modeled stochastic or deterministic.

6.2 Distribution of withdrawals

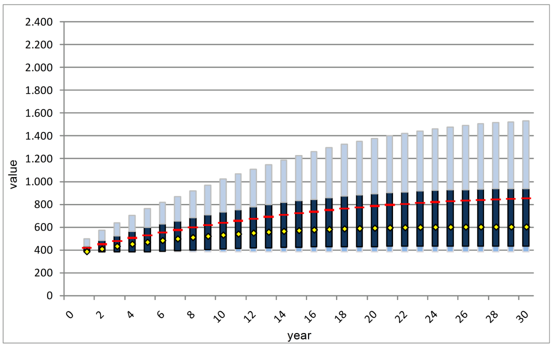
In this section, we compare the distributions of the guaranteed withdrawal benefits (given the policyholder is still alive and did not surrender the contract) for each policy year and for all four different ratchet mechanisms that were presented in chapter 2. We use the Black-Scholes-Merton model for all simulations in this chapter and assume a risk-free rate of interest $r = 4\%$, a drift of the underlying of $\mu = 7\%$ and



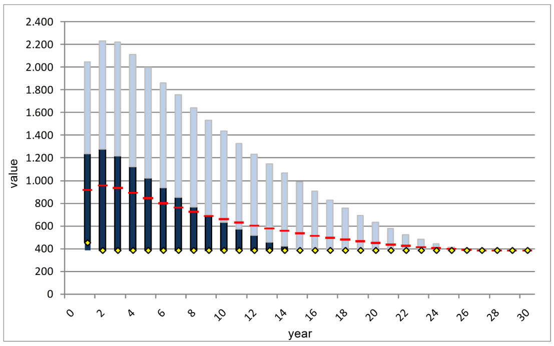
(i) No Ratchet



(ii) Lookback Ratchet



(iii) Remaining WBB Ratchet



(iv) Performance Bonus

Figure 6.1: Development of empirical percentiles (25^{th} - 75^{th} dark blue area, 10^{th} - 90^{th} light blue area), median (yellow points) and mean (red line) of the guaranteed withdrawal amount over policy years 0 to 30 for each ratchet type and a single premium of 10,000. The Black-Scholes-Merton model with parameters $\mu = 7\%$ and $\sigma_{BS} = 22\%$ was used for data generation of the simulation.

a constant volatility of $\sigma_{BS} = 22\%$.
For all four ratchet types, we use the guaranteed withdrawal rates derived in section 6.1 (without surrender). In figure 6.1, for each product design we show the development of arithmetic average, median, 10^{th} - 90^{th} percentiles, and 25^{th} - 75^{th} percentiles of the guaranteed annual withdrawal amount over time, assuming the policyholder paid a single premium of 10,000.

Obviously, the different considered product designs lead to significantly different risk/return profiles for the policyholder. While the No Ratchet case provides deterministic cash flows over time, the other product designs differ quite considerably. Both ratchet products have potentially increasing benefits. For the Lookback Ratchet, however, the 25th percentile remains constant at the level of the first withdrawal amount. Thus, the probability that a ratchet never happens is higher than

25%. The median increases for the first 10 years and then reaches some constant level implying that with a probability of more than 50% no withdrawal increments will take place thereafter.

Product III (Remaining WBB Ratchet) provides more potential for increasing withdrawals: For this product, the 25^{th} percentile increases over the first few years and the median is increasing for around 20 years. In the 90^{th} percentile, the guaranteed annual withdrawal amount reaches $1,500$ after slightly more than 25 years while the Lookback Ratchet hardly reaches $1,200$. On average, the annual guaranteed withdrawal amount more than doubles over time while the Lookback Ratchet does not, of course this is only possible since the guaranteed withdrawal at $t = 0$ is lower.

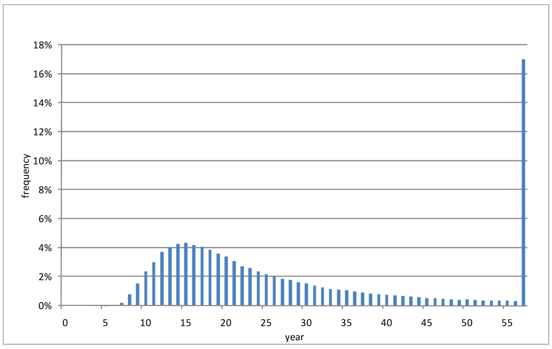
A completely different profile is achieved by the fourth product design, the product with Performance Bonus. Here, annual withdrawal amounts are rather high in the first years and are falling later. After 15 years, with a 75% probability no more performance bonus is paid, after 25 years, with a probability of 90% no more performance bonus is paid.

For all three product designs with some kind of bonus, the probability distribution of the annual withdrawal amount is rather skewed: the arithmetic average is significantly above the median. For the product with Performance Bonus, the median exceeds the guarantee only in the first year. Thus, the probability of receiving a performance bonus in later years is less than 50%. The expected value, however, is more than twice as high.

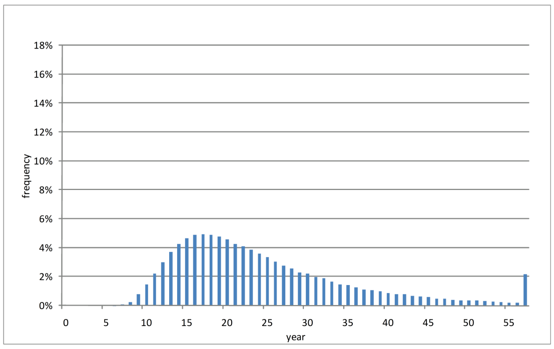
6.3 Distribution of the guarantee's trigger time

In this section, we analyze the point in time τ_G, when the guarantee of the contract triggers, i.e. we analyze the point in time when the insurer has to compensate for the guarantee payments to the policyholder for the first time since policy inception. For this analysis, we assume that the policyholder does not die and does not surrender during the observation period, i.e. the guaranteed withdrawal amount is deducted every year.

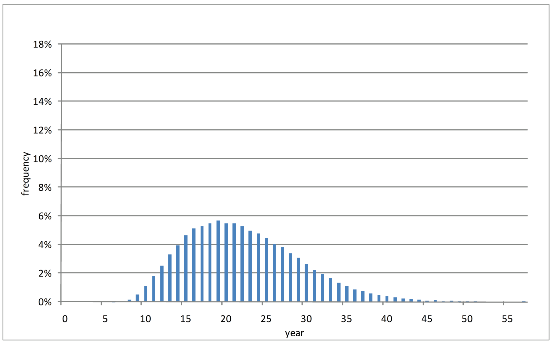
In figure 6.2, for each of the products, we show the simulation results for the probability distribution of the trigger time τ_G. Any probability mass at the end of the simulation period refers to scenarios where the guarantee is not triggered. The assumptions and parameters used for this simulation are the same as in section 6.2.



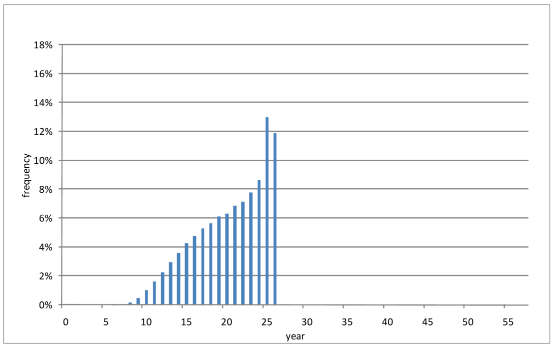
(i) No Ratchet



(ii) Lookback Ratchet



(iii) Remaining WBB Ratchet



(iv) Performance Bonus

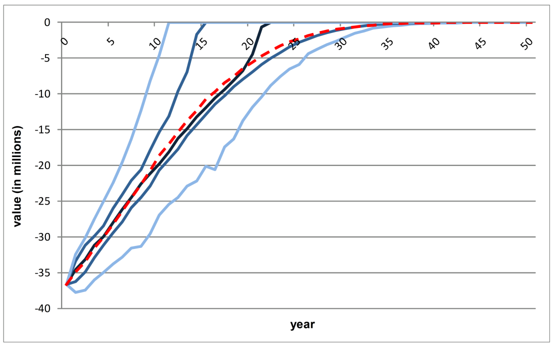
Figure 6.2: Distribution of guarantee's trigger time for each of the product designs. The Black-Scholes model with parameters $\mu = 7\%$ and $\sigma_{BS} = 22\%$ was used for the data generation of the simulation.

For the No Ratchet product, trigger times vary from 7 to over 55 years. With a probability of 17%, there is still some account value available at age 121 (the limiting age of the mortality table). For this product, on the one hand, the insurance company's uncertainty with respect to if and when guarantee payments have to be paid is very high; on the other hand, there is a significant chance that the guarantee is not triggered at all, meaning that the insurer never has to compensate for the guaranteed withdrawals of the insured, even if the insured survives until the limiting age of the mortality table.

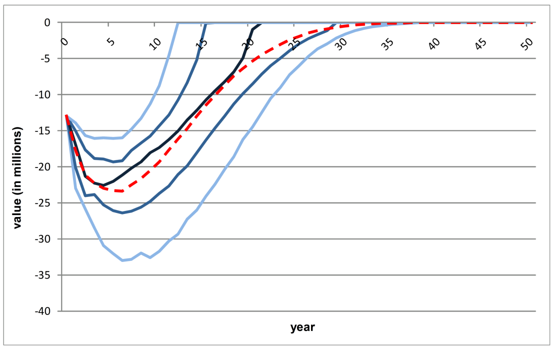
For the products with ratchet features, very late or even no triggers appear to be less likely. The more valuable a ratchet mechanism is for the client, the earlier the guarantee tends to trigger. While for the Lookback Ratchet still 2% of the contracts do not trigger at all, the Remaining WBB Ratchet almost certainly triggers within the first 40 years. However, on average the guarantee is triggered rather late, after around 20 years.

The least uncertainty in the trigger time appears to be in the product with Performance Bonus. While the probability distribution looks very similar to that of the Remaining WBB Ratchet for the first 15 years, trigger probabilities then increase rapidly and reach a maximum at $t = 25$ and 26 years. Later trigger times did not occur at all within our simulation. The reason for this is quite obvious: The Performance Bonus is calculated as 50% of the difference between the current account value and the remaining withdrawal benefit base. However, the remaining withdrawal benefit base is annually reduced by the initially guaranteed withdrawal amount and therefore reaches 0 after 26 years $(1/3.85\%)$. Thus, after 20 years, almost half of the account value is paid out as bonus every year.

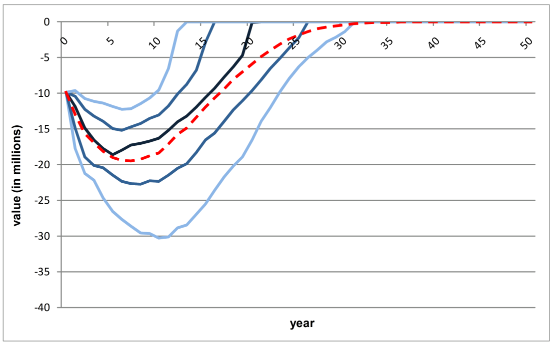
This, of course, leads to a tremendously decreasing account value in later years. Therefore, there is not much uncertainty with respect to the trigger time on the insurance company's side. On the other hand, the complete longevity tail risk remains with the insurer. Whenever the guarantee is triggered, the insurance company needs to pay an annual lifelong annuity equal to the last guaranteed annual withdrawal amount. This is the guarantee that needs to be hedged by the insurer. Thus, in the following section, we have a closer look on the Greeks of the guarantees of the different product designs.



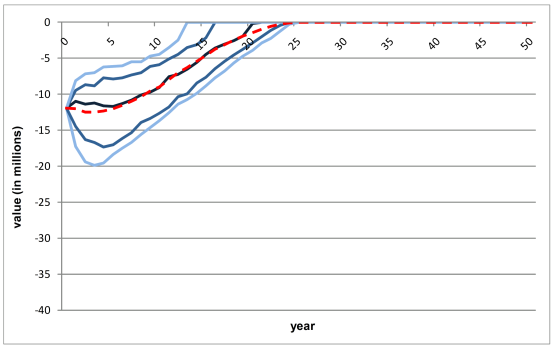
(i) No Ratchet



(ii) Lookback Ratchet



(iii) Remaining WBB Ratchet



(iv) Performance Bonus

Figure 6.3: Development over time of the empirical percentiles (dark blue line is the median, blue lines are 25$^{\text{th}}$ - 75$^{\text{th}}$, light blue lines 10$^{\text{th}}$ - 90$^{\text{th}}$ percentiles) and mean (red dashed line) of the delta of a pool of policies, at each time multiplied with the then current spot S_t of the underlying. The delta was computed within the Black-Scholes model with parameters with $r = 4\%$, $\mu = 7\%$ and $\sigma_{BS} = 22\%$, which was also used for data generation of the simulation.

6.4 The Delta of the GLWB rider option

Within our Monte-Carlo simulation, for each scenario we can calculate different sensitivities of the option value as defined in section 2.3, the so called Greeks. All Greeks are calculated for a pool of identical policies with a total single premium volume of US$ 100m under assumptions of future mortality and future surrender. All the results shown in this section are calculated under standard mortality and no surrender assumptions.

In figure 6.3, we chose to show different empirical percentiles as well as median and arithmetic average of the so-called delta, i.e. the sensitivity of the option value as defined in section 2.3 with respect to changes in the price of the underlying.

First of all, it is rather clear that all products throughout do have negative deltas since the value of the guarantee increases with falling stock markets and vice versa. Once the guarantee is triggered, no more account value is available and thus, from this point on, the delta is zero. Thus, in what follows, we call delta to be “high” whenever its absolute amount is large.

At outset, the product without any ratchet or bonus does have the biggest delta and thus the highest sensitivity with respect to changes in the underlying’s price. The reason for this mainly is that the guarantee is not adjusted when fund prices rise. In this case, the value of the guarantee decreases much stronger than with any product where either ratchet lead to an increasing guarantee or a performance bonus leads to a reduction of the account value. On the other hand, if fund prices decrease, the first product is deeper in-the-money since it does have the highest guarantee at outset. Over time, all percentiles of the delta in the No Ratchet case are decreasing, and the decay of the delta[2] is quite steady in comparison to the other product designs.

For products II and III, the guarantee can never be far out-of-the-money due to the ratchet feature. Thus delta increases in the first few years. All percentiles reach a maximum after ten years and tend to be decreasing from then on, i.e. there is a change of sign from plus to minus for the delta decay.

For the product with Performance Bonus, delta exposure is by far the lowest. This is consistent with our results of the previous section where we concluded that the uncertainty for the insurance company is the highest in the No Ratchet case and the lowest in the Performance Bonus case.

[2]Delta decay is also known as the “Charm” or “DdeltaDtime” of an option, and is mathematically defined as the second derivative of the option’s value, once to the spot price of the underlying asset and once to time, that is $\frac{\partial^2 V_t^G}{\partial S_t \partial t}$.

Chapter 7

Analysis of Hedge Efficiency

In this chapter we use the hedging strategies presented in chapter 4 to apply and analyze them within Monte Carlo simulations, where we use different models for data-generating, that is, we use different models for scenario generation of the underlying asset's spot price.

7.1 Objective and risk measures

We use the hedging strategies in order to hedge a pool of identical GLWB policies, i.e. policies that have the same product design, the same age at inception of the insured and the same product parameters like for instance the withdrawal rate x_{WL} and the fee structure. However, the insureds themselves are assumed to be separate individuals with independent decisions to lapse the contract and with independent time of deaths.
We assume the number of policyholders in the pool to be large enough to allow for the use of the Law of Large Numbers with respect to mortality and lapsation within the pool, and thus deterministic mortality and surrender rates may be applied for projection of the cohort of policyholders.
In each scenario, we start with a pool of identical contracts that is then projected until the limiting age of the mortality table is reached. We use the same denotations as in chapter 4, that is the time- t value of the guarantees embedded in the pool of policies is denoted by Ψ_t, whereas the time- t value of the hedge portfolio implemented by the insurer is denoted by Π_t^{Hedge}, and the insurer's portfolio balance Π_t at time t for the pool of policies is given by the difference between the value of the embedded

guarantees and the hedge portfolio, i.e. by

$$\Pi_t = -\Psi_t + \Pi_t^{\text{Hedge}} , \tag{7.1}$$

where all of the cash-flows generated by the pool of policies are transacted through the hedge portfolio Π^{Hedge} of the insurer, i.e. there are no transactions by the insurance company to or from the portfolio Π_t up until the end of the simulation (at time T), where the insurer's final profit/loss (short: P/L), Π_T, has to be settled.

We use the following three ratios to compare the performance of the different hedging strategies, all in normalized form as a percentage of the premiums paid to the insurer at $t = 0$:

- $E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$, the discounted expectation of the final value of the insurer's profit/loss under the real-world measure $\mathbb{P}$, where $T = \omega - x_0$ is the last policy calculation date. This is a measure for the insurer's expected profit, where a value of 1 means that, in expectation, for a single premium of 100 paid by the client, the insurance company's expected profit is 1.

- $\text{CTE}_{1-\alpha}(\chi) = E_{\mathbb{P}}\left[-\chi|-\chi \geq \text{VaR}_\alpha(\chi)\right]$, the conditional tail expectation of the random variable χ, where χ is defined as the minimum of the discounted values of the insurer's profit/loss at all policy calculation dates, i.e. $\chi = \min\left\{e^{-r\tilde{t}_i}\Pi_{\tilde{t}_i}\middle|i = 0, \ldots, M\right\}$, and VaR denotes the Value at Risk operator, that is $\text{VaR}_\alpha(\chi) = \inf\left\{l \in \mathbb{R}|\mathbb{P}(-\chi > l) \leq \alpha\right\}$. This is a measure for the insurer's downside risk given a certain hedging strategy: It can be interpreted as the additional amount of money that would be necessary at outset such that the insurer's profit/loss would never become negative over the life of the contract, even if the financial market develops according to the average of the α (e.g. 10%) worst scenarios in the stochastic model. In this sense, a value of 1 means that, for a single premium of 100 paid by the client, the insurance company would need to hold 1 additional unit of capital upfront.

- $\text{CTE}_{1-\alpha}(e^{-rT}\Pi_T) = E_{\mathbb{P}}\left[-e^{-rT}\Pi_T\middle|-e^{-rT}\Pi_T \geq \text{VaR}_\alpha(e^{-rT}\Pi_T)\right]$, the conditional tail expectation of the discounted final profit/loss Π_T. This again is a risk measure, however, with the difference to the previously defined $\text{CTE}_{0.9}(\chi)$, that it focuses on the value of the profit/loss at maturity T, i.e. after all liabilities have been met (as in the model all of the insured have deceased), and does therefore not factor in negative portfolio values that may have occurred over time. Thus a value of 1 in means that, in expectation over the

α (e.g. 10%) worst scenarios, for a premium of 100 paid by the client, the insurance company's expected discounted loss is 1. By definition, it holds $\mathrm{CTE}_{1-\alpha}(\chi) \geq \mathrm{CTE}_{1-\alpha}(e^{-rT}\Pi_T)$.

7.2 Simulation results

For all of the considered hedging strategies we assume that the hedge portfolio is rebalanced on a monthly basis.

In the numerical analyses below, we set $\alpha = 10\%$ for both risk measures and assume a pool of identical policies with parameters as given in chapter 6, assuming that none of the policyholders surrenders.
Our analysis focuses on model risk rather than parameter risk. Therefore, we use the benchmark parameters for the capital market models presented in chapter 6 for both, the hedging and the data-generating model, and we assume no risk regarding mortality, thus using the same mortality tables[1] for projection of the insured cohort and for hedging.

7.2.1 Comparison of hedge results

Table 7.1 gives the results for different hedging strategies and different data-generating models as a percentage of the single premium paid by the client. In the table, the hedging strategies are abbreviated as follows: (NH) represents the "no active hedging" strategy presented in chapter 4, (D-BSM) represents the delta-hedging strategy where the delta is computed using the Black-Scholes-Merton model. (D-H) stands for the local risk-minimizing strategy performed using the Heston model, whereas (DV-BSM) indicates the delta-vega combination hedge using the additional straddle options computed under the Black-Scholes-Merton model. (DV-H) finally represents the extension of the local risk-minimizing strategy (D-H) for vega hedging also using straddle options.

If no hedging (strategy (NH)) is in place, the insurance company's portfolio has a positive delta due to the put character (and therefore negative delta, cf. section 6.4) of the sold GLWB guarantees. This effectively means the insurance company has a (indirect) long position in the underlying and thus faces a rather high expected

[1] Best-estimate mortality probabilities given in the DAV 2004R table published by the German Actuarial Society (DAV).

		Data-generating model							
		Black-Scholes-Merton Product				**Heston** Product			
Strategy	Statistic	I	II	III	IV	I	II	III	IV
(NH)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$	10.43	07.77	06.67	3.88	10.36	7.97	6.82	04.13
	$\text{CTE}_{0.9}(\chi)$	25.29	20.07	17.54	15.12	25.76	20.97	18.54	15.97
	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$	23.41	18.27	15.90	13.35	22.93	18.55	16.25	13.51
(D-BSM)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$	0.48	0.27	0.21	0.17	0.57	0.29	0.17	0.13
	$\text{CTE}_{0.9}(\chi)$	1.71	3.25	3.12	2.02	2.77	4.76	4.51	3.35
	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$	1.44	2.74	2.71	1.78	2.44	4.14	3.99	3.02
(D-H)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$					0.52	0.42	0.34	0.21
	$\text{CTE}_{0.9}(\chi)$					2.63	4.59	4.44	3.36
	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$					2.28	4.03	3.98	2.95
(DV-BSM)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$					0.82	0.81	0.75	0.47
	$\text{CTE}_{0.9}(\chi)$					1.75	2.41	3.01	1.88
	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$					1.35	1.80	2.40	1.53
(DV-H)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$					0.49	0.41	0.33	0.19
	$\text{CTE}_{0.9}(\chi)$					1.40	1.99	1.95	1.49
	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$					1.15	1.58	1.60	1.21

Table 7.1: Results for different hedging strategies and different data-generating models as a percentage of the single premium paid by the client. The strategies are "no active hedging" (NH), delta-hedging with the Black-Scholes-Merton model (D-BSM), local risk-minimizing strategy using the Heston model (D-H), delta-vega hedge using the Black-Scholes-Merton model (DV-BSM), and the extension for vega hedging of the local risk-minimizing strategy under Heston (DV-H).

return combined with high risk. The results show that no active hedging effectively means that the insurance company, on average over the worst 10% scenarios, would need additional capital between 15% and 25%, depending on the respective product design, of the total premium volume paid by the clients in order to avoid a loss over time.

The $\text{CTE}_{0.9}(e^{-rT}\Pi_T)$, i.e. the conditional tail expectation of the insurer's discounted final profit/loss Π_T, is around 23 for product type I (No Ratchet) and around 13 for product IV (Performance Bonus) for both data-generating models, Black-Scholes-Merton and Heston, which means that in expectation over the 10% worst scenarios, for a premium of 100 paid by the client, the insurance company's expected discounted loss would be 23 or 13 respectively.

The corresponding values for the products with ratchet are in between. The difference in risk and expected return between the two data-generating models is rather small.

If the insurance company sets up a delta-hedging strategy based on the Black-Scholes-Merton model, risk is significantly reduced for all products and under both data-generating models. If the data-generating model is also the Black-Scholes-Merton model, risk is reduced to less than 10% of its unhedged value for product I (No Ratchet). On the other side, the expected profit of the insurer is also reduced.
While without hedging, the No Ratchet product appeared to have the product design that creates the most risk, with a delta-hedging strategy in place, the products with a ratchet feature (Lookback Ratchet and Remaining WBB Ratchet) now are the riskiest of the four considered designs.
The reason for this is that the delta is rather "volatile" for the products with ratchet, cf. figure 6.3 in section 6.4. Since fast changes in the delta lead to hedging errors due to time-discrete readjusting of the hedge portfolio, this leads to an increase of the risk for the two ratchet type products. This result shows the impact of the gamma (the second order derivative of the option value with respect to the underlying price) on the hedgeability of an option. The larger the gamma, the higher discretization errors and thus the higher the risk of a time-discrete delta-only hedge.

We now switch the data-generating model to the Heston model and analyze how the results of the Black-Scholes-Merton delta-only hedge change. By sole introduction of stochastic equity volatility into the capital market, the riskiness of the GLWB products, throughout all designs, is increased by roughly 50%. This is an indication of the magnitude of the model risk that the insurance company bears. While the risk significantly increases, the insurance company's expected return hardly changes, meaning that the risk-return profile of the products worsens considerably.

In the last set of simulations of the delta-only strategies, we analyze what happens if the calculation of the hedge ratios is performed within the Heston model (strategy (D-H)), i.e. we analyze what happens if the hedging is done with the "correct" model. However, the risk is only reduced by a small amount in comparison to the Black-Scholes-Merton delta-only hedge, which, in a reverse conclusion, can be interpreted as the Black-Scholes-Merton not being such a poor choice in comparison to more sophisticated models, even if it is known that the model is only a crude approximation to the real dynamics.
Nevertheless, for both products with a ratchet mechanism and the Performance Bonus product, the insurer's expected profit is significantly increased if the Heston model is used. Thus, in conclusion, by adapting the hedging model to the data-generating model, the insurance company's profit increases while its risk is only

slightly reduced.

In the last two sets of simulations, we analyze the two hedging strategies where volatility risk is also tackled. The (DV-BSM) hedge further reduces risk significantly compared to the two delta-only hedges, even though the hedge is set up with a model that assumes equity volatility to be deterministic, even constant.
Risk is further reduced by almost 50% and the results are even better than a (D-BSM) hedge with the Black-Scholes-Merton model as scenario generator, which is certainly not surprising, considering the hedge instrument used for vega hedging (a one-year ATMF straddle) also is sensitive to changes in delta and therefore (as vega and gamma tend to bear resemblance to each other) introduces a partial hedge against the gamma of the insurer's liability.
If the vega hedge is set up using the Heston model (DV-H), results improve even further. The market risk within our simulation model now is below 2% of the initial single premium paid by the client for all four product designs. Also, the differences between the riskiness of the different product designs seems more evenly, as product types II and III benefit the most in terms of reducing risk if stategy (DV-H) is used instead of (DV-BSM).

7.2.2 Distribution of hedge portfolio's value and insurer's balance

Figure 7.1 shows for each product type the empirical percentiles of the value of the insurer's hedge portfolio Π_t^{Hedge} over time, assumed a Black-Scholes-Merton delta-only hedging strategy is in place and with the data-generating model also being the Black-Scholes model with the same parameters that were used in chapter 6. If the implemented delta hedge resulted in a perfect hedge, the value of the hedge portfolio at time t, Π_t^{Hedge}, should coincide with the value of the pool of policies, Ψ_t, for any t. Thus, if Ψ_t is negative, also Π_t^{Hedge} should be negative. The value of a GLWB contract becomes negative, if, for instance, the underlying fund has performed well, the policyholder's fund assets therefore are still substantial and thus the guarantee fees (as percentage of the fund assets) paid by the policyholder are comparably high, whereas the present value of the future guarantee payments of the GLWB rider decreases as ruin is less likely and the guarantee therefore is farther out-of-the-money.

As the guaranteed withdrawal amount is never increased within product design I (No Ratchet), the scenario of a negative option value – and thus a negative hedge portfolio – is quite often seen in the first 25 years after inception of the contract. If the policyholder decides to surrender the contract in such a scenario, the insurance company has to close the hedge position, and thus realizes a loss. Hence, surrender risk is high in this case.

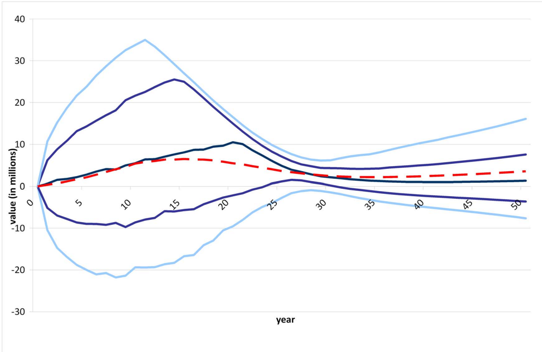
For product designs II (Lookback Ratchet) and III (Remaining WBB Ratchet), due to their ratchet mechanisms that cause the present value of the future guarantee payments of the GLWB rider to decrease less (or even to increase) if the underlying fund has performed well, the decline in the contract's value is mitigated, and therefore, while the hedge portfolio still may become negative, it is less likely to happen than with the No Ratchet product design. This effect is more distinctive with product design III, since – as seen in chapter 6 – the Remaining WBB Ratchet increases the guaranteed withdrawal amount faster than the Lookback Ratchet design.
In the fourth product design, the Performance Bonus design, the value of the contract does not vary as much as in the other products, especially the downwards variation is limited, because any surplus is reduced quickly by the performance bonus payments to the policyholder.

Although the results in the previous section indicate that product design I is the easiest to hedge (which coincides with the finding of a in comparison rather steady delta decay, cf. section 6.4), the variation of the value of the hedge portfolio during the first 25 years of the contract is the highest for this product design, which is, on the one hand due to the high initial guaranteed withdrawal amount, and on the other hand due to the missing upside potential of the guarantee. This greater variation may make the insurer more vulnerable to policyholder behavior.

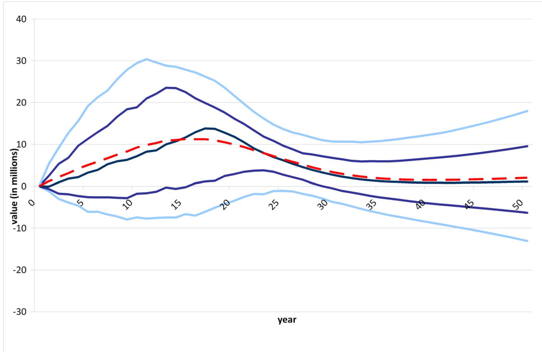
After the guarantee has triggered, the active hedging ends, and the hedge portfolio's assets are completely transfered into the money-market account, where they are compounded at the risk-free rate r and the guarantee payments are deducted annually. This proceeding is responsible for the steady development of the percentiles in the later policy years (starting between year 20 to 30, depending on the product design).

In the very end, the value of the hedge portfolio coincides with the final profit/loss of the insurer, i.e. it holds $\Pi_T = \Pi_T^{\text{Hedge}}$.

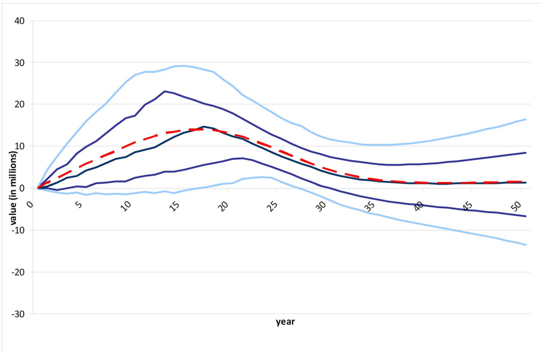
Finally, table 7.2 shows the empirical percentiles of the balance (or cumulative profit/loss) Π_t of the insurer. Product designs II and III show almost no differences, apart from the higher upside potential of design II manifesting in a slightly skewed distribution where the mean is higher than the median. However, product I seems to have a better combination of upside potential and downside risk, leading to the



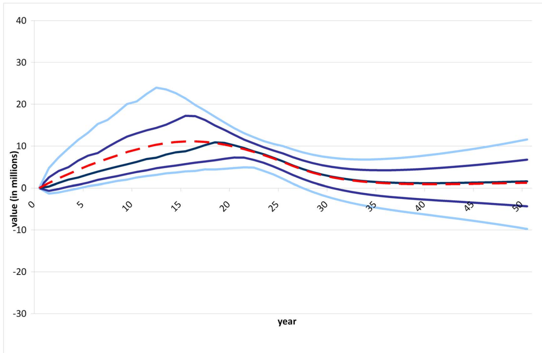
(i) No Ratchet



(ii) Lookback Ratchet



(iii) Remaining WBB Ratchet

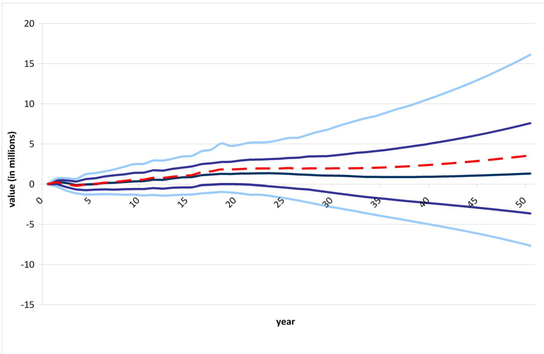


(iv) Performance Bonus

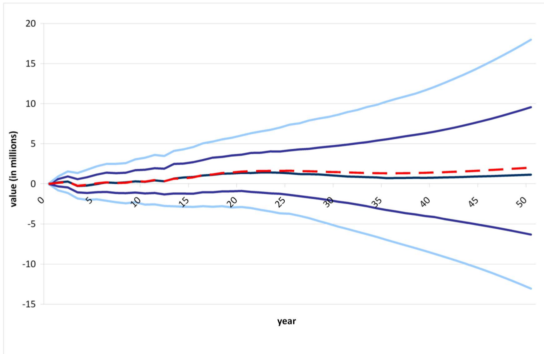
Figure 7.1: For each product design, development over time of the empirical percentiles (dark blue line represents the median, blue lines are 25^{th} - 75^{th}, light blue lines are 10^{th} - 90^{th} percentiles) and mean (red dashed line) of the hedge portfolio Π_t^{Hedge} for a pool of identical policies with a total single premium volume of US$ 100m, using the Black-Scholes model for both, delta hedging and as data-generating model.

highest mean of the final profit/loss amongst all product designs.

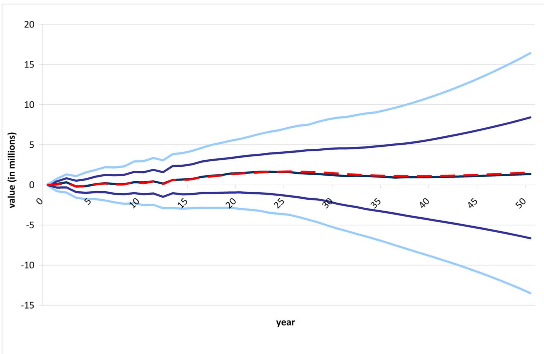
In comparison, product design IV leads to a more symmetrical distribution of the profit/loss of the insurer, with reduced risk in comparison to product design II and III, and the lowest upside potential of all products.



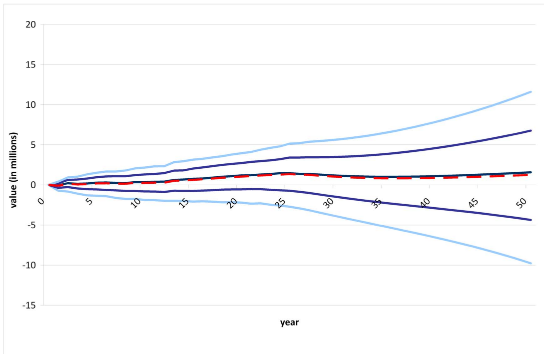
(i) No Ratchet



(ii) Lookback Ratchet



(iii) Remaining WBB Ratchet



(iv) Performance Bonus

Figure 7.2: For each product design, development over time of the empirical percentiles (dark blue line represents the median, blue lines are 25^{th} - 75^{th}, light blue lines are 10^{th} - 90^{th} percentiles) and mean (red dashed line) of the insurer's portfolio balance Π_t for a pool of identical policies with a total single premium volume of US\$ 100m, using the Black-Scholes model for both, delta hedging and as data-generating model.

7.2.3 Unmodified Vega hedge

In this last section of the chapter, we would like to add some comments about vega hedging. First of all, there is not *one* equity volatility, but quite a few different types of equity volatility (e.g. actual volatility, realized/historical volatility and (Black-Scholes-Merton) implied volatilities) that all can change in different ways, like for example in their level, skew, slope, or convexity. On the other side, there is a great variety of instruments that are liquidly traded in the market or that are commonly available over-the-counter (OTC) from investment banks that exhibit different and often complex sensitivities to changes in volatilities. Therefore, a single, unique vega-hedging strategy that is generally applicable is hard to imagine to exist.

The second comment is in regard to the shortcomings of a somewhat intuitive and straightforward way of setting up a delta-vega hedge within the Black-Scholes model: One could simply calculate the vega of the portfolio's option value Ψ_t, defined as the first-order derivate with respect to the Black-Scholes volatility parameter σ_{BS}, and then use the ratio between this vega of Ψ_t and the vega of hedge instrument, in our setting the vega of X_t, the straddle option's value, in order to set up a vega hedge consisting of a position of straddle options.
This strategy, however, results in a rather bad hedge performance (at least within our analyses, with decade-long cash-flows) due to the following reason: A change in current equity volatility under the Heston model would mean a change in short-term volatility and a much smaller change in long-term volatility, due to the mean-reversion property of the Heston model. This effect can also be seen in figure 3.2, where different implied volatility surfaces are shown that are all generated by the Heston model, but using different start parameters for the local volatility process.
Since volatility in the Black-Scholes model is assumed to be constant over time, and thus also the change in volatility is assumed to be permanent and constant over time, the change in value of a long-dated option would be significantly overestimated (at least if the Heston model is considered the "correct" model), and thus lead to a too large position in the straddle options.
The resulting hedge portfolio would lead to an increase of risk, because the vega hedge is off target and may even introduce additional sensitivity to volatility risks into the insurer's portfolio, and because the insurer additionally needs to hedge the delta of the straddle options, which leads to further sources of hedging errors. Therefore, instead of reducing the insurer's risk, this additional vega hedge would increase the insurer's risk, foiling the very idea of hedging.

To illustrate this effect, we calculated the risk measures introduced in the beginning of this chapter for the unmodified vega hedge, using the Heston model for data generation. The results are shown in table 7.2.

		Product			
Strategy	Statistic	I	II	III	IV
(D-BSM)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$	0.57	0.29	0.17	0.13
	$\text{CTE}_{0.9}(\chi)$	2.77	4.76	4.51	3.35
	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$	2.44	4.14	3.99	3.02
(DV-BSM)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$	0.82	0.81	0.75	0.47
	$\text{CTE}_{0.9}(\chi)$	1.75	2.41	3.01	1.88
mod. vega	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$	1.35	1.80	2.40	1.53
(DV-BSM)	$E_{\mathbb{P}}\left[e^{-rT}\Pi_T\right]$	10.43	7.77	6.67	3.88
	$\text{CTE}_{0.9}(\chi)$	25.29	20.07	17.54	15.12
unmod. vega	$\text{CTE}_{0.9}(e^{-rT}\Pi_T)$	23.41	18.27	15.90	13.35

Table 7.2: Results under the Heston model as data-generating model for the BSM delta hedge, the BSM delta-vega hedge using a modified version of the vega for setting up the position in the straddle options, and the BSM delta-vega hedge using the unmodified vega.

Chapter 8

Conclusion

8.1 Summary

Our main concern in the present thesis was the analysis of the properties of different types of Guaranteed Living Withdrawal Benefits (GLWB) riders, the latest guarantee feature within Variable Annuities, from both, a client's perspective and an insurer's perspective, in particular in regard to pricing and hedging of this new type of insurance contract.

In chapter 2, we first gave a high-level description of the considered Guaranteed Living Withdrawal Benefits rider options and explained their general functionality. Subsequently, we presented different step-up and ratchet mechanisms in the product designs of the GLWB options, that allow for potential (permanent) increases of the guaranteed withdrawal amount during the lifetime of the contract, given the underlying fund of the variable annuity performed sufficiently well.
In the last part of chapter 2, we described the general pricing framework that is used for evaluation of the GLWB rider option and for the purpose of finding the "fair" guaranteed withdrawal rate. We also described and discussed the assumptions that were made regarding the policyholder behavior (the policyholder's surrender behavior to be precise) and regarding the mortality of the insured.

In chapter 3, we presented the models of the financial market that we used within our analyses. The models are used for different purposes: first, as the financial market model that is assumed to be used by the insurance company for pricing and hedging of the GLWB options, in particular for determing the positions in the hedge instruments within the insurer's hedge portfolio. Second, as the (external) data-generating model in our analyses of the GLWB contracts in chapter 6, as well as in

chapter 7, where the efficiency of the hedging strategies is analyzed.
We used two models of the financial market, one with deterministic (and constant) equity volatility, the Black-Scholes-Merton model, and a second one, the Heston model, in which the instantaneous variance of the underlying asset's spot price is assumed to evolve according to an one-factor mean-reverting square-root process, similar to the process that is used within the Cox-Ingersoll-Ross model. We further described how an equivalent martingale measure can be derived for both models, which then can be used for pricing purposes within the general framework that we presented in the last section of chapter 2.

In section 3.3 of chapter 3 we described the numerical methods that we used within our analyses. Besides pricing and calculation of partial derivatives (Greeks) via Monte-Carlo techniques, we also described the valuation of European standard options under both financial market models that were presented in the previous chapter, in particular the time-efficient valuation via Fourier inversion of European standard options under the Heston model.

In the last section of chapter 3, we introduced the notion of the Black-Scholes-Merton *Implied Volatility*. Afterwards, we presented an illustration of the implied volatility surface generated by the Heston model, using the parameters that we used for our analyses. Additionally, we showed how the shape of the implied volatility surface generated by the Heston model changes if the starting value of the local variance process is changed.

As mentioned, we analyzed the hedge efficiency of different hedging strategies that may be applied by the insurance company in order to reduce the risk that originates from selling GLWB options. The analyzed strategies were presented in chapter 4, where we described different types of dynamic hedging strategies that differ mainly in the type and number of the hedge instruments that are used within the respective strategy, i.e. some strategies use the money-market account and the underlying only, while others make additional use of standard options on the underlying. For both financial market models, Black-Scholes-Merton and Heston, we specified the hedge ratios that correspond to the different hedging strategies in table 4.1 and table 4.2.

Chapter 5 deals with the design, architecture and implementation of the software solution that was used to conduct our analyses. We gave details on how we made use of Microsoft Excel as an user interface to our software solution, and on how we seamlessly embedded the functionality of the software solution into Excel by means of the open-source library `xlw`. We also specified and described all of the open-source libraries that we used within our software solution and concluded the chapter with a detailed usage example of the final software solution.

In chapter 6, we presented the first set of analyses that dealt with the pricing of the GLWB options and the distinctions in the characteristics of the product designs that originate from the ratchet and bonus features that are implemented in the respective product types. We found that different ratchet and bonus features can lead to significantly different cash-flows to the insured. Similarly, the probability that, at some point in time, the insurer has to compensate for guaranteed payments, the amount of the guaranteed payments, and the distribution of the point in time when the guarantee of the GLWB option triggers differ significantly for the different product designs, even if they all come at the same guarantee fee. We also found that the development of the Greeks - that is the sensitivities of the GLWB option value with respect to changes in certain market parameters - over time is also significantly different, depending on the selected product features.

In the subsequent chapter 7 we analyzed different (dynamic) hedging strategies (no active hedging, delta only, delta and vega) and analyzed the distribution of the insurer's cumulative profit/loss and certain risk measures hereof. We found that the insurer's risk can be significantly reduced by suitable hedging strategies, but with considerable differences between the considered hedging strategies. However, we also found that – which is in line with our findings in chapter 6 – the product design has a significant impact on how risky a product is to the insurer and on how well it is hedgeable. Thus both, the constitution of a hedging portfolio (following a certain hedging strategy) and the insurer's risk after hedging, differ significantly for the analyzed products.

We then quantified the model risk that is inherent in the use of the considered hedging strategies by using different models of the financial market for data generation and calculation of the hedge positions. The results hereof can be used as an indication for the model risk that an insurer takes who utilizes a certain model for hedging, whilst in the real world, financial markets behave differently. Within this analysis, we focused on the risk an insurer takes by assuming a constant equity volatility in the hedging model whereas in the real world equity volatility is stochastic, and showed that this risk can be substantial.

With the last analysis in chapter 7, we were able to show that, while – using the Black-Scholes-Merton model for hedging – a delta-vega hedging strategy based on a modified version of vega can lead to a significant reduction of volatility risk, even though the model assumes deterministic volatility, a somewhat more intuitive and straightforward attempt to hedge against volatility risk based on the unmodified vega may lead to results inferior to the case with no vega hedging at all, which would be clearly not the intention of implementing a vega hedge in the first place.

Our results - in particular with respect to model risk - should be of interest to both, insurers and regulators, in particular in regard to the neglected model risk that is present if the analysis of the hedge efficiency of a certain strategy is done under the same data-generating model that is used by the insurer as hedging model.

8.2 Outlook and future research

Further research could aim at extending our findings in several areas, starting with the product design of the GLWB riders: For instance, we could add designs to our analyses that include a deferment period with a guaranteed compounding of the withdrawal benefit base during that period, or we could extend our analyses to product designs that allow for partial surrender of the policyholder or a withdrawal amount below the guaranteed withdrawal amount.
Also, different forms of defining and deducting the guarantee fees would be an interesting feature to examine, e.g. guarantee fees that are not fixed as a percentage of the policyholder's current fund assets, but as a percentage of the current withdrawal benefit base, which would result in much more foreseeable guarantee payments received by the insurer.
Furthermore, the GLWB riders considered in this thesis all expect an initial single premium payment of the policyholder. Therefore it would be interesting how GLWB riders that allow for regular premium payments compare against these products in terms of riskiness to the insurer.

Similarly, the models of the financial market that we used in this work could be extended in various ways, for instance by incorporating jumps in the dynamics of the underlying – since jumps can be expected to considerably influence the performance of the considered hedging strategies. The jumps would be added on top of the modeled stochastic equity volatility, as it is the case e.g. in the Bates (1996, [4]) model, where the Heston stochastic volatility model and the Merton (1976, [23]) jump-diffusion model were combined. Also, adding stochastic interest rates to the existing model seems important, especially if product designs are considered that allow for regular premiums.
Likewise, different approaches to how the stochasticity of the actual and/or the im-

plied equity volatility are modeled are thinkable. For instance, using the SABR model (Hagan, 2002, [15]) for data-generating, while hedging is done under the Heston model, or even considering separate models for implied and actual volatility.
In addition, a systematic analysis of parameter risk appears worthwhile, with respect to the financial market parameters as well as to mortality probabilities and deterministic surrender rates.

In the present work policyholder behavior is modeled deterministic (probabilistic), and the policyholder's decision is furthermore restricted to just two options: full surrender or withdrawal of the guaranteed withdrawal amount. Therefore, it is certainly of interest how the results of our analyses change if the policyholder behavior is modeled differently: First, under a framework that allows for arbitrary withdrawal amounts between nil and all of the remaining fund assets, and, second, under optimal and/or stochastic policyholder behavior.
For instance, we could follow Le Courtois and Nakagawa (2009, [22]), who used an approach for the modeling of stochastic policyholder behavior via a mixed Poisson process (also known as *doubly stochastic Poisson process* or *Cox process*) with a stochastic intensity process that is correlated to the underlying asset's spot price. This approach models stochastic surrender rates that are influenced by market movements and that allows for the modeling of external effects like e.g. higher surrender rates due to a deteriorated reputation of the insurance company. However, within this approach, the calibration of the Poisson process could constitute a problem.
Further, an approach where the moneyness or the value of the GLWB option's value directly affects the policyholder's likelihood to surrender appears useful.
Regarding the computation of the optimal policyholder behavior, an approach similar to that used in Bacinello, Biffis, and Millossovich (2009, [3]), which is based on the Least-Squares Monte-Carlo method, seems promising.
In general, an analysis of the robustness of the hedging strategies against policyholder behavior appears worthwhile.

Also, within the scope of future research work, there is much room for refinement regarding the numerical methods used in our analyses: We could aim at improving the quality of the calculated Greeks, as well as decreasing the computational effort that is necessary to calculate them. Possible starting points would be the use of different techniques for valuating the GLWB rider options, and more efficient schemes for the numerical calculation of the partial derivatives, like, for instance, the Likelihood Ratio method (cf. Glasserman, 2003, [14]).

Hedging by product design: Likewise, it would be interesting to analyze how the insurer can reduce risk by means of product design, e.g. by offering investments funds within their products that are managed to keep the volatility of the fund's returns constant at some prespecified volatility target (so-called "VolTarget"[1] funds), or by reserving the right to switch the policyholder's assets to less risky funds (e.g. funds with an higher portion in bonds or the money market), should market volatilities increase. Ideally, such product designs would allow for pricing and hedging using a model that assumes only constant equity volatility.

[1]At the time of writing this thesis, such products were for instance offered by Société Générale, like e.g. the "SGI Vol Target BRIC" index.

Bibliography

[1] Andersen, L. (2007), Efficient Simulation of the Heston Stochastic Volatility Model, Working paper, Banc of America Securities.

[2] Bauer, D., Kling, A., and Russ, J. (2008), A Universal Pricing Framework for Guaranteed Minimum Benefits in Variable Annuities. ASTIN Bulletin, Volume: 38 (2), 621–651, November 2008.

[3] Bacinello, A. R., Biffis, E., and Millossovich, P. (2009), Regression-Based Algorithms for Life Insurance Contracts with Surrender Guarantees, to appear in Quantitative Finance, version as of April 7, 2009.

[4] Bates, D. (1996), Jump and Stochastic Volatility: Exchange Rate Processes Implicit in Deutsche Mark Options, Review of Financial Studies 9: 69–107.

[5] Bingham, N. H., Kiesel, R. (2004), Risk-Neutral Valuation: Pricing and Hedging of Financial Derivatives, Springer Verlag, Berlin.

[6] Black, F. and Scholes, M. (1973), The Pricing of Options and Corporate Liabilities. Journal of Political Economy 81, 1973, 637–654.

[7] Coleman, T.F., Kim, Y., Li, Y., and Patron, M. (2005), Hedging Guarantees in Variable Annuities Under Both Equity and Interest Rate Risks, Cornell University, New York.

[8] Coleman, T.F., Kim, Y., Li, Y., and Patron, M. (2007), Robustly Hedging Variable Annuities With Guarantees Under Jump and Volatility Risks, The Journal of Risk and Insurance, 2007, Vol. 74, No.2, 347–376.

[9] Cont, R., and Da Fonseca, J. (2002), Dynamics Of Implied Volatility Surfaces, Quantitative Finance, Vol. 2, No. 1, 2002.

[10] Cox, J.C., Ingersoll, J.E., and Ross, S.A. (1985), A Theory of the Term Structure of Interest Rates. Econometrica 53, 385-407.

[11] Eraker, B. (2004), Do stock prices and volatility jump? Reconciling evidence from spot and option prices, Journal of Finance, Volume 59, Number 3, pp. 1367–1404, June 2004.

[12] Ewald, C.-O., Poulsen, R., Schenk-Hoppe, K. R. (2007), Risk Minimization in Stochastic Volatility Models: Model Risk and Empirical Performance, Swiss Finance Institute Research Paper Series N07 - 10.

[13] Gander, W., and Gautschi, W. (2000), Adaptive Quadrature - Revisited. BIT, 40(1): 84–101, March 2000.

[14] Glasserman, P. (2003), Monte Carlo Methods in Financial Engineering, Springer, Berlin.

[15] Hagan, P., Kumar, D., Lesniewski, A., and Woodward, D. (2002), Managing smile risk, Wilmott Magazine, September, 84-108, 2002.

[16] Heston, S. (1993), A closed form solution for options with stochastic volatility with applications to bond and currency options, Review of Financial Studies 6, no. 2, 327-344.

[17] Holz, D., Kling, A., and Ru, J. (2007), GMWB For Life An Analysis of Lifelong Withdrawal Guarantees. Working Paper, Ulm University, 2007.

[18] Hull, J.C. (2008), Options, Futures and Other Derivatives, Prentice Hall.

[19] Joshi, M. S. (2003), The Concepts and Practice of Mathematical Finance, Cambridge University Press.

[20] Joshi, M. S. (2004), C++ Design Patterns and Derivatives Pricing, Cambridge University Press.

[21] Kahl, C. and Jckel, P. (2005), Not-so-complex logarithms in the Heston model, Wilmott Magazine, 94-103, September 2005.

[22] Le Courtois, O., and Nakagawa, H. (2009), Surrender Risk and the Default of Insurance Companies, working paper, EM Lyon Business School, France, and Hitotsubashi University, Japan.

[23] Merton, R. (1976), Option pricing when underlying stock return are discontinuous, Journal of Financial Economics 3: 125–144.

[24] Mikhailov, S. and Ngel, U. (2003), Hestons Stochastic Volatility Model Implementation,Calibration and Some Extensions, Wilmott Magazine (2003).

[25] Lord, R., Koekkoek, R., and van Dijk, D. (2008), A Comparison of biased simulation schemes for stochastic volatility models, Working Paper, Tinbergen Institute.

[26] Milevsky, M., and Posner, S.E. (2001), The Titanic Option: Valuation of the Guaranteed Minimum Death Benefit in Variable Annuities and Mutual Funds. The Journal of Risk and Insurance, Vol. 68, No. 1, 91-126, 2001.

[27] Milevsky, M. and Salisbury, T.S. (2006), Financial Valuation of Guaranteed Minimum Withdrawal Benefits. Insurance: Mathematics and Economics: 38, 21-38, 2006.

[28] Taleb, N. (1997), Dynamic Hedging: Managing Vanilla and Exotic Options, Wiley.

[29] Van Haastrecht, A., Pelsser, A. (2008), Efficient, Almost Exact Simulation of the Heston Stochastic Volatility Model, Working paper, Delta Lloyd and University of Amsterdam.

[30] Wilmott, P. (2006), Paul Wilmott on Quantitative Finance, Wiley & Sons.

[31] Wong, B., Heyde, C. C. (2006), On changes of measure in stochastic volatility models, Journal of Applied Mathematics and Stochastic Analysis, Volume 2006, Article ID 18130, 1-13.

Ehrenwörtliche Erklärung

Ich erkläre hiermit ehrenwörtlich, dass ich die vorliegende Arbeit selbstständig angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht. Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ich bin mir bewusst, dass eine unwahre Erklärung rechtliche Folgen haben wird.

Ulm, den 22. September 2009 ______________________________

(Unterschrift)