Policy Shifts and Markov-Switching in Financial Markets

Inauguraldissertation
zur Erlangung des akademischen Grades eines
Doktors der Wirtschaftswissenschaften
durch die

Wirtschaftswissenschaftliche Fakultät
der
Westfälischen Wilhelms-Universität Münster

vorgelegt von Diplom-Mathematiker Diplom-Volkswirt
Gerrit Reher
aus Münster

Dekan: Prof. Dr. Thomas Apolte
Erstgutachter: Prof. Dr. Bernd Wilfling
Zweitgutachter: Prof. Dr. Martin Bohl
Tag der mündlichen Prüfung: 22.11.2010

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Behavior in financial markets is affected by different kinds of parameters. On the one hand, there are legal framework conditions made by political authorities. As financial market participants act in an international environment, legal framework conditions must jointly be coordinated across different economies in order to be effective. Furthermore, the behavior of the market participants can be influenced by prospect of change in this framework, e.g. the key interest rate of the central banks. On the other hand, there is a broad range of expectations that influence prices and behavior on financial markets such as expectation of the business cycle, exchange rate changes or performance of the stock market. Moreover, the attitude of market participants towards future events or developments influences behavior and thus prices on financial markets.

Economic and econometric models should be able to capture changes in legal framework conditions. Furthermore, these models should be able to cope behavior modifications referring to policy switches or environmental changes. Making inference on unobservable variables, anticipation of environmental changes and market sentiment can be revealed. Neglecting environmental changes, considerable mispricing can occur or misleading conclusions can be drawn.

We distinguish two cases of environmental changes: on the one hand, there are permanent changes, e.g. the German reunification or the European monetary union. On the other hand, there are temporary changes. Examples are change of government, change of legal framework conditions or an attitude change of market participants during a financial crisis. Suitable models should allow for such dynamic behavior.

In chapter two we analyze a situation where the political authorities of two open economies plan to enter a currency union in the future. This is done in a scenario that is comparable to the introduction of the euro for new EMU entrants. Future EMU accession countries are very likely to enter the currency union with similar terms. In such a scenario we analyze the dynamics of zero-coupon bond options. Therefore, we make use of recent theoretical work on the continuous-time dynamics of interest-rate differentials between the economies involved and derive a closed-form pricing formula for a European call option on zero-coupon bonds. In a Monte-Carlo simulation study we show that significant option-pricing errors can occur when the key features of interest-rate dynamics during the run-up to the currency union are ignored. The developed interest-rate dynamics and zero-coupon bond option formulas should be of particular relevance for market participants of European economies that have not yet become EMU members, but may join the EMU in the future, e.g. Poland, Sweden or the UK. We attach importance to the traceability to our results: MATLAB programming codes with extensive comments in appendix A make our results open to scrutiny.

In chapter three we establish a class of regime-switching models that capture different specifications of models with autoregressive conditional heteroskedasticity in each regime.

Regime-switching models are well-established models in the econometric literature. The main advantages of regime-switching models are that they (a) provide a high goodness of fit in face of policy switches, environmental or behavior changes, (b) can visualize unobservable variables such as credibility of announcements or anticipation of forthcoming events, and (c) allow valid forecasts that take the prospect of changes into account.

In financial data several stylized facts are observable. There are alternating episodes of high and low volatility. This phenomenon is called clustering, and autoregressive conditional heteroskedasticity (ARCH) models have been developed to incorporate this stylized fact. Furthermore, there are two other prominent empirical findings on financial data. At first, after a negative shock the increase of volatility is higher than after a positive shock of the same magnitude. This asymmetric behavior is called leverage effect. The second stylized fact is that stock returns are leptokurtic. There are many variations of the ARCH model that have been developed to incorporate these empirical findings. The most prominent models are GARCH, ARCH-in-mean and EGARCH models. It is possible to nest the most prominent variations of ARCH models into one parametric family.

Combinations of the regime-switching approach and conditional heteroskedasticity models only exist for GARCH and EGARCH models. We extend the existing literature by incorporating the regime-switching approach into the whole parametric family of conditional heteroskedasticity models. Thus we allow for e.g. TGARCH-TGARCH models or even different specifications of models in each regime, e.g. EGARCH-GARCH models. We apply this new model class to excess returns of the German stock market. The MATLAB programming code for this new model class is provided in appendix B. Extensive comments in this code explain how to modify the code of the general model for the estimation of particular specifications.

In chapter four we use the model that is derived in chapter three to analyze the impact of short selling constraints. After the collapse of Lehman Brothers, the BaFin enacted uncovered short selling restrictions for certain shares. All involved incorporations belong to the financial sector. The aim of the chosen measure was to stabilize the financial market and prevent a breakdown of the banking system. The aim of our analysis is to investigate the intended stabilizing effect of this short selling constraint. In fact, we shed some light on causality of behavioral change: does market behavior induce BaFin to constrain the uncovered short-selling or does BaFin's restriction affect market behavior? We show that there is no evidence for the desired impact of the measure. The MATLAB programming code for this application of the new regime-switching model class is given in appendix C.

Chapter five summarizes the main findings of this thesis and concludes outlining possible paths for future research.

Chapter 2

An exact pricing formula for European call options on zero-coupon bonds in the run-up to a currency union

2.1 Introduction

Closed-form solutions for European options on pure discount bonds and on discount bond portfolios have been established in a classical option-pricing framework by Jamshidian (1989). Using Vasicek's (1977) mean-reverting Gaussian interest-rate model and assuming that the term structure is completely determined by the value of the instantaneous interest rate, the author derives a closed-form Black-Scholes-type pricing formula. In this thesis we leave this classical option-pricing framework and modify Jamshidian's (1989) results by taking into account that a country's interest-rate dynamics—which is relevant to option-pricing—may be closely linked to the interest rates of the partner countries via the current exchange-rate system.

Two alternative exchange-rate arrangements under which the interest rates of the countries involved are intimately connected to each other are well-documented in the economic literature. The first arrangement is a so-called exchange-rate target zone as introduced by Krugman (1991). The dynamic interrelationships between the participating countries' interest rates (of arbitrary terms) are derived in Svensson (1991a, 1991b). The second exchange-rate arrangement is represented by the time period prior to the fixing of a currently floating exchange rate on a given future date at a publicly announced fixing parity. In a stylized model, Wilfling (2003) derives the term structure of the bilateral interest-rate differentials under such an exchange-rate regime, thus providing dynamic equations for the link governing the interest rates in both countries.

Owing to its political topicality, this thesis focusses on the second of the just-mentioned exchange-rate regimes. In practice, the introduction of a common currency is typically initiated by a switch in exchange-rate system from (more or less) floating exchange rates to completely fixed rates. For example, the introduction of the euro among the member countries of the European Monetary Union (EMU) was implemented by the irreversible fixing of the EMU countries' bilateral exchange rates at their respective central parities from the European Exchange Rate Mechanism (ERM) from 1 January 1999 onwards. Since then, the same exchange-rate fixing procedure has been applied to all later EMU entrants, and it is very likely that future EMU accession countries will also enter the currency union at conversion rates equal to their ERM central parities vis-à-vis the euro.

Up to date, the EMU consists of 16 countries including the large economies of France, Germany, Italy and Spain. There are, however, several other major European economies that have not yet become EMU members, but are likely to adopt the euro in the future (like Poland, Sweden and the UK). It is the financial market participants operating in these future EMU accession countries to whom our closed-form formulas for zero-coupon bond options established below should be of particular relevance.

The remainder of the chapter is organized as follows. Section 2.2 reviews some previous results on exchange-rate dynamics and on international interest-rate differentials in the run-up to a currency union. Based on these results we derive the interest-rate dynamics crucial to our option-pricing problem. In Section 2.3 we first value zero-coupon bonds under the new interest-rate dynamics and then value European call options on these pure discount bonds. In Section 2.4 we conduct a Monte-Carlo simulation study in order to assess the validity of our option-pricing formula. Section 2.5 offers some concluding comments.

2.2 Previous results on exchange-rate and interest-rate dynamics

In what follows we consider a world with two open economies under perfect capital mobility and assume the domestic economy to be small. In this general setting, let the political authorities of the two economies decide to create a currency union in the future. On the analogy of Stage III of EMU, the authorities therefore announce at date t_A to irreversibly fix the exchange rate from the future date t_S onwards (i.e. $t_A < t_S$) at the specific exchange-rate parity s.

The exchange-rate dynamics under such a time-contingent switch in exchange-rate regime has been characterized in the literature by various authors on the basis of the well-known monetary exchange-rate model with flexible prices (see, among others, Sutherland, 1995; De Grauwe et al., 1999; Wilfling and Maennig, 2001). In this continuous-time equilibrium model with rational expectations, the logarithmic spot exchange rate at time $t, x(t)$, equals the sum of two components: (a) an exogenously given 'macroeconomic fundamental variable' $k(t)$, and (b) a speculative term representing the expected (instantaneous) rate of change in the nominal exchange rate:

$$x(t) = k(t) + \alpha \cdot \frac{E[dx(t)|\phi_t]}{dt}, \alpha > 0. \tag{2.1}$$

In Eq. (2.1), $E[\cdot|\phi_t]$ denotes the expectation operator conditional on the information set ϕ_t which contains all information available to market participants at time t. The parameter α represents the semi-elasticity of money demand with respect to the instantaneous interest rate. Alternatively, α may simply be interpreted as a parameter weighting the fundamental component against the speculative motives for currency valuation.

In the monetary flex-price model, the fundamental $k(t)$ represents an aggregate of given macroeconomic variables (such as the domestic and foreign money supplies and outputs) as well as stochastic shocks to money demand. Via the domestic and foreign money supplies, $k(t)$ is under direct control of the two central banks involved and, prior to the fixing-date t_S, $k(t)$ should follow an appropriate continuous-time stochastic process. In this thesis, we model the evolution of $k(t)$ over time (up to t_S) by a driftless Brownian motion with stochastic differential representation

$$dk(t) = \tilde{\sigma} \cdot d\bar{W}(t), t < t_S, \tag{2.2}$$

with (constant) infinitesimal standard deviation $\tilde{\sigma} > 0$ and $d\bar{W}(t)$ the increment of standard Wiener process. The driftless Brownian motion is particularly adequate when modeling a situation in which the central banks refrain from intervening in the foreign exchange market. Thus, modeling the fundamental $k(t)$ as in Eq. (2.2) is consistent with assuming a pure free-float exchange-rate regime prior to the currency union.[1]

Given the specification (2.2) of the fundamental process, the general law of exchange-rate dynamics in Eq. (2.1) constitutes a stochastic differential equation. This can be solved by standard techniques and the imposition of adequate economic constraints, which appropriately reflect the anticipations of foreign exchange market participants with regard to the entrance of both economies into the currency union on date t_S at the parity s. Ruling out currency-arbitrage opportunities at the moment of transition into the currency

[1] More interventionist exchange-rate policies prior to the currency union can be modeled by specifying alternative driving processes for the fundamental $k(t)$. Sondermann et al. (2010), for example, model (a) an exchange-rate system of managed floating and (b) a system of continuously increasing interventionist activity towards the entrance into the currency union by letting the fundamental $k(t)$ follow an Ornstein-Uhlenbeck process and a scaled Brownian bridge, respectively.

union (i.e. imposing the condition $\lim_{t \to t_S} x(t) = s$ with probability 1), it is straightforward to check that the (bubble-free) solution to Eq. (2.1) is given by

$$x(t) = \begin{cases} k(t) & \text{for } t < t_A \\ k(t) + e^{(t-t_S)/\alpha} \cdot [s - k(t)] & \text{for } t \in [t_A, t_S) \\ s & \text{for } t \geq t_S \end{cases} . \tag{2.3}$$

Next, we establish the interest-rate dynamics in the two economies by adopting the model set-up presented in Wilfling (2003). Let $P(t,T)$ denote the price at time t of a domestic zero-coupon bond maturing at time T, $t \leq T$, with unit maturity value $P(T,T) = 1$, and define $P^*(t,T)$ to be the analogous price of a foreign-currency discount bond. Furthermore, let us denote the domestic and the foreign instantaneous short rates at time t by $r(t)$ and $r^*(t)$, respectively, and suppose that the small domestic economy cannot affect the foreign short rate by economic policy, but has to accept $r^*(t)$ as exogenously given. We further assume (a) perfect international capital mobility, and (b) that international investors consider the domestic and the foreign discount bonds as perfect substitutes. Under this scenario the following form of the uncovered interest parity condition should hold among the instantaneous short rates at all points in time:[2]

$$\text{SRD}(t) \equiv r(t) - r^*(t) = \frac{E[dx(t)|\phi_t]}{dt}. \tag{2.4}$$

The exchange-rate path (2.3) plus the uncovered interest parity condition (2.4) now allow us to represent the short-rate differential SRD(t) in closed form. To this end, we apply Ito's lemma to the exchange-rate path (2.3), which yields the stochastic differential $dx(t)$. After

[2] We understand the uncovered interest parity as an equilibrium condition in the sense that the foreign exchange market is in equilibrium when deposits of all currencies offer the same expected rate of return (with respect to the country-specific short rates). This is (approximately) the case if the short-rate differential equals the expected instantaneous rate of change in the exchange rate.

taking conditional expectations and dividing by dt we obtain the short-rate differential during the run-up to the currency union:

$$\mathrm{SRD}(t) = r(t) - r^*(t) = \begin{cases} 0 & \text{for } t < t_A \\ e^{(t-t_S)/\alpha} \cdot \dfrac{s - k(t)}{\alpha} & \text{for } t \in [t_A, t_S) \\ 0 & \text{for } t \geq t_S \end{cases} . \tag{2.5}$$

Finally, we follow Vasicek (1977) and let the exogenously given foreign short rate $r^*(t)$ evolve according to a mean-reverting Ornstein-Uhlenbeck process with stochastic differential

$$dr^*(t) = b(c - r^*(t))dt + \sigma dW_1(t), \tag{2.6}$$

where b, c, σ are positive constants and $W_1(t)$ denotes a standard Wiener process. Given the initial value $r_0^* \equiv r^*(0)$, the solution to Eq. (2.6) is known to be

$$r^*(t) = (r_0^* - c)e^{-bt} + c + A(t), \tag{2.7}$$

with $A(t)$ defined as

$$A(t) \equiv \sigma e^{-bt} \int_0^t e^{bs} dW_1(s).$$

Inserting Eq. (2.7) into Eq. (2.5) and taking as given the initial value $k_0 \equiv k(0)$ for the fundamental process (2.2), we obtain the domestic short-rate process:

$$r(t) = \begin{cases} (r_0^* - c)e^{-bt} + c + A(t) & \text{for } t < t_A \\ (r_0^* - c)e^{-bt} + c + A(t) + e^{(t-t_S)/\alpha} \cdot \dfrac{s - k_0 - \tilde{\sigma}\bar{W}(t)}{\alpha} & \text{for } t \in [t_A, t_S) \\ (r_0^* - c)e^{-bt} + c + A(t) & \text{for } t \geq t_S \end{cases} . \tag{2.8}$$

In what follows we assume that the Wiener processes $\bar{W}(t)$ and $W_1(t)$ from the Eqs. (2.2) and (2.6) are interrelated by $\bar{W}(t) = \beta W_1(t) + \sqrt{1-\beta^2} W_2(t)$ with $-1 \leq \beta \leq 1$ and $W_2(t)$ being an intermediary Wiener process independent of $W_1(t)$. Via this assumption, we allow our driving Wiener processes $\bar{W}(t)$ and $W_1(t)$ to be correlated with constant correlation coefficient β (i.e. $\mathrm{Corr}[\bar{W}(t), W_1(t)] = \beta$ for all t).

2.3 Bond and option valuation

For the purpose of bond and option valuation, we denote the (risk neutral) martingale measure by Q. Following the well-established martingale modeling approach, we specify our short-rate dynamics from Eq. (2.8) under Q.[3] In Section 2.3.1 we first value zero-coupon bonds under our Q-dynamics for the short rate and then proceed with the pricing of zero-coupon bond options in Section 2.3.2.

2.3.1 Valuation of zero-coupon bonds

The price $P(\theta, T)$ at time θ of a domestic zero-coupon bond maturing at time T is given by the risk-neutral valuation formula

$$P(\theta, T) = E^Q \left[e^{-\int_\theta^T r(t)dt} | \phi_\theta \right] \tag{2.9}$$

(see for example Björk, 2004, p. 322). To calculate this conditional expectation under Q, three distinct cases concerning the dates θ and T have to be distinguished:

Case 1: $\theta < t_A$ or $\theta \geq t_S$.

Case 2: $t_A \leq \theta < t_S$ and $T < t_S$.

Case 3: $t_A \leq \theta < t_S$ and $T \geq t_S$.

Case 1 represents the following two extreme scenarios. (a) If $\theta < t_A$, the prospective currency union has not yet been announced so that financial market participants are currently not aware of the future currency union. (b) If $\theta \geq t_S$, our two economies already live in the currency union. In contrast to these two scenarios, the Cases 2 and 3 represent a transitional setting (the so-called interim period) in that for $t_A \leq \theta < t_S$ the currency union has already been announced to financial market participants, but has not yet been implemented. However, according to the Eqs. (2.3) and (2.5), the mere announcement of entering a currency union in the future already affects today's exchange-rate as well as

[3]For alternative classical models of the Q-dynamics for the short rate $r(t)$ see, among others, Vasicek (1977), Cox et al. (1985), Ho and Lee (1986), Hull and White (1994).

today's domestic short-rate dynamics and, consequently, also has an impact on today's pricing of zero-coupon bonds. Moreover, as will become evident below, the exact bond-pricing formula additionally hinges on the question of whether the maturity date T lies before or after the start of the currency union (Case 2 or Case 3).

The calculation of the conditional expectation on the right-hand side of Eq. (2.9) requires knowledge of the probability distribution of the short rate $r(t)$. In view of Eq. (2.8) it is straightforward to verify that $\{r(t)\}$ is a Gaussian process and is thus completely characterized in terms of its first and second moments. Setting the present date $\theta = 0$ for ease of notation, we summarize the expectations, variances and covariances of $\{r(t)\}$ in the following lemma.

Lemma 2.3.1 *The expectations, variances and covariances of the short-rate process $\{r(t)\}$ are given as follows:*

(a) For $t < t_A$ we have

$$\begin{aligned} E[r(t)] &= (r_0^* - c)e^{-bt} + c, \\ \mathrm{Var}[r(t)] &= \frac{\sigma^2}{2b}(1 - e^{-2bt}), \\ \mathrm{Cov}[r(t), r(t')] &= \frac{\sigma^2}{2b} e^{-b(t+t')}(e^{2b\min\{t,t'\}} - 1). \end{aligned}$$

(b) For $t_A \leq t, t' < t_S$ we have

$$\begin{aligned} E[r(t)] &= (r_0^* - c)e^{-bt} + c + e^{\frac{t-t_S}{\alpha}} \frac{s - k_0}{\alpha}, \\ \mathrm{Var}[r(t)] &= \frac{\sigma^2}{2b}(1 - e^{-2bt}) + \frac{\tilde{\sigma}^2}{\alpha^2} e^{\frac{2t-2t_S}{\alpha}} t \\ &\quad - 2\beta \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} e^{-bt} e^{\frac{t-t_S}{\alpha}} \min\{e^{2bt} - 1, t\}, \\ \mathrm{Cov}[r(t), r(t')] &= \frac{\sigma^2}{2b} e^{-b(t+t')}(e^{2b\min\{t,t'\}} - 1) + \frac{\tilde{\sigma}^2}{\alpha^2} e^{\frac{t+t'-2t_S}{\alpha}} \min\{t, t'\} \\ &\quad - \beta \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} e^{-bt'} e^{\frac{t-t_S}{\alpha}} \min\{e^{2bt'} - 1, t\} - \beta \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} e^{-bt} e^{\frac{t'-t_S}{\alpha}} \min\{e^{2bt} - 1, t'\}. \end{aligned}$$

(c) For $t \geq t_S$ we have

$$\begin{aligned} E[r(t)] &= (r_0^* - c)e^{-bt} + c, \\ \text{Var}[r(t)] &= \frac{\sigma^2}{2b}(1 - e^{-2bt}), \\ \text{Cov}[r(t), r(t')] &= \frac{\sigma^2}{2b}e^{-b(t+t')}(e^{2b\min\{t,t'\}} - 1). \end{aligned}$$

Next, we address the integral of the short rate $r(t)$ appearing on the right-hand side of Eq. (2.9). The following lemma provides helpful insight into the probabilistic nature of this integral. Its proof is sketched in Elliot and Kopp (2005, p. 265).

Lemma 2.3.2 *Let $\{X(t)\}$ be a Gaussian process with continuous sample paths and mean and covariance functions $m(t) \equiv E[X(t)]$ and $n(t,t') \equiv \text{Cov}[X(t), X(t')]$. Then, the process defined by*

$$Z(t) \equiv \int_0^t X(s)ds$$

is also a Gaussian process with mean and covariance functions given by $\int_0^t m(s)ds$ and $\int_0^t \int_0^{t'} n(u,v)dudv$, respectively.

Lemma 2.3.2 implies that the process defined by $\int_0^T r(t)dt$ is a Gaussian process. Thus, the random variable $\exp\{-\int_0^T r(t)dt\}$ appearing on the right-hand side of Eq. (2.9) has a lognormal distribution, the expectation of which is uniquely determined by the expectation and the variance of $\int_0^T r(t)dt$. These latter moments follow from Lemma 2.3.2 and are compiled in the following lemma.

Lemma 2.3.3 *For the three cases considered above the expectations and variances of $\int_0^T r(t)dt$ are given as follows:*

Case 1: For $\theta = 0 < t_A$ or for $\theta = 0 \geq t_S$ we have

$$\begin{aligned} E\left[\int_0^T r(t)dt\right] &= cT + \frac{r_0^* - c}{b}(1 - e^{-bT}), \\ \text{Var}\left[\int_0^T r(t)dt\right] &= \frac{\sigma^2}{2b^3}(2bT - 3 + 4e^{-bT} - e^{-2bT}). \end{aligned}$$

Case 2: For $t_A \leq \theta = 0 < t_S$ and $T < t_S$ we have

$$\begin{aligned} E\left[\int_0^T r(t)dt\right] &= cT + \frac{r_0^* - c}{b}(1 - e^{-bT}) + (s - k_0)(e^{\frac{T - t_S}{\alpha}} - e^{\frac{-t_S}{\alpha}}), \\ \mathrm{Var}\left[\int_0^T r(t)dt\right] &= \frac{\sigma^2}{2b^3}(2bT - 3 + 4e^{-bT} - e^{-2bT}) + \frac{\tilde{\sigma}^2\alpha}{2}e^{\frac{2T - 2t_S}{\alpha}}(2\frac{T}{\alpha} - 3 + 4e^{-\frac{T}{\alpha}} - e^{-2\frac{T}{\alpha}}) \\ &\quad - 2\beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}e^{-\frac{t_S}{\alpha}}\int_0^T\int_0^T e^{\frac{v}{\alpha}}e^{-bu}\min\{e^{2bu} - 1, v\}dudv. \end{aligned}$$

Case 3: For $t_A \leq \theta = 0 < t_S$ and $T > t_S$ we have

$$\begin{aligned} E\left[\int_0^T r(t)dt\right] &= cT + \frac{r_0^* - c}{b}(1 - e^{-bT}) + (s - k_0)(1 - e^{\frac{-t_S}{\alpha}}), \\ \mathrm{Var}\left[\int_0^T r(t)dt\right] &= \frac{\sigma^2}{2b^3}(2bT - 3 + 4e^{-bT} - e^{-2bT}) + \frac{\tilde{\sigma}^2\alpha}{2}(2\frac{t_S}{\alpha} - 3 + 4e^{-\frac{t_S}{\alpha}} - e^{-2\frac{t_S}{\alpha}}) \\ &\quad - 2\beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}e^{-\frac{t_S}{\alpha}}\int_0^{t_S}\int_0^T e^{\frac{v}{\alpha}}e^{-bu}\min\{e^{2bu} - 1, v\}dudv. \end{aligned}$$

Finally, we exploit the well-known result that for a normally distributed random variable $X \sim N(\mu, \sigma^2)$ the transformed variable $Y \equiv \exp\{-X\}$ has a lognormal distribution with expected value $E(Y) = \exp\{-\mu + \sigma^2/2\}$. Using this relationship, we are able to calculate the expectation on the right-hand side of Eq. (2.9) and thus obtain our bond-price formulas which we compile in the following proposition.

Proposition 2.3.4 *In the run-up to a currency union, the price $P(\theta, T)$ at time $\theta = 0$ of a domestic zero-coupon bond maturing at time T is given as follows:*

Case 1: For $\theta = 0 < t_A$ or for $\theta = 0 \geq t_S$ the bond price is given by

$$P(0,T) = \exp\left\{-cT + \frac{r_0^* - c}{b}(e^{-bT} - 1) + \frac{\sigma^2}{4b^3}(2bT - 3 + 4e^{-bT} - e^{-2bT})\right\}.$$

Case 2: For $t_A \leq \theta = 0 < t_S$ and $T < t_S$ the bond price is given by

$$\begin{aligned} P(0,T) &= \exp\left\{-cT + \frac{r_0^* - c}{b}(e^{-bT} - 1) - (s - k_0)(e^{\frac{T - t_S}{\alpha}} - e^{\frac{-t_S}{\alpha}})\right. \\ &\quad + \frac{\sigma^2}{4b^3}(2bT - 3 + 4e^{-bT} - e^{-2bT}) + \frac{\tilde{\sigma}^2\alpha}{4}e^{\frac{2T - 2t_S}{\alpha}}(2\frac{T}{\alpha} - 3 + 4e^{-\frac{T}{\alpha}} - e^{-2\frac{T}{\alpha}}) \\ &\quad \left. - \beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}e^{-\frac{t_S}{\alpha}}\int_0^T\int_0^T e^{\frac{v}{\alpha}}e^{-bu}\min\{e^{2bu} - 1, v\}dudv\right\}. \end{aligned}$$

Case 3: For $t_A \leq \theta = 0 < t_S$ and $T \geq t_S$ the bond price is given by

$$\begin{aligned} P(0,T) &= \exp\Big\{ -cT + \frac{r_0^* - c}{b}(e^{-bT} - 1) - (s - k_0)(1 - e^{\frac{-t_S}{\alpha}}) \\ &+ \frac{\sigma^2}{4b^3}(2bT - 3 + 4e^{-bT} - e^{-2bT}) + \frac{\tilde{\sigma}^2\alpha}{4}(2\frac{t_S}{\alpha} - 3 + 4e^{-\frac{t_S}{\alpha}} - e^{-2\frac{t_S}{\alpha}}) \\ &- \beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} e^{-\frac{t_S}{\alpha}} \int_0^{t_S}\int_0^T e^{\frac{v}{\alpha}} e^{-bu} \min\{e^{2bu} - 1, v\} du dv \Big\}. \end{aligned}$$

2.3.2 Valuation of call options on zero-coupon bonds

We now consider a European call option on a zero-coupon bond with maturity date T. Denoting the exercise date of the option by τ ($\tau < T$) and the option's strike price by K, we can write its contract function as $\max\{P(\tau,T) - K, 0\}$, and the risk-neutral valuation formula of the European call option is given by

$$C(0) = E^Q\left[e^{-\int_0^\tau r(t)dt} \cdot \max\{P(\tau,T) - K, 0\} | \phi_0\right], \tag{2.10}$$

where again we have set the current date equal to 0 for ease of notation.

It is important to note here that the bond price $P(\tau,T)$ constitutes a random variable for all exercise dates $\tau > 0$. Thus, the calculation of the expected value on the right-hand side of Eq. (2.10) requires knowledge of the following three distributions:

(a) the distribution of $\int_0^\tau r(t)dt$,

(b) the distribution of $P(\tau,T)$,

(c) the joint distribution of $\int_0^\tau r(t)dt$ and $P(\tau,T)$.

Since the normal distribution of $\int_0^\tau r(t)dt$ has already been characterized by Lemma 3.3, it remains to find the distributions of the random variable from item (b) and the random vector from item (c). To this end, the following four cases concerning the dates τ and T have to be distinguished:

Case (a): $0 < t_A$ or $0 \geq t_S$.

Case (b): $t_A \le 0 < \tau < T < t_S$.

Case (c): $t_A \le 0 < \tau < t_S \le T$.

Case (d): $t_A \le 0 < t_S \le \tau < T$.

According to Proposition 2.3.4 we can write the *stochastic* bond prices $P(\tau, T)$ as follows.

Lemma 2.3.5 *Case (a): For $0 < t_A$ or for $0 \ge t_S$ the bond price can be written as*

$$\begin{aligned} P(\tau,T) &= \exp\Big\{-c(T-\tau) + \frac{r^*(\tau)-c}{b}\left[e^{-b(T-\tau)}-1\right] \\ &\quad + \frac{\sigma^2}{4b^3}\left[2b(T-\tau)-3+4e^{-b(T-\tau)}-e^{-2b(T-\tau)}\right]\Big\}. \end{aligned}$$

Case (b): For $t_A \le 0 < \tau < T < t_S$ the bond price can be written as

$$\begin{aligned} P(\tau,T) &= \exp\Big\{-c(T-\tau) + \frac{r^*(\tau)-c}{b}\left[e^{-b(T-\tau)}-1\right] - (s-k(\tau))\left[e^{\frac{T-t_S}{\alpha}} - e^{\frac{\tau-t_S}{\alpha}}\right] \\ &\quad + \frac{\sigma^2}{4b^3}\left[2b(T-\tau)-3+4e^{-b(T-\tau)}-e^{-2b(T-\tau)}\right] \\ &\quad + \frac{\tilde{\sigma}^2\alpha}{4}e^{\frac{2T-2t_S}{\alpha}}\left[2\frac{T-\tau}{\alpha}-3+4e^{-\frac{T-\tau}{\alpha}}-e^{-2\frac{T-\tau}{\alpha}}\right] \\ &\quad - \beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}e^{\frac{\tau-t_S}{\alpha}}\int_0^{T-\tau}\int_0^{T-\tau} e^{\frac{v}{\alpha}}e^{-bu}\min\{e^{2bu}-1, v\}dudv\Big\}. \end{aligned}$$

Case (c): For $t_A \le 0 < \tau < t_S \le T$ the bond price can be written as

$$\begin{aligned} P(\tau,T) &= \exp\Big\{-c(T-\tau) + \frac{r^*(\tau)-c}{b}\left[e^{-b(T-\tau)}-1\right] - (s-k(\tau))\left[1 - e^{\frac{\tau-t_S}{\alpha}}\right] \\ &\quad + \frac{\sigma^2}{4b^3}\left[2b(T-\tau)-3+4e^{-b(T-\tau)}-e^{-2b(T-\tau)}\right] \\ &\quad + \frac{\tilde{\sigma}^2\alpha}{4}\left[2\frac{t_S-\tau}{\alpha}-3+4e^{-\frac{t_S-\tau}{\alpha}}-e^{-2\frac{t_S-\tau}{\alpha}}\right] \\ &\quad - \beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}e^{\frac{\tau-t_S}{\alpha}}\int_0^{t_S-\tau}\int_0^{T-\tau} e^{\frac{v}{\alpha}}e^{-bu}\min\{e^{2bu}-1, v\}dudv\Big\}. \end{aligned}$$

Case (d): For $t_A \leq 0 < t_S \leq \tau < T$ the bond price can be written as

$$\begin{aligned} P(\tau, T) &= \exp\left\{ -c(T-\tau) + \frac{r^*(\tau) - c}{b} \left[e^{-b(T-\tau)} - 1 \right] \right. \\ &\qquad \left. + \frac{\sigma^2}{4b^3} (2b(T-\tau) - 3 + 4e^{-b(T-\tau)} - e^{-2b(T-\tau)}) \right\}. \end{aligned}$$

One immediate consequence of Lemma 2.3.5 is that for each of the four Cases (a) to (d) the required joint distribution of $\int_0^\tau r(t)dt$ and $P(\tau, T)$ is completely characterized in terms of the following joint distributions:

Case (a): $\left(\int_0^\tau r(t)dt, r^*(\tau)\right)$.

Case (b): $\left(\int_0^\tau r(t)dt, r^*(\tau)\left[1 - e^{-b(T-\tau)}\right] + k(\tau)\left[be^{\frac{\tau - t_S}{\alpha}} - be^{\frac{T-t_S}{\alpha}}\right]\right)$.

Case (c): $\left(\int_0^\tau r(t)dt, r^*(\tau)\left[1 - e^{-b(T-\tau)}\right] + k(\tau)\left[be^{\frac{\tau - t_S}{\alpha}} - b\right]\right)$.

Case (d): $\left(\int_0^\tau r(t)dt, r^*(\tau)\right)$.

It is obvious from the preceding section that the random variables $\int_0^\tau r(t)dt$, $r^*(\tau)$ and $k(\tau)$ have normal distributions and that the latter bivariate random vectors all have bivariate normal distributions which are completely characterized in terms of their respective marginal expectations, variances and covariances. Exact expressions for these magnitudes are given in the technical appendix.

From here, we are able to find the joint distribution of $\int_0^\tau r(t)dt$ and $P(\tau, T)$ and thus, ultimately, to calculate the expectation on the right-hand side of Eq. (2.10). We defer the technical details of this procedure to the appendix. The following Proposition 3.6 summarizes the results by stating price equations for a European call option on zero-coupon bonds in the run-up to a currency union. In these case-specific option-pricing formulas we introduce some new notation. $\Phi(\cdot)$ denotes the standard normal cumulative distribution function, while $\Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S)$ is a case-specific parameter-dependent function, the intricate structural form of which is given in the Eqs. (A.2.8) to (A.2.10) of the appendix. Moreover, the pricing formulas contain the parameters $\mu_1, \mu_2, \sigma_1, \sigma_2$ and ρ which have not yet been defined. As described in the equation blocks (A.2.2) to (A.2.5) of the appendix these case-specific auxiliary parameters are certain functions of previously defined parameters.

Proposition 2.3.6 *In the run-up to a currency union the current price $C(0)$ of a European call option on a zero-coupon bond maturing at time T with strike price K and exercise date τ is given as follows:*

Case (a): For $0 < t_A$ or for $0 \geq t_S$ the option price is given by

$$\begin{aligned} C(0) &= P(0,T) \cdot \Phi\left(\frac{y_0 - \left[\mu_2 - \rho\sigma_1\sigma_2 + \frac{\sigma_2^2}{b}(e^{-b(T-\tau)} - 1)\right]}{\sigma_2} \right) \\ &\quad - K \cdot P(0,\tau) \cdot \Phi\left(\frac{y_0 - (\mu_2 - \rho\sigma_1\sigma_2)}{\sigma_2} \right). \end{aligned}$$

Case (b): For $t_A \leq 0 < \tau < T < t_S$ the option price is given by

$$\begin{aligned} C(0) &= P(0,T) \cdot \Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S) \cdot \Phi\left(\frac{y_0 - (\mu_2 - \rho\sigma_1\sigma_2 - \frac{\sigma_2^2}{b})}{\sigma_2} \right) \\ &\quad - K \cdot P(0,\tau) \cdot \Phi\left(\frac{y_0 - (\mu_2 - \rho\sigma_1\sigma_2)}{\sigma_2} \right). \end{aligned}$$

Case (c): For $t_A \leq 0 < \tau < t_S \leq T$ the option price is given by

$$\begin{aligned} C(0) &= P(0,T) \cdot \Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S) \cdot \Phi\left(\frac{y_0 - (\mu_2 - \rho\sigma_1\sigma_2 - \frac{\sigma_2^2}{b})}{\sigma_2} \right) \\ &\quad - K \cdot P(0,\tau) \cdot \Phi\left(\frac{y_0 - (\mu_2 - \rho\sigma_1\sigma_2)}{\sigma_2} \right). \end{aligned}$$

Case (d): For $t_A \leq 0 < t_S \leq \tau < T$ the option price is given by

$$\begin{aligned} C(0) &= P(0,T) \cdot \Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S) \cdot \Phi\left(\frac{y_0 - \left[\mu_2 - \rho\sigma_1\sigma_2 + \frac{\sigma_2^2}{b}(e^{-b(T-\tau)} - 1)\right]}{\sigma_2} \right) \\ &\quad - K \cdot P(0,\tau) \cdot \Phi\left(\frac{y_0 - (\mu_2 - \rho\sigma_1\sigma_2)}{\sigma_2} \right). \end{aligned}$$

We end this section by remarking that the option-price dynamics presented in Case (a) of Proposition 2.3.6 coincides with a well-known bond-option formula that has been derived by several authors under the classical scenario in which no currency union is planned (see for example Björk, 2004, pp. 337, 338).

2.4 Simulation study

In this section we implement a Monte-Carlo simulation to assess the potentiality for option mispricing that might emerge from ignoring the specific exchange-rate and interest-rate dynamics during the run-up to a currency union. To this end, we assume that the currency union is announced at date t_A—implying that the option-price dynamics from Proposition 2.3.6 constitutes the 'correct' model—and simulate pricing paths of some zero-coupon bonds plus corresponding pricing paths of some bond options. We further suppose that, despite of the fact that the currency union has been announced, agents ignore the 'correct' option-price dynamics given by the Cases (b), (c), (d) of Proposition 2.3.6 and erroneously presume instead that the bond-option dynamics from Proposition 2.3.6(a) still continues to be in force after t_A. As a result, agents misprice newly issued options by using this wrong option-price dynamics.

Our simulation starts in $t = 0$ and ends in $t = 2$. The dates relevant to the currency union are chosen as $t_A = 0.5$ and $t_S = 1.5$ implying an interim period of one year. For every parameter constellation we run a Monte-Carlo simulation with 10000 iterations and choose the distance between two points in time as 0.01. We set the mean-reversion level in the foreign short-rate process (2.6) to $c = 0.05$ and specify the irreversible exchange-rate fixing level as $s = \ln(1.00) = 0$. Following the line of argument in Wilfling (2003), we choose $\alpha = 2$ and, to simplify numerical procedures, set $\beta = 0$ implying that all $\Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S)$ function-values in Proposition 2.3.6 take on the constant value 1. For the parameters b, σ and $\tilde{\sigma}$ we choose the alternative setups shown in Table 2.2.

Based on these parameter constellations, we first simulate paths of the short rate $r(t)$ according to the dynamics given in Eq. (2.8). In a second step, we calculate five zero-coupon bond prices for the respective maturities of 1, 3, 6, 12, 24 months. Using these bond prices and the option-valuation formulas from Proposition 2.3.6, we then price six distinct options with strike prices K, option maturities τ and bond maturities T as shown in Table 2.1. It should be noted that these eleven bond and option prices represent arbitrage-free market prices.

Table 2.1 about here

In a next step, we price six newly issued zero-coupon bond options with strike prices $K \in \{0.915, 0.920, \dots, 0.940\}$, option maturity $\tau = 2$ months and bond maturity $T = 14$ months according to Proposition 2.3.6 at every of our supporting points in time. In contrast to our 'correct' pricing scheme, agents price these six options according to their erroneous assessment of option-price dynamics described above. In particular, using the 11 arbitrage-free bond and option prices observable in the market, agents calibrate their misspecified short-rate model consisting of the parameters b, c, σ thus obtaining different prices for the six newly issued options.

Table 2.2 displays the differences in the option prices obtained from (a) our pricing scheme (correct price), and (b) the misspecified valuation scheme employed by the agents (wrong price). We computed two measures of deviation, namely the average percentage deviation defined as the arithmetic mean of the values '$100 \times$ (wrong price $-$ correct price) $\div$ wrong price' and the average absolute percentage deviation defined as the mean of '$100 \times |\text{wrong price} - \text{correct price}| \div$ wrong price'. Both measures were computed at the dates 3, 6, 9 months after the announcement date t_A.

Table 2.2 about here

Figure 2.1 about here

Table 2.2 reveals that both deviation measures exhibit (*ceteris paribus*) the tendency to increase as the strike price K increases. In particular, given the values of the parameters $b, \sigma, \tilde{\sigma}$ under the strike price $K = 0.940$, we observe substantial deviations of more than 61 per cent. To gain deeper insight into the nature of such deviations, Figure 2.1 plots the paths of average percentage deviations generated from the 10000 replications in our simulation study using the parameter values $b = 1, \sigma = 0.01, \tilde{\sigma} = 0.05$ and the distinct strike prices $K \in \{0.915, 0.920, 0.925, 0.930\}$. For comparative reasons, we have chosen a common range of the deviations along the vertical axis, thus truncating many deviation paths in the lower panels. In accordance with Case (a) of Proposition 2.3.6, all deviations are equal to zero before t_A and after t_S simply reflecting the fact that no mispricing occurs before the announcement of the currency union and after the union has been implemented. In all of the four panels, however, two striking features of the deviation dynamics during the interim period between t_A and t_S become apparent. (a) Deviations tend to exhibit a heteroskedastic fluctuation pattern over time. (b) During the first half of the interim period most deviations are positive, while we find more negative than positive deviations during the second half.

Figure 2.2 about here

To characterize the distribution of the pricing error, we have fitted kernel densities to the deviations measured at some specifically chosen points in time. Figure 2.2 displays the kernel densities obtained under the parameters $b = 1, \sigma = 0.01, \tilde{\sigma} = 0.05$ and strike prices $K = 0.915, 0.920$ at the dates $t_1 = 0.75$ (3 months after t_A), $t_2 = 1.0$ (6 months after t_A) and $t_3 = 1.25$ (9 months after t_A). Obviously, the kernel density at t_1 exhibits more mass at positive deviations while the reverse holds for the densities at t_2 and t_3. Moreover, higher strike prices appear to be associated with more leptokurtic error distributions.

2.5 Conclusions

Based on a continuous-time modeling framework characterizing the dynamic link between international interest rates in the run-up to a currency union, this thesis derives closed-form valuation formulas for European call options on zero-coupon bonds. Taking into account the specific interest-rate dynamics induced by the switch in the exchange-rate regime, we extend the classical option-pricing framework and obtain novel pricing formulas. As the key result of our simulation study we find that disregarding the specific dynamic link between international interest rates prior to the currency union can generate substantial option-pricing errors.

It is obvious that our option-valuation formula may be used to price more complex contingent claims. As an example, we could consider interest-rate floors which can be viewed as a portfolio of European call options on zero-coupon bonds. Interest-rate floors typically are among the most traded of all interest-rate derivatives so that our results should be of high value for traders in all sorts of financial and derivative markets located in the upcoming EMU accession countries. It is worth noting, however, that our option-price dynamics is not confined to the episode of a future entrance into a currency union. In fact, it is also applicable to comparable transitional periods in the international monetary system such as the run-up to an exchange-rate peg or the implementation of a currency board.

The exact forms of our option-pricing formulas crucially hinge on two of our specifications chosen in Section 2.2, namely (a) the Vasicek-dynamics of the foreign short rate $r^*(t)$ in Eq. (2.6), and (b) the driftless Brownian-motion specification of the exchange-rate fundamental $k(t)$ in Eq. (2.2). Clearly, alternative specifications are conceivable for both variables such as the classical short-rate models proposed by Cox et. al. (1985), Ho and Lee (1986) or Hull and White (1994) for $r^*(t)$.

In this context it should be recalled that the specification of the exchange-rate fundamental $k(t)$ is of particular importance, since the k-dynamics characterizes the monetary policy regime during the run-up to the currency union. As described in Section 2.2, our (driftless) Brownian-motion specification represents a free-float exchange-rate regime between the countries involved. However, more interventionist exchange-rate policy stances prior to the currency union are conceivable and have indeed been pursued by some countries during the run-up to EMU (see Sondermann et al., 2010). Such active policy regimes can be modelled by Ornstein-Uhlenbeck and Brownian-bridge specifications for $k(t)$ (cf. Footnote 1) and one possible line of future research could be the investigation of how these alternative specifications affect our option-valuation dynamics derived in Proposition 2.3.6.

Appendix

To obtain the price dynamics of a European call option presented in Proposition 2.3.6, we have to calculate the expectation given on the right-hand side of Eq. (2.10):

$$E^Q\left[e^{-\int_0^\tau r(t)dt} \cdot \max\{P(\tau,T)-K,0\}|\phi_0\right]. \tag{A.2.1}$$

To this end, we follow the line of argument in Section 2.3.2 and consider the four distinct Cases (a) to (d) along with the corresponding bivariate probability distributions

$$\left(\int_0^\tau r(t)dt, r^*(\tau)\right), \qquad \text{[Case (a)]}$$

$$\left(\int_0^\tau r(t)dt, r^*(\tau)\left[1-e^{-b(T-\tau)}\right]+k(\tau)\left[be^{\frac{\tau-t_S}{\alpha}}-be^{\frac{T-t_S}{\alpha}}\right]\right), \qquad \text{[Case (b)]}$$

$$\left(\int_0^\tau r(t)dt, r^*(\tau)\left[1-e^{-b(T-\tau)}\right]+k(\tau)\left[be^{\frac{\tau-t_S}{\alpha}}-b\right]\right), \qquad \text{[Case (c)]}$$

$$\left(\int_0^\tau r(t)dt, r^*(\tau)\right). \qquad \text{[Case (d)]}$$

For ease of notation let us denote the first marginal distribution of any arbitrary bivariate random vector by X with expectation $\mu_1 \equiv E(X)$ and variance $\sigma_2^2 \equiv \text{Var}(X)$ and the respective magnitudes of the second marginal distribution by $Y, \mu_2 \equiv E(Y)$ and $\sigma_2^2 \equiv \mathrm{Var}(Y)$. Furthermore, let us write the covariance of X and Y as $\mathrm{Cov}(X,Y) = \rho\sigma_1\sigma_2$. It is straightforward to obtain all these magnitudes for the case-specific bivariate random vectors from above by standard means of probability calculus.

Case (a): For $(X,Y) = \left(\int_0^\tau r(t)dt, r^*(\tau)\right)$ we have

$$
\begin{aligned}
\mu_1 &= c\tau + \frac{r_0^* - c}{b}\left(1 - e^{-b\tau}\right), \\
\sigma_1^2 &= \frac{\sigma^2}{2b^3}(2b\tau - 3 + 4e^{-b\tau} - e^{-2b\tau}), \\
\mu_2 &= (r_0^* - c)e^{-b\tau} + c, \\
\sigma_2^2 &= \frac{\sigma^2}{2b}\left(1 - e^{-2b\tau}\right), \\
\rho\sigma_1\sigma_2 &= \frac{\sigma^2}{2b^2}\left(1 - e^{-b\tau}\right)^2.
\end{aligned}
\tag{A.2.2}
$$

Case (b): For $(X,Y) = \left(\int_0^\tau r(t)dt, r^*(\tau)\left[1-e^{-b(T-\tau)}\right] + k(\tau)\left[be^{\frac{\tau-t_S}{\alpha}} - be^{\frac{T-t_S}{\alpha}}\right]\right)$ we have

$$
\begin{aligned}
\mu_1 &= c\tau + \frac{r_0^* - c}{b}\left(1-e^{-b\tau}\right) + (s-k_0)\left(e^{\frac{\tau-t_S}{\alpha}} - e^{\frac{-t_S}{\alpha}}\right), \\
\sigma_1^2 &= \frac{\sigma^2}{2b^3}\left(2b\tau - 3 + 4e^{-b\tau} - e^{-2b\tau}\right) + \frac{\tilde{\sigma}^2\alpha}{2}e^{\frac{2\tau-2t_S}{\alpha}}\left(2\frac{\tau}{\alpha} - 3 + 4e^{-\frac{\tau}{\alpha}} - e^{-2\frac{\tau}{\alpha}}\right) \\
&\quad - 2\beta\frac{\sigma\tilde{\sigma}}{\alpha\sqrt{2b}}e^{-\frac{t_S}{\alpha}}\int_0^\tau\int_0^\tau e^{\frac{v}{\alpha}}e^{-bu}\min\left\{e^{2bu}-1, v\right\}dvdu, \\
\mu_2 &= \left(1-e^{-b(T-\tau)}\right)\left[(r_0^*-c)e^{-b\tau} + c\right] + k_0\left[be^{\frac{\tau-t_S}{\alpha}} - be^{\frac{T-t_S}{\alpha}}\right], \qquad \text{(A.2.3)} \\
\sigma_2^2 &= \left(1-e^{-b(T-\tau)}\right)^2\frac{\sigma^2}{2b}\left(1-e^{-2b\tau}\right) + \left(be^{\frac{\tau-t_S}{\alpha}} - be^{\frac{T-t_S}{\alpha}}\right)^2\tilde{\sigma}^2\tau \\
&\quad + 2\beta\frac{\tilde{\sigma}\sigma}{\sqrt{2b}}\left(1-e^{-b(T-\tau)}\right)\left(be^{\frac{\tau-t_S}{\alpha}} - be^{\frac{T-t_S}{\alpha}}\right)e^{-b\tau}\min\left\{e^{2b\tau}-1,\tau\right\}, \\
\rho\sigma_1\sigma_2 &= \frac{\sigma^2}{2b^2}\left(1-e^{-b\tau}\right)^2\left(1-e^{-b(T-\tau)}\right) + b\alpha\tilde{\sigma}^2e^{\frac{\tau-2t_S}{\alpha}}\left(e^{\frac{T-\tau}{\alpha}} - 1 - \frac{\tau}{\alpha}e^{\frac{\tau}{\alpha}} + e^{\frac{\tau}{\alpha}} + \frac{\tau}{\alpha}e^{\frac{T}{\alpha}} - e^{\frac{T}{\alpha}}\right) \\
&\quad + \left(be^{\frac{\tau-t_S}{\alpha}} - be^{\frac{T-t_S}{\alpha}}\right)\frac{\tilde{\sigma}\sigma}{\sqrt{2b}}\beta\int_0^\tau e^{-bu}\min\left\{e^{2bu}-1,\tau\right\}du \\
&\quad + \left(e^{-bT} - e^{-b\tau}\right)\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}\beta\int_0^\tau e^{\frac{u-t_S}{\alpha}}\min\left\{e^{2b\tau}-1,u\right\}du.
\end{aligned}
$$

Case (c): For $(X,Y) = \left(\int_0^\tau r(t)dt, r^*(\tau)\left[1-e^{-b(T-\tau)}\right] + k(\tau)\left[be^{\frac{\tau-t_S}{\alpha}} - b\right]\right)$ we have

$$
\begin{aligned}
\mu_1 &= c\tau + \frac{r_0^* - c}{b}\left(1-e^{-b\tau}\right) + (s-k_0)\left(e^{\frac{\tau-t_S}{\alpha}} - e^{\frac{-t_S}{\alpha}}\right), \\
\sigma_1^2 &= \frac{\sigma^2}{2b^3}\left(2b\tau - 3 + 4e^{-b\tau} - e^{-2b\tau}\right) + \frac{\tilde{\sigma}^2\alpha}{2}e^{\frac{2\tau-2t_S}{\alpha}}\left(2\frac{\tau}{\alpha} - 3 + 4e^{-\frac{\tau}{\alpha}} - e^{-2\frac{\tau}{\alpha}}\right) \\
&\quad - 2\beta\frac{\sigma\tilde{\sigma}}{\alpha\sqrt{2b}}e^{-\frac{t_S}{\alpha}}\int_0^\tau\int_0^\tau e^{\frac{v}{\alpha}}e^{-bu}\min\left\{e^{2bu}-1, v\right\}dudv, \\
\mu_2 &= \left(1-e^{-b(T-\tau)}\right)\left[(r_0^*-c)e^{-b\tau} + c\right] + k_0\left[be^{\frac{\tau-t_S}{\alpha}} - b\right], \qquad \text{(A.2.4)} \\
\sigma_2^2 &= \left(1-e^{-b(T-\tau)}\right)^2\frac{\sigma^2}{2b}\left(1-e^{-2b\tau}\right) + \left(be^{\frac{\tau-t_S}{\alpha}} - b\right)^2\tilde{\sigma}^2\tau \\
&\quad + 2\beta\frac{\tilde{\sigma}\sigma}{\sqrt{2b}}\left(1-e^{-b(T-\tau)}\right)\left(be^{\frac{\tau-t_S}{\alpha}} - b\right)e^{-b\tau}\min\left\{e^{2b\tau}-1,\tau\right\},
\end{aligned}
$$

$$\rho\sigma_1\sigma_2 = \frac{\sigma^2}{2b^2}\left(1-e^{-b\tau}\right)^2\left(1-e^{-b(T-\tau)}\right) - b\alpha\tilde{\sigma}^2 e^{\frac{\tau-t_S}{\alpha}}\left(1-\frac{\tau}{\alpha}-e^{-\frac{\tau}{\alpha}}+\frac{\tau}{\alpha}e^{\frac{\tau-t_S}{\alpha}}+e^{-\frac{t_S}{\alpha}}-e^{\frac{\tau-t_S}{\alpha}}\right)$$
$$+\left(be^{\frac{\tau-t_S}{\alpha}}-b\right)\frac{\tilde{\sigma}\sigma}{\sqrt{2b}}\beta\int_0^\tau e^{-bu}\min\left\{e^{2bu}-1,\tau\right\}du$$
$$+\left(e^{-bT}-e^{-b\tau}\right)\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}\beta\int_0^\tau e^{\frac{u-t_S}{\alpha}}\min\left\{e^{2b\tau}-1,u\right\}du.$$

Case (d): For $(X,Y)=\left(\int_0^\tau r(t)dt, r^*(\tau)\right)$ we have

$$\mu_1 = c\tau+\frac{r_0^*-c}{b}\left(1-e^{-b\tau}\right)+(s-k_0)\left(1-e^{\frac{-t_S}{\alpha}}\right),$$
$$\sigma_1^2 = \frac{\sigma^2}{2b^3}\left(2b\tau-3+4e^{-b\tau}-e^{-2b\tau}\right)+\frac{\tilde{\sigma}^2\alpha}{2}\left(2\frac{t_S}{\alpha}-3+4e^{-\frac{t_S}{\alpha}}-e^{-2\frac{t_S}{\alpha}}\right)$$
$$-2\beta\frac{\sigma\tilde{\sigma}}{\alpha\sqrt{2b}}e^{-\frac{t_S}{\alpha}}\int_0^{t_S}\int_0^\tau e^{\frac{v}{\alpha}}e^{-bu}\min\left\{e^{2bu}-1,v\right\}dudv,$$
$$\mu_2 = (r_0^*-c)e^{-b\tau}+c, \tag{A.2.5}$$
$$\sigma_2^2 = \frac{\sigma^2}{2b}\left(1-e^{-2b\tau}\right),$$
$$\rho\sigma_1\sigma_2 = \frac{\sigma^2}{2b^2}\left(1-e^{-b\tau}\right)^2-\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}}e^{-b\tau}\beta\int_0^{t_S}e^{\frac{u-t_S}{\alpha}}\min\left\{e^{2b\tau}-1,u\right\}du.$$

Using this case-specific notation and noting that the bond price $P(\tau,T)$ is a function of the second marginal distribution Y, which justifies our expressing the bond price as $P(\tau,T,Y)$, we may rewrite the expectation (A.2.1) as

$$E^Q\left[e^{-X}\cdot\max\{P(\tau,T,Y)-K,0\}|\phi_0\right]$$
$$=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}e^{-x}\max\{P(\tau,T,y)-K,0\}f(x,y)dxdy, \tag{A.2.6}$$

where $f(x,y)$ represents the joint probability density function of (X,Y). The integrand of the double integral in Eq. (A.2.6) contains a maximum function. However, it is easy to check that the term $P(\tau,T,y)-K$ is monotone decreasing in y so that we can simplify the double integral by (a) calculating those case-specific values y_0 for which this term becomes zero, and (b) by changing the integration limits accordingly. These y_0-values are given as follows:

Case (a):

$$y_0 = \frac{-b\ln(K) - bc(T-\theta) + c\left(1 - e^{-b(T-\theta)}\right) + \frac{\sigma^2}{4b^2}\left[2b(T-\theta) - 3 + 4e^{-b(T-\theta)} - e^{-2b(T-\theta)}\right]}{1 - e^{-b(T-\theta)}}.$$

Case (b):

$$\begin{aligned} y_0 = & -b\ln(K) - bc(T-\theta) + c\left(1 - e^{-b(T-\theta)}\right) + \frac{\sigma^2}{4b^2}\left[2b(T-\theta) - 3 + 4e^{-b(T-\theta)} - e^{-2b(T-\theta)}\right] \\ & + \frac{b\tilde{\sigma}^2\alpha}{4} e^{\frac{2T-2t_S}{\alpha}}\left(2\frac{T-\theta}{\alpha} - 3 - e^{-\frac{2T-2\theta}{\alpha}} + 4e^{-\frac{T-\theta}{\alpha}}\right) - bs\left(e^{\frac{T-t_S}{\alpha}} - e^{\frac{\theta-t_S}{\alpha}}\right) \\ & - b\beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} e^{\frac{\theta-t_S}{\alpha}} \int_0^{T-\theta}\int_0^{T-\theta} e^{\frac{v}{\alpha}} e^{-bu} \min\left\{e^{2bu} - 1, v\right\} du dv. \end{aligned}$$

Case (c):

$$\begin{aligned} y_0 = & -b\ln(K) - bc(T-\theta) + c\left(1 - e^{-b(T-\theta)}\right) + \frac{\sigma^2}{4b^2}\left[2b(T-\theta) - 3 + 4e^{-b(T-\theta)} - e^{-2b(T-\theta)}\right] \\ & + \frac{b\tilde{\sigma}^2\alpha}{4}\left(2\frac{t_S-\theta}{\alpha} - 3 - e^{-\frac{2t_S-2\theta}{\alpha}} + 4e^{-\frac{t_S-\theta}{\alpha}}\right) - bs\left(1 - e^{\frac{\theta-t_S}{\alpha}}\right) \\ & - b\beta\frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} e^{\frac{\theta-t_S}{\alpha}} \int_0^{t_S-\theta}\int_0^{T-\theta} e^{\frac{v}{\alpha}} e^{-bu} \min\left\{e^{2bu} - 1, v\right\} du dv. \end{aligned}$$

Case (d):

$$y_0 = \frac{-b\ln(K) - bc(T-\theta) + c\left(1 - e^{-b(T-\theta)}\right) + \frac{\sigma^2}{4b^2}\left[2b(T-\theta) - 3 + 4e^{-b(T-\theta)} - e^{-2b(T-\theta)}\right]}{1 - e^{-b(T-\theta)}}.$$

Using these case-specific y_0-values, we can remove the maximum function and simplify the double integral in Eq. (A.2.6) to give

$$\int_{-\infty}^{y_0}\int_{-\infty}^{\infty} e^{-x}\left[P(\tau, T, y) - K\right] f(x, y) dx dy.$$

Inserting the explicit form of the bivariate normal probability density function for $f(x, y)$ and performing some straightforward manipulations, we obtain

$$\int_{-\infty}^{y_0} [P(\tau, T, y) - K] \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(y-\mu_2)^2}{2\sigma_2^2}} \int_{-\infty}^{\infty} e^{-x} \frac{1}{\sqrt{2\pi}\sigma_1\sqrt{(1-\rho^2)}} e^{-\frac{1}{2}\frac{(x-(\mu_1+\rho y\frac{\sigma_1}{\sigma_2}-\rho\mu_2\frac{\sigma_1}{\sigma_2}))^2}{(1-\rho^2)\sigma_1^2}} dxdy.$$

The second integral in the latter term constitutes the expected value of a lognormal distribution yielding the equivalent expression

$$\int_{-\infty}^{y_0} [P(\tau, T, y) - K] \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(y-\mu_2)^2}{2\sigma_2^2}} e^{-(\mu_1+\rho y\frac{\sigma_1}{\sigma_2}-\rho\mu_2\frac{\sigma_1}{\sigma_2})+\frac{(1-\rho^2)\sigma_1^2}{2}} dy,$$

which can be expanded to give

$$\begin{aligned}&\int_{-\infty}^{y_0} P(\tau, T, y) \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(y-\mu_2)^2}{2\sigma_2^2}} e^{-(\mu_1+\rho y\frac{\sigma_1}{\sigma_2}-\rho\mu_2\frac{\sigma_1}{\sigma_2})+\frac{(1-\rho^2)\sigma_1^2}{2}} dy \\ &- \int_{-\infty}^{y_0} K \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(y-\mu_2)^2}{2\sigma_2^2}} e^{-(\mu_1+\rho y\frac{\sigma_1}{\sigma_2}-\rho\mu_2\frac{\sigma_1}{\sigma_2})+\frac{(1-\rho^2)\sigma_1^2}{2}} dy.\end{aligned}$$

The second integral in this last term can be expressed in terms of the standard normal cumulative distribution function (cdf) $\Phi(\cdot)$, yielding

$$\begin{aligned}&\int_{-\infty}^{y_0} P(\tau, T, y) \frac{1}{\sqrt{2\pi}\sigma_2} e^{-\frac{(y-\mu_2)^2}{2\sigma_2^2}} e^{-(\mu_1+\rho y\frac{\sigma_1}{\sigma_2}-\rho\mu_2\frac{\sigma_1}{\sigma_2})+\frac{(1-\rho^2)\sigma_1^2}{2}} dy \\ &- K \cdot P(0, \tau) \cdot \Phi\left(\frac{y_0 - (\mu_2 - \rho\sigma_1\sigma_2)}{\sigma_2}\right). \qquad \text{(A.2.7)}\end{aligned}$$

Note that $P(0, \tau)$ in Eq. (A.2.7) is the price of a zero-coupon bond maturing at time τ, the (case-specific) form of which is established in Proposition 2.3.4.

Finally, substituting the bond price $P(\tau, T, y)$ by its explicit formulas for the four distinct cases (a) to (d) and applying analogous steps as before, we are able to write the remaining integral in the expression (A.2.7) in terms of the standard normal cumulative distribution function. More precisely, the integral can be expressed as the product of the following three factors: (a) the bond price $P(0, T)$, (b) an auxiliary function $\Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S)$, and (c) a specific value of the cdf $\Phi(\cdot)$. The exact forms of the product are given in the respective first lines of the option-valuation formulas in Proposition 2.3.6 (cases (a) to (d)). The precise form of the auxiliary function Γ, which can only be different from 1 for the cases (b) to (d), are given as follows:

Case (b): $\Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S) =$

$$\exp\Big\{ - \beta e^{-\frac{t_S}{\alpha}} \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} \int_0^{T-\tau} \int_0^{T-\tau} e^{\frac{v+\tau}{\alpha}} e^{-b(u+\tau)} \min\left\{e^{2bu} - 1, v\right\} dudv$$
$$- \beta e^{-\frac{t_S}{\alpha}} \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} \int_0^{T} \int_0^{T} e^{\frac{v}{\alpha}} e^{-bu} \Big(\min\left\{e^{2bu} - 1, v, e^{2b\tau} - 1, \tau\right\}$$
$$- \min\left\{e^{2bu} - 1, v\right\} \Big) dudv \Big\}. \qquad \text{(A.2.8)}$$

Case (c): $\Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S) =$

$$\exp\Big\{ - \beta e^{-\frac{t_S}{\alpha}} \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} \int_0^{t_S-\tau} \int_0^{T-\tau} e^{\frac{v+\tau}{\alpha}} e^{-b(u+\tau)} \min\left\{e^{2bu} - 1, v\right\} dudv$$
$$- \beta e^{-\frac{t_S}{\alpha}} \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} \int_0^{t_S} \int_0^{T} e^{\frac{v}{\alpha}} e^{-bu} \Big(\min\left\{e^{2bu} - 1, v, e^{2b\tau} - 1, \tau\right\}$$
$$- \min\left\{e^{2bu} - 1, v\right\} \Big) dudv \Big\}. \qquad \text{(A.2.9)}$$

Case (d): $\Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S) =$

$$\exp\Big\{ \beta e^{-\frac{t_S}{\alpha}} \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} \int_0^{t_S} \int_\tau^{T} e^{\frac{v}{\alpha}} e^{-bu} \min\left\{e^{2bu} - 1, v\right\} dudv$$
$$- \beta e^{-\frac{t_S}{\alpha}} \frac{\tilde{\sigma}\sigma}{\alpha\sqrt{2b}} \int_0^{t_S} \frac{1}{b} e^{\frac{v}{\alpha}} e^{-b\tau} \min\left\{e^{2b\tau} - 1, v\right\} dv \Big\}. \qquad \text{(A.2.10)}$$

In each of the three cases (b) to (d), the function $\Gamma(b, \alpha, \beta, \sigma, \tilde{\sigma}, \tau, T, t_S)$ depends *inter alia* on the parameter β which represents the (constant) correlation coefficient of the Wiener processes $W_1(t)$ and $\bar{W}(t)$ driving the foreign short rate $r^*(t)$ and the instantaneous short-rate differential $\text{SRD}(t) = r(t) - r^*(t)$, respectively (cf. Section 2.2). If these Wiener processes are uncorrelated, i.e. $\beta = 0$, the Γ-function takes on the value 1, which considerably simplifies our option-valuation formulas in Proposition 3.6.

Figures

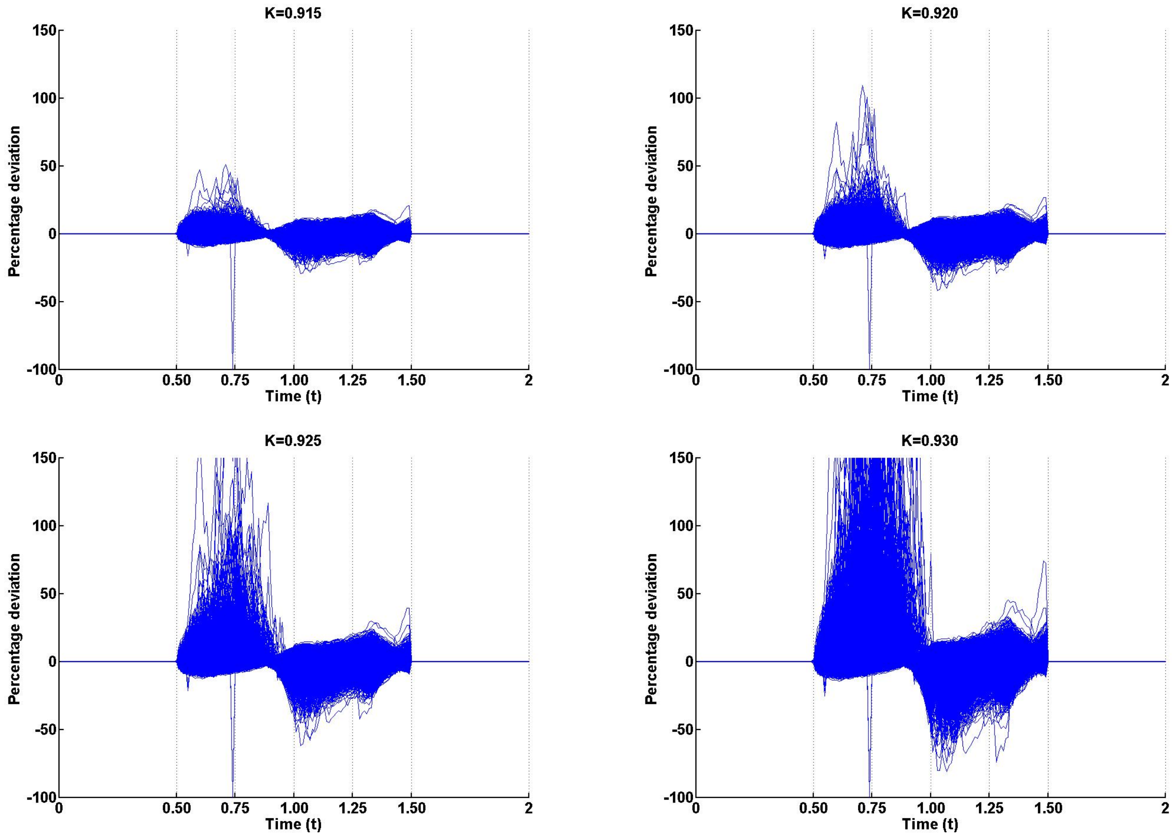

Figure 2.1: Average percentage deviations under the parameters $b = 1, \sigma = 0.01, \tilde{\sigma} = 0.05$ for alternative strike prices K

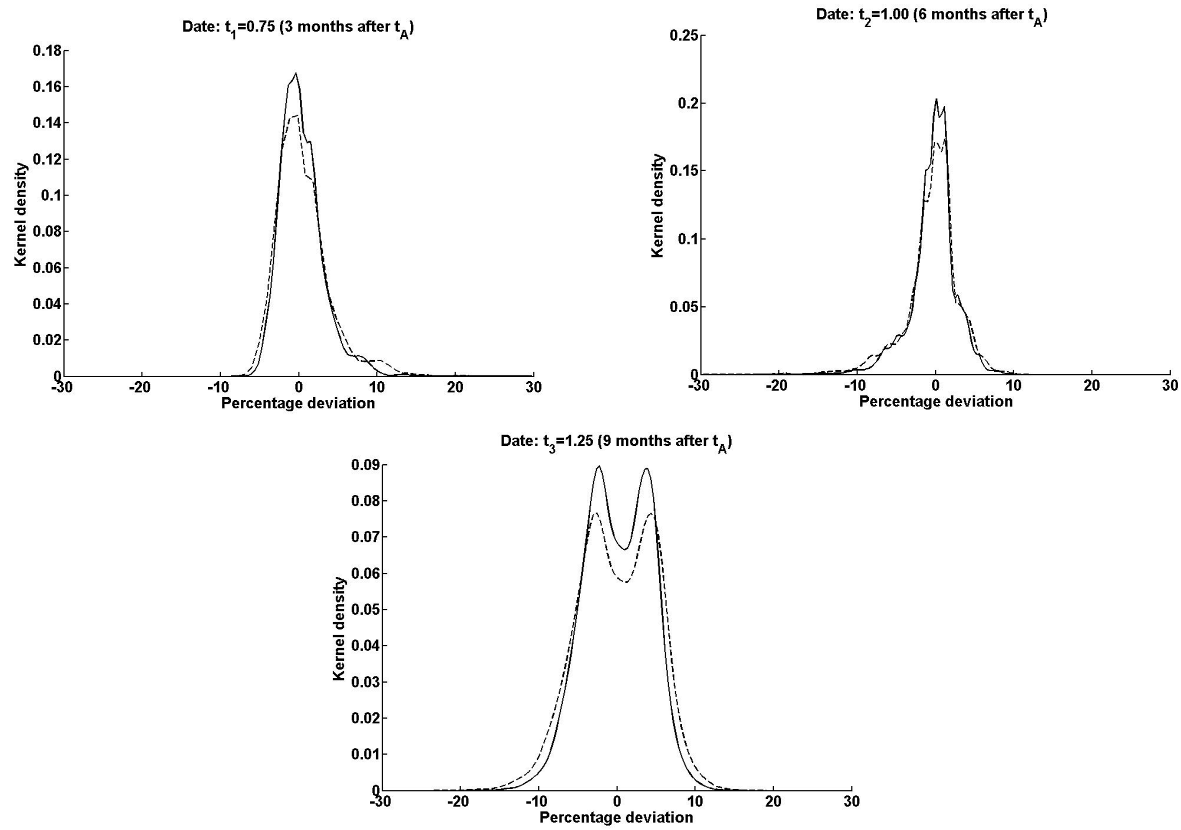

Figure 2.2: Kernel densities with $b = 1, \sigma = 0.01, \tilde{\sigma} = 0.05, K = 0.915$ (solid lines), $K = 0.92$ (dashed lines) for alternative points in time after t_A

Tables

τ (in months)	T (in months)	K
1	3	0.99
1	6	0.98
1	12	0.96
3	6	0.99
3	12	0.97
6	12	0.98

Table 2.1: Parameters of options used for yield inversion

		$K = 0.915$		$K = 0.920$	
Time after t_S (in months)	Parameter setup	Average perc. dev.	Average abs. perc. dev.	Average perc. dev.	Average abs. perc. dev.
3	$b = 1$,	0.413	1.928	0.720	2.442
6	$\sigma = 0.015$,	−0.262	2.025	−0.383	2.384
9	$\tilde{\sigma} = 0.050$	−0.037	3.584	−0.073	4.208
3	$b = 1$,	0.072	0.950	0.109	1.119
6	$\sigma = 0.015$,	−0.087	0.962	−0.113	1.118
9	$\tilde{\sigma} = 0.025$	0.146	2.149	0.169	2.519
3	$b = 1$,	0.374	2.094	0.618	2.569
6	$\sigma = 0.010$,	−0.309	2.055	−0.453	2.432
9	$\tilde{\sigma} = 0.050$	0.035	3.489	0.007	4.077
3	$b = 1$,	0.086	1.005	0.122	1.176
6	$\sigma = 0.010$,	−0.130	0.916	−0.170	1.070
9	$\tilde{\sigma} = 0.025$	0.139	2.091	0.155	2.439
3	$b = 2$,	0.596	2.319	0.839	2.793
6	$\sigma = 0.015$,	−0.339	1.464	−0.474	1.747
9	$\tilde{\sigma} = 0.050$	−0.159	3.049	−0.219	3.558
3	$b = 2$,	0.078	1.124	0.114	1.311
6	$\sigma = 0.015$,	−0.047	0.544	−0.068	0.636
9	$\tilde{\sigma} = 0.025$	0.027	1.529	0.024	1.781

Note: The deviation measures 'Average percentage deviation' and 'Average absolute percentage deviation' are defined as the arithmetic means of the values '$100 \times (\text{wrong price} - \text{correct price}) \div \text{wrong price}$' and '$100 \times |\text{wrong price} - \text{correct price}| \div \text{wrong price}$', respectively.

Table 2.2: Deviations of option prices for alternative parameters when valuing (a) under the 'correct' and (b) under the 'wrong' bond-option dynamics

Table 2.2 (continued)

Time after t_S (in months)	Parameter setup	$K = 0.925$ Average perc. dev.	Average abs. perc. dev.	$K = 0.930$ Average perc. dev.	Average abs. perc. dev.
3	$b = 1$,	1.488	3.487	3.697	6.102
6	$\sigma = 0.015$,	−0.515	2.861	−0.434	3.654
9	$\tilde{\sigma} = 0.050$	−0.142	5.110	−0.295	6.562
3	$b = 1$,	0.197	1.388	0.475	1.933
6	$\sigma = 0.015$,	−0.120	1.335	0.043	1.785
9	$\tilde{\sigma} = 0.025$	0.203	3.050	0.257	3.905
3	$b = 1$,	1.200	3.455	2.833	5.509
6	$\sigma = 0.010$,	−0.681	2.966	−0.948	3.701
9	$\tilde{\sigma} = 0.050$	−0.051	4.907	−0.179	6.173
3	$b = 1$,	0.191	1.429	0.359	1.861
6	$\sigma = 0.010$,	−0.226	1.282	−0.288	1.591
9	$\tilde{\sigma} = 0.025$	0.174	2.928	0.194	3.669
3	$b = 2$,	1.342	3.598	2.634	5.307
6	$\sigma = 0.015$,	−0.710	2.177	−1.151	2.881
9	$\tilde{\sigma} = 0.050$	−0.319	4.271	−0.507	5.341
3	$b = 2$,	0.176	1.576	0.303	1.991
6	$\sigma = 0.015$,	−0.102	0.767	−0.169	0.968
9	$\tilde{\sigma} = 0.025$	0.017	2.131	−0.001	2.653

Note: The deviation measures 'Average percentage deviation' and 'Average absolute percentage deviation' are defined as the arithmetic means of the values '$100\times$(wrong price$-$correct price)$\div$wrong price' and '$100 \times |$wrong price $-$ correct price$| \div$ wrong price', respectively.

Table 2.2 (continued)

		$K = 0.935$		$K = 0.940$	
Time after t_S (in months)	Parameter setup	Average perc. dev.	Average abs. perc. dev.	Average perc. dev.	Average abs. perc. dev.
3	$b = 1$,	11.555	14.751	56.106	61.371
6	$\sigma = 0.015$,	0.977	5.827	10.314	16.466
9	$\tilde{\sigma} = 0.050$	−0.702	9.461	−1.354	17.122
3	$b = 1$,	1.667	3.613	9.590	12.978
6	$\sigma = 0.015$,	1.232	3.298	9.891	12.432
9	$\tilde{\sigma} = 0.025$	0.297	5.741	0.571	11.791
3	$b = 1$,	8.332	11.645	34.768	39.374
6	$\sigma = 0.010$,	−0.636	4.871	4.693	10.522
9	$\tilde{\sigma} = 0.050$	−0.512	8.388	−1.506	13.541
3	$b = 1$,	0.973	2.887	4.985	7.726
6	$\sigma = 0.010$,	−0.165	2.186	2.634	5.110
9	$\tilde{\sigma} = 0.025$	0.206	4.957	0.234	8.139
3	$b = 2$,	6.956	10.234	30.862	35.236
6	$\sigma = 0.015$,	−1.979	4.092	−3.388	6.196
9	$\tilde{\sigma} = 0.050$	−0.916	7.144	−1.917	10.739
3	$b = 2$,	0.666	2.788	2.262	5.140
6	$\sigma = 0.015$,	−0.313	1.316	−0.518	1.946
9	$\tilde{\sigma} = 0.025$	−0.051	3.513	−0.191	5.210

Note: The deviation measures 'Average percentage deviation' and 'Average absolute percentage deviation' are defined as the arithmetic means of the values '$100 \times$(wrong price−correct price)$\div$wrong price' and '$100 \times |\text{wrong price} - \text{correct price}| \div$ wrong price', respectively.

Chapter 3

Markov-switching in conditional heteroskedasticity models: a unifying framework with an application to the German stock market

3.1 Introduction

Since the seminal papers of Engle (1982) and Bollerslev (1986) GARCH (generalized autoregressive conditional heteroskedasticity) models have become a standard tool in modeling the conditional variances of the returns from financial time series data. The popularity of these models stems from (1) their compatibility with major stylized facts for asset returns, (2) the existence of efficient statistical methods for estimating model parameters, and (3) the availability of useful volatility forecasts.

In order to cover specific volatility features like the well-known leverage effect and other asymmetries in financial returns (e.g. Black, 1976; Christie, 1982; Schwert, 1989), a plethora of GARCH specifications have been suggested in the literature among the most prominent being the exponential GARCH (EGARCH) model introduced by Nelson (1991) and the threshold GARCH (TGARCH) model of Zakoian (1994). However, Hentschel

(1995) establishes a connection between all these models by showing that all specifications are special cases of a Box-Cox (1964) transformation to the conditional standard deviation.

While all the above-mentioned single-regime GARCH specifications have been well-established from a statistical point of view and have become standard routines in many econometric software packages, their two-regime Markov-switching counterparts are less straightforward to implement. Apart from the (typically) large number of parameters that have to be estimated this lack may be due to a phenomenon known as *path dependence* which stems from the GARCH lag structure and causes the regime-specific conditional variance to depend on the entire history of the data in a Markov-switching GARCH model. As pointed out by Cai (1994) and Hamilton and Susmel (1994) path dependence typically entails severe estimation problems if not carefully handled. However, Gray (1996) establishes a path-independent Markov-switching GARCH framework that permits direct estimation of all model parameters using (quasi) maximum likelihood techniques. Gray's model was later refined by Klaassen (2002) and it is their Markov-switching framework that we will expand in this chapter.

Today, Markov-switching (or regime-switching) GARCH models, which are designed to capture discrete shifts in the volatility process of time series data, are in widespread use in various fields of financial economics. Most recent empirical applications of Markov-switching GARCH models to stock-return and exchange-rate return data are presented, *inter alia*, in Gelman and Wilfling (2009), Henry (2009), Bauwens et al. (2010) and Bohl et al. (2010). However, all two-regime Markov-switching GARCH specifications hitherto estimated in the economics literature have one feature in common that appears unnecessarily restrictive. Despite the fact that the parameters in the variance equations are allowed to switch across both regimes, the overall functional forms of the two regime-specific GARCH equations are modeled as identical. For example, apart from Henry (2009) all authors of the above-cited empirical applications specify two-regime Markov-switching models with standard GARCH equations in each Markov regime while Henry (2009) uses EGARCH specifications in both regimes.

In this chapter we develop a more flexible setup by incorporating Hentschel's (1995) results on the nesting of distinct symmetric and asymmetric single-regime GARCH models into Gray's (1996) Markov-switching GARCH model. In this way, we establish a general regime-switching framework that enables us to estimate complex GARCH equations of different functional forms across the Markov regimes. To give an example, our setup allows us to specify an EGARCH equation in regime 1 while regime 2 might be described by a conventional GARCH specification. To our best knowledge such a flexible Markov-switching GARCH framework has not yet been implemented in the economics literature. In the empirical part of the chapter we apply our general Markov-switching GARCH approach to the excess returns generated by the German stock index DAX and demonstrate that our flexible setup econometrically outperforms all Markov-switching GARCH models hitherto estimated in the financial literature.

The remainder of the chapter is organized as follows. Section 3.2 formally establishes our general Markov-switching GARCH framework. For ease of readability we derive the complete maximum likelihood estimation procedure in the technical appendix to the chapter. Section 3.3 describes the data set and presents the estimation results. The final Section 3.4 summarizes the main results and concludes the paper.

3.2 A general Markov-switching GARCH model

In this section we establish our general Markov-switching model that enables us to specify and estimate GARCH equations of different functional forms in each of the distinct Markov regimes. For this we assume that the data generating process (DGP) of the financial return r_t is affected by an unobserved latent random variable S_t representing the regime the DGP is in at time t. For simplicity we assume only the two distinct regimes 1 and 2 at any point in time, that is, we assume either $S_t = 1$ or $S_t = 2$ for all $t = 1, 2, \ldots$.

As a starting point of our derivation we will follow Hentschel's (1995) exposition and build up the so-called *Absolute Value GARCH* model for the return process $\{r_t\}$. However, we expand Hentschel's single-regime framework to a two-regime Markov-switching model. To this end we let the return dynamics depend on the regime indicator $S_t = i, i = 1, 2$ and specify

$$r_{t+1} = \lambda_i + \gamma_i \sqrt{h_{i,t}} + \sqrt{h_{i,t}} \epsilon_{t+1}. \tag{3.1}$$

In Eq. (3.1), λ_i and γ_i are regime-specific constants while $\{\epsilon_t\}$ denotes an i.i.d. process of standard normal variates. $h_{i,t}$ represents the conditional variance in regime i the modeling of which will be treated below. The term $\lambda_i + \gamma_i \sqrt{h_{i,t}}$ on the right-hand side of Eq. (3.1) constitutes the mean equation of the return in regime i and is known as the GARCH-in-Mean (GARCH-M) model suggested by Engle et al. (1987) which has been used in many empirical studies on the behavior of stock returns. Based on these assumptions, the conditional distribution of the return is a mixture of two normal distributions which can be written as

$$r_{t+1}|\phi_t \sim \begin{cases} N(\lambda_1 + \gamma_1 \sqrt{h_{1,t}}, h_{1,t}) & \text{with probability} \quad p_{1,t} \\ N(\lambda_2 + \gamma_2 \sqrt{h_{2,t}}, h_{2,t}) & \text{with probability} \quad (1 - p_{1,t}) \end{cases}. \tag{3.2}$$

In Eq. (3.2) ϕ_t defines the information set as of date t and $p_{1,t} \equiv \Pr\{S_t = 1|\phi_t\}$ denotes the so-called *ex-ante* probability of being in regime 1 at date t. It is instructive to note that the information set ϕ_t basically coincides with the return path $\tilde{r}_t = \{r_t, r_{t-1}, \ldots\}$, but does not contain the path of the unobservable regime indicator S_t.

In the modeling of our regime-specific GARCH equations, we follow the path-independent methodology developed in Gray (1996). In order to circumvent the aforementioned problem of path dependence, we specify the dynamics of the regime-specific conditional variance $h_{i,t}$ in terms of a lagged variance h_{t-1} and a shock term δ_t which are both appropriately weighted aggregates of the past conditional variances $h_{1,t-1}$ and $h_{2,t-1}$ from both Markov regimes. At this point we make use of an econometric improvement on Gray's approach suggested by Klaassen (2002). Klaassen's idea is to exploit all available information when integrating out the unobserved regimes in order to establish the aggregated variances and shock terms while Gray (1996) uses only part of it. To be more precise, in specifying the volatility $h_{i,t}$ valid in regime i, Klaassen computes the aggregated variance h_{t-1} and the shock terms δ_t on the basis of probabilities which explicitly take into account that we consider regime i at time t. This modeling improvement is particularly efficient when the Markov regimes appear to be highly persistent. In order to indicate the use of this additional information we denote the aggregated variance for date $t-1$ conditional on the fact that we are in regime i on date t by $h_{t-1}^{(i)}$, and accordingly the shock terms by $\delta_t^{(i)}$. In particular, we specify both quantities as

$$\begin{aligned} h_{t-1}^{(i)} &= p_{1,t-1}^{(i)} h_{1,t-1} + (1 - p_{1,t-1}^{(i)}) h_{2,t-1} \\ &\quad + p_{1,t-1}^{(i)} (1 - p_{1,t-1}^{(i)}) \left[\lambda_1 + \gamma_1 \sqrt{h_{1,t-1}} - (\lambda_2 + \gamma_2 \sqrt{h_{2,t-1}}) \right]^2 \end{aligned} \tag{3.3}$$

and

$$\delta_t^{(i)} = p_{1,t-1}^{(i)} \frac{r_t - (\lambda_1 + \gamma_1 \sqrt{h_{1,t-1}})}{\sqrt{h_{1,t-1}}} + (1 - p_{1,t-1}^{(i)}) \frac{r_t - (\lambda_2 + \gamma_2 \sqrt{h_{2,t-1}})}{\sqrt{h_{2,t-1}}}, \tag{3.4}$$

respectively, where the probabilities $p_{1,t-1}^{(i)}$ are calculated from Eq. (A.3.13) in the appendix.[1]

Based on the aggregated variance and shock terms $h_{t-1}^{(i)}$ and $\delta_t^{(i)}$ from the Eqs. (3.3) and (3.4), we now define a first preliminary two-regime conditional volatility equation as

$$\sqrt{h_{i,t}} = \sqrt{Var(r_{t+1} | \phi_t, S_t = i)} = \omega_i + \alpha_i \sqrt{h_{t-1}^{(i)}} |\delta_t^{(i)}| + \beta_i \sqrt{h_{t-1}^{(i)}}, \tag{3.5}$$

where ω_i, α_i and β_i denote regime-specific volatility parameters to be estimated from the data. It is obvious that the volatility equation (3.5) constitutes a conventional GARCH(1,1) model in which the conditional variance terms and the shock terms have been replaced by the conditional standard deviations and the absolute shock terms, respectively. However, an important drawback of this volatility equation is its incapability of capturing empirically well-documented asymmetries in the volatility of financial re-

[1] Instead of using the more informative Klaassen probabilities $p_{1,t-1}^{(i)}$, Gray (1996) uses the *ex-ante* probabilities $p_{1,t-1}$ from Eq. (3.2). This implies that Gray's aggregated variances and shock terms are equal irrespective of the Markov regime considered at date t.

turns. In order to resolve this deficit, we follow Hentschel (1995), who generalizes the volatility equation (3.5) in a single-regime framework, and specify the second version of our two-regime conditional volatility equation as

$$\sqrt{h_{i,t}} = \omega_i + \alpha_i \sqrt{h_{t-1}^{(i)}} f_i(\delta_t^{(i)}) + \beta_i \sqrt{h_{t-1}^{(i)}} \tag{3.6}$$

with

$$f_i(\delta_t^{(i)}) = |\delta_t^{(i)} - b_i| - c_i(\delta_t^{(i)} - b_i), \tag{3.7}$$

where b_i, c_i represent regime-specific parameters. In what follows, we refer to the Eqs. (3.6) and (3.7) as the *Absolute Value GARCH* (AVGARCH) model.

Although the AVGARCH specification is interesting in its own right, Hentschel (1995) demonstrates that a Box-Cox (1964) transformation of the conditional standard deviation in the Eqs. (3.6) and (3.7) produces a rich class of models that includes many well-known symmetric and asymmetric GARCH models as special cases. Adapting this approach to our two-regime Markov-switching framework by introducing the regime-specific parameters μ_i and ν_i, we transform our conditional volatility Eq. (3.6) to

$$\frac{\sqrt{h_{i,t}}^{\mu_i} - 1}{\mu_i} = \omega_i + \alpha_i \sqrt{h_{t-1}^{(i)}}^{\mu_i} f_i^{\nu_i}(\delta_t^{(i)}) + \beta_i \frac{\sqrt{h_{t-1}^{(i)}}^{\mu_i} - 1}{\mu_i}. \tag{3.8}$$

The parameter μ_i determines the shape of the Box-Cox transformation in regime i. For $0 \leq \mu_i \leq 1$ the transformation of the conditional standard deviation $\sqrt{h_{i,t}}$ is concave while it is convex for $\mu_i > 1$. The parameter ν_i transforms the regime-specific function $f_i(\cdot)$. For $0 < \nu_i < 1$ the function $f^{\nu_i}(\cdot)$ becomes concave on either side of b_i while it becomes convex for $\nu_i > 1$. A convenient choice of the parameter c_i on the right-hand side of Eq. (3.7) is $|c_i| \leq 1$ since this condition guarantees a positive value of $f^{\nu_i}(\delta_t^{(i)})$. However, $|c_i| \leq 1$ is neither a necessary nor a sufficient condition to ensure $\sqrt{h_{i,t}} \geq 0$. Table 3.1 compiled from Table 1 in Hentschel (1995, p. 79) reveals how our volatility Eq. (3.8) for regime i nests many GARCH models scattered in the literature by imposing appropriate restrictions on the parameters μ_i, ν_i, b_i and c_i.

Table 3.1 about here

Finally, we close our econometric model by specifying the probabilistic nature of the regime indicator S_t. In our study we let $\{S_t\}$ follow a two-state first-order Markov process with time-varying transition probabilities and write this as

$$\begin{aligned}
\Pr(S_t = 1 | S_{t-1} = 1, r_t) &= P_t, \\
\Pr(S_t = 2 | S_{t-1} = 1, r_t) &= 1 - P_t, \\
\Pr(S_t = 1 | S_{t-1} = 2, r_t) &= 1 - Q_t, \\
\Pr(S_t = 2 | S_{t-1} = 2, r_t) &= Q_t.
\end{aligned} \tag{3.9}$$

The probability of being in regime i for $i = 1, 2$ depends on realizations in $\tilde{r}_t$ and $\{S_t\}$ only through S_{t-1}. For the time-varying transition probabilities we assume

$$\begin{aligned}
P_t &= \Phi(d_1 + e_1 \cdot r_t), \\
Q_t &= \Phi(d_2 + e_2 \cdot r_t)
\end{aligned} \tag{3.10}$$

with $\Phi(\cdot)$ denoting the cumulative distribution function of a standard normal variate and d_1, d_2, e_1, e_2 representing parameters to be estimated from the data.

Our Markov-switching GARCH model established in the Eqs. (3.1) to (3.10) can now be estimated using (quasi) maximum likelihood techniques. The log-likelihood function is constructed recursively and we present its exact form in the Eqs. (A.3.1) to (A.3.14) of the appendix. In the next section we apply this general Markov-switching GARCH framework to the daily excess returns of the German stock index DAX.

3.3 Empirical application

3.3.1 Data

We now analyze the mean and volatility structure of the daily excess returns sampled from the German stock market between 3 January 2000 and 31 December 2009 (2554 observations). We construct the excess returns r_t by subtracting an appropriately defined risk-free interest rate from the returns of the German stock index DAX.[2] Our DAX returns used for calculating the excess returns are adjusted for dividend payments. As the risk-free interest rate we use the *Euro OverNight Index Average* EONIA which we convert into daily returns by dividing the given annualized EONIA rate by 250.[3]

Figure 3.1 about here

Figure 3.1 displays the German stock index DAX (left panel) and the corresponding DAX excess returns r_t (right panel) during the sampling period. The trajectory of the excess returns clearly exhibits the two most prominent features well-documented in the financial literature on asset-return dynamics, namely volatility clustering and a time-varying mean. We now turn to analyzing these dynamic structures within our Markov-switching GARCH framework developed in Section 3.2.

Table 3.2 about here

3.3.2 Estimation results

Table 3.2 displays the maximum-likelihood (ML) estimates of five distinct Markov-switching GARCH models represented by the Eqs. (3.1) to (3.10). We numerically maximized the log-likelihood functions from the Eqs. (A.3.1) to (A.3.14) by the use of the BFGS-algorithm as implemented in the FMINCON module of the software package MATLAB. Our estimation results are robust to different starting values. To circumvent numerical problems stemming from the absolute value function appearing on the right-hand side of Eq. (3.7), we follow Hentschel (1995) and replace the argument of the absolute value function by a hyperbolic approximation.[4] Standard errors were computed from the diagonal of the heteroskedasticity-consistent (White-robust) covariance matrix.

[2]Our interest-rate data is provided by the *Deutsche Bundesbank* (the German Central Bank) while we obtain the stock-market data from *Datastream* (daily closing prices).

[3]We divide by 250 in order to be consistent with the approximate number of observations per year available for the DAX returns.

[4]Technical details on the estimation procedure are available upon request.

Our Markov-switching GARCH framework developed in Section 3.2 is so general that it enables us to specify and estimate a large number of distinct two-regime Markov-switching GARCH models. Restrictions on the regime-dependent parameters μ_i, ν_i, b_i and c_i may lead to specific functional forms of the two variance equations, for example to an EGARCH equation in regime 1 ($\mu_1 = 0, \nu_1 = 1, b_1 = 0, c_1 = \text{free}$) and a conventional GARCH equation in regime 2 ($\mu_2 = 2, \nu_2 = 2, b_2 = 0, c_2 = 0$). In what follows, we refer to this latter model as a Markov-switching EGARCH-GARCH model and, based on the terminology in Table 3.1, we analogously use the phrasing TGARCH-GARCH, EGARCH-EGARCH and so on. Because of space constraints, we confine ourselves to estimating five distinct two-regime Markov-switching specifications for the DAX excess returns, namely (1) a conventional GARCH-GARCH model, (2) an AVGARCH-AVGARCH model, (3) an EGARCH-GARCH model, (4) an EGARCH-EGARCH model, and (5) a so-called Free-Free model without any parameter restrictions.

The parameter estimates and standard errors for our five Markov-switching GARCH specifications reported in Table 3.2 can be used to assess the statistical significance of the model parameters. To this end, we consider the conventional t-statistic the exact finite-sample distribution of which is generally unknown in our estimation setup. However, we can make asymptotic inference by noticing (1) that our ML estimators are asymptotically normally distributed, and (2) that our standard errors constitute (weakly) consistent estimates of the true standard deviations of the ML estimators. Consequently, under the null hypothesis of a single parameter being equal to 0, our t-statistics should converge in distribution towards a standard normal variate implying critical values of 1.6449, 1.9600 and 2.5758 at the 10, 5, and 1% levels, respectively, for the absolute value of the t-statistic (see Greene 2008, Appendix D). Following this reasoning, we find (1) that all parameters are statistically significant at least at the 10% level and (2) that the overwhelming majority (namely 80 out 85) parameters are significant at the 1% level.

An important econometric issue concerns the persistence of volatility shocks. In a conventional single-regime GARCH(1,1)-equation of the form $h_t = \omega + \alpha \cdot h_{t-1}\delta_t^2 + \beta h_{t-1}$, the persistence of volatility shocks is typically measured by the sum $\alpha + \beta$. The higher the value of $\alpha + \beta$, the longer it takes until a volatility shock dies out. In particular, when $\alpha + \beta = 1$ volatility shocks have a permanent effect and the unconditional variance of the process gets infinitely large. In view of these considerations in a single-regime framework, it appears natural to measure the persistence of volatility shocks in a two-regime Markov-switching GARCH(1,1) model by the regime-specific sums $\alpha_i + \beta_i$ for $i = 1, 2$. Unfortunately, matters turn out to be more complicated, since in general it is the interaction between the regime-specific volatility parameters and the transition probabilities of the regime indicator S_t which determines the variance-stability of a Markov-switching GARCH model.[5]

Unfortunately, since exact mathematical conditions covering the variance-stability of Markov-switching GARCH models are not available in the literature, we are restricted to analyzing the persistence of volatility shocks within each Markov regime. From Column 1 of Table 3.2 we find that the respective regime-specific sums $\hat{\alpha}_i + \hat{\beta}_i$ for our Markov-switching GARCH-GARCH model are given by 0.9857 and 0.9873 indicating covariance stationarity with high degrees of volatility persistence in both Markov-regimes. A very similar result holds for Regime 2 of our Markov-switching EGARCH-GARCH model (Column 3 of Table 3.2) for which we find $\hat{\alpha}_2 + \hat{\beta}_2 = 0.9884$. For the most general Markov-switching Free-Free model a sufficient condition for covariance stationarity in regime i is given by

$$E\left[\left[\alpha_i \cdot \mu . f^{\nu_i}(\epsilon_t) + \beta_i\right]^{2/\mu_i}\right] < 1 \tag{3.11}$$

(see Nelson, 1990). Hentschel (1995) shows that for an AVGARCH specification with $\mu_i = \nu_i = 1$ condition (3.11) is equivalent to

$$\begin{aligned} &\alpha_i^2(1+b_i^2)(1+c_i^2) + \beta_i^2 + 2\alpha_i\beta_i b_i c_i + 4\alpha_i(\beta_i + \alpha_i b_i c_i)\phi(b_i) \\ &+ 2\alpha_i(\beta_i b_i + \alpha_i(1+b_i^2)c_i)(2\Phi(b_i) - 1) < 1, \end{aligned} \tag{3.12}$$

[5]See for example, Wilfling (2009) and the literature cited there

with $\phi(\cdot)$ and $\Phi(\cdot)$ denoting the probability density and cumulative distribution function of the standard normal distribution, while for a regime-specific EGARCH equation condition (3.11) converges to

$$\beta_i < 1. \tag{3.13}$$

For both AVGARCH regimes in our second Markov-switching specification the estimates from Column 2 of Table 3.2 yield the values 0.9778 an 0.9684 when inserted into the left-hand side of condition (3.12) thus again indicating covariance stationarity with high degrees of volatility persistence in both Markov-regimes. An analogous empirical result obtains for all EGARCH Markov-regimes for which we find estimates of the parameters β_1 and β_2 that are all close to but smaller than 1. Only for the Markov-switching Free-Free specification there is no closed-form solution to the expectation on the left-hand side of condition (3.11). However, we calculated this expectation by numerical integration again finding evidence of covariance stationarity and volatility persistence in both Markov-regimes.

Our time-varying transition probabilities P_t and Q_t from Eq. (3.10) represent the likelihood that no switch in the Markov-regimes occurs between the dates $t-1$ and t. In all of our 5 Markov-switching specifications the probabilities P_t and Q_t are larger than 0.97 at (nearly) every point in time indicating an extremely high degree of regime persistence.

Next, we address several specification issues. As a first diagnostic check we may test for first- and higher-order serial correlation of the squared standardized residuals. To this end we performed Ljung-Box-Q-tests for serial correlation out to various lags for our five Markov-switching specifications. The tests do not reveal any statistical evidence in favor of autocorrelation in the residuals except for the GARCH-GARCH specification for which higher-order serial correlation is detected.[6]

An important specification issue concerns the number of Markov-regimes modeled in our regime-switching representation (3.1) – (3.10). Testing the significance of a second

[6]Details of the autocorrelation tests are available upon request.

Markov-regime is a non-trivial task due to an identification problem known as the Davies Problem (see Davies, 1987). The identification problem implies that a conventional likelihood ratio test (LRT) may be statistically improper since we cannot assume the validity of the χ^2-approximation to the LRT statistic under the null hypothesis of a single Markov-regime any longer. However, Gelman and Wilfling (2009) assess the finite-sample properties of the conventional LRT statistic (defined as twice the difference in the log-likelihoods of the two-regime Markov-switching and the single-regime specifications) for a GARCH-GARCH model by a parametric bootstrapping procedure. Their results indicate that the null distribution of the LRT statistic typically does not exhibit large deviations from the χ^2-distribution with degrees of freedom equal to the difference in the number of parameters between the two-regime and the single-regime specifications. Encouraged by their simulation results, we have conducted the conventional LRT tests for all our five Markov-switching specifications. In all cases the LRT statistics are so extreme that they exceed all critical values used in practice thus endorsing our two-regime specifications estimated in Table 3.2.[7]

Next, we address the question as to which of our five alternative Markov-switching specifications provides the best fit to the data. Obviously, we cannot test all models against each other since two distinct specifications need to be nested in order to assure a likelihood ratio test to be valid. Since our Markov-switching Free-Free model nests all the other specifications (see Table 3.1), we restrict attention to the four testing problems (1) 'H_0: GARCH-GARCH versus H_1: Free-Free', (2) 'H_0: AVGARCH-AVGARCH versus H_1: Free-Free', (3) 'H_0: EGARCH-GARCH versus H_1: Free-Free' and (4) 'H_0: EGARCH-EGARCH versus H_1: Free-Free'.

Table 3.3 about here

Table 3.3 displays the log-likelihood values of all Markov-switching specifications along with the LRT statistics of the four testing problems just mentioned. Obviously, the LR

[7] Details of the LR tests are available upon request.

tests clearly reject the GARCH-GARCH, the AVGARCH-AVGARCH and the EGARCH-EGARCH models against the Free-Free model at significance levels far below the 1% level. Only for the EGARCH-GARCH model the specification testing results are slightly less clear-cut. The p-value 0.0315 indicates that the EGARCH-GARCH model is rejected against the Free-Free model at the 5%, but not at the 1% level.

However, some technical remarks on this latter testing problem are in order. To this end, consider for a moment a single-regime EGARCH and a single-regime Free model. Although theoretically the EGARCH model is nested within the Free model class, testing the EGARCH model against the Free model may cause practical problems. The reason is that in order to guarantee a positive standard deviation for the Free model, we implemented a lower bound for the parameters ω, α and β at zero. Within the Free model class these parameter restrictions ensure positive standard deviations for all models with $\mu \neq 0$. Theoretically, for specifications within the Free model class with $\mu = 0$ these restrictions are no longer necessary. However, when estimating the Free model specification we retained the parameter restrictions for ω, α and β to (1) facilitate numerical optimization, and (2) to be capable of computing standard errors of our estimates. By contrast, when estimating an EGARCH specification with $\mu = 0, \nu = 1, b = 0$, we followed standard practice and did not impose the (unnecessary) restrictions on the parameters ω, α and β. Since in this setting the Free model does not really nest the EGARCH model, it is theoretically possible that a two-regime Markov-switching model with an EGARCH specification in at least one regime might have a higher log-likelihood value than the alternative Free-Free model. From a probabilistic point of view this implies an increased Type II error of the test and thus a lower power of the test.

Figure 3.2 about here

Figure 3.2 displays the *ex ante* regime probabilities calculated according to Eq. (A.3.7) along with the conditional variances of the daily excess returns of the German stock market index DAX as estimated by our five Markov-switching GARCH specifications. For all five models the conditional variances exhibit a strikingly uniform pattern during the sample interval between the years 2000 and 2010. The beginning of the decade started with a period of relatively high volatility in the German stock market with a pronounced peak in conditional variances around 11 September 2001. After a short phase of normalization, an extended period of high stock-market volatility occurred between mid-2002 and the end of 2003 reflecting the German bear market in which the DAX fell from about 5000 to 2000 index points. Between 2004 and the beginning of the year 2008 the conditional volatility of the DAX was comparably low. This period of low market fluctuation came to an abrupt end at the beginning of the year 2008 when the German stock market began to respond to the subprime crisis by plummeting stock prices. However, the highest volatility peak occurred around 15 September 2008 when Lehman Brothers Holdings Inc. filed for *Chapter 11 bankruptcy protection*.

Analyzing the *ex-ante* probabilities in Figure 3.2, we find that all our five Markov-switching models generate two or more pronounced regime switches. Some of these regime switches appear to occur at the same time irrespective of the chosen Markov-switching specification. The most clear-cut example is the switch at the end of the year 2008 possibly indicating a structural change in the German excess returns since the financial crisis. Four out of five specifications—including our EGARCH-GARCH and Free-Free models—report a regime switch around June 2006 when a sustained bullish trend in the German stock market began. Obviously, the regimes 1 and 2 estimated via the *ex-ante* probabilities of our five Markov-switching models do not necessarily coincide with the low- and high-volatility periods depicted in the neighboring panels. A first explanation of this finding is that each Markov-switching specification allows for switching mean and switching volatility equations so that a regime-switch may solely be induced by a switch in the mean equation alone. A second explanation is that each regime-specific variance specification is capable of capturing certain qualitative volatility features (e.g. specific volatility asymmetries) which do not directly affect the volatility level, but which may nevertheless perform a structural switch from one regime to another.

However, the most efficient way of investigating switching volatility structures is to analyze the Free-Free model, which clearly outperforms all other specifications. In this model class we can test for pairwise equality of the corresponding regime-specific volatility parameters (i.e. $\mu_1 = \mu_2, \nu_1 = \nu_2, \omega_1 = \omega_2, \alpha_1 = \alpha_2, \beta_1 = \beta_2, b_1 = b_2, c_1 = c_2$). Apart from the parameters β_1 and β_2, all other corresponding volatility parameter appear to be considerably different from each other across both regimes thus indicating substantial structural differences between both volatility regimes in the German stock index DAX.

3.4 Conclusion

In this chapter we establish a two-regime Markov-switching GARCH model which enables us to estimate complex functional GARCH specifications within each regime. Combining Gray's (1996) and Klaassen's (2002) Markov-switching framework with Hentschel's (1995) approach of nesting alternative single-regime GARCH models, our framework unifies all Markov-switching GARCH models that have been estimated hitherto in the financial literature. Apart from complex regime-specific GARCH specifications, our model features two further empirically relevant attributes, namely (1) a GARCH-in-Mean specification of the mean equation, and (2) time-varying transition probabilities describing the dynamics of the latent regime-indicator.

In the technical appendix of this chapter, we develop a reliable maximum likelihood estimation algorithm for our model which we apply to appropriately constructed daily excess returns of the German stock index DAX for the time between January 2000 and December 2009. Our empirical analysis reveals that our model unambiguously outperforms all alternative Markov-switching GARCH models applied so far in the literature. Moreover, we find significant Markov-switching in the German stock market with substantially differing volatility structures across both Markov-regimes.

A natural line of future research could be the extension of our framework to more than two Markov-regimes. This, however, leads to highly parameterized models which become increasingly difficult to estimate. However, other estimation procedures than our ML approach may be implemented, for example Bayesian Markov-Chain Monte-Carlo (MCMC) algorithms which have the potential to provide an alternative way of circumventing the problem of path dependence (see Bauwens et al., 2010).

Appendix

In this appendix we construct the log-likelihood function for our Markov-switching GARCH model established in Section 3.2. We only consider the two-regime case although a theoretical extension of the entire framework to more Markov regimes is straightforward.

The conditional probability distribution of r_{t+1} is shown in Eq. (3.2). The corresponding probability density function has the form

$$\begin{aligned} f(r_{t+1}|\phi_t) &= \sum_{i=1}^{2} f(r_{t+1}, S_t = i|\phi_t) \\ &= \sum_{i=1}^{2} \Pr(S_t = i|\phi_t) \cdot f(r_{t+1}|S_t = i, \phi_t) \\ &= \sum_{i=1}^{2} p_{i,t} \cdot f(r_{t+1}|S_t = i, \phi_t), \end{aligned} \tag{A.3.1}$$

where, as in the main text, $p_{i,t} \equiv \Pr(S_t = i|\phi_t)$ denotes the *ex-ante* probability. The information set ϕ_t consists of the entire history of $\tilde{r}_t = \{r_t, r_{t-1}, \ldots\}$.

Since the regime indicator S_t follows a first-order Markov process the *ex-ante* probability $p_{i,t}$ depends only on S_{t-1} and r_t. Using the *Theorem of Total Probabilities*, we obtain

$$p_{i,t} = \sum_{j=1}^{2} \Pr(S_t = i|S_{t-1} = j, \tilde{r}_t) \Pr(S_{t-1} = j|\tilde{r}_t). \tag{A.3.2}$$

The first probability $\Pr(S_t = i|S_{t-1} = j, \tilde{r}_t)$ on the right-hand side of (A.3.2) does not depend on the entire history of $\tilde{r}_t$ so that we replace $\tilde{r}_t$ by r_t in this latter probability. Thus, we can insert the probabilities specified in Eq. (3.9) in Eq. (A.3.2) and obtain

$$\begin{aligned} p_{1,t} &= P_t \cdot \Pr(S_{t-1} = 1|\tilde{r}_t) + (1 - Q_t) \cdot \Pr(S_{t-1} = 2|\tilde{r}_t) \\ &= P_t \cdot \Pr(S_{t-1} = 1|\tilde{r}_t) + (1 - Q_t) \cdot (1 - \Pr(S_{t-1} = 1|\tilde{r}_t)), \end{aligned} \tag{A.3.3}$$

and analogously

$$p_{2,t} = Q_t \cdot (1 - \Pr(S_{t-1} = 1|\tilde{r}_t)) + (1 - P_t) \cdot \Pr(S_{t-1} = 1|\tilde{r}_t). \tag{A.3.4}$$

The remaining probability $\Pr(S_{t-1} = 1|\tilde{r}_t)$ in the Eqs. (A.3.3) and (A.3.4) can be written as a function of $p_{1,t-1} = \Pr(S_{t-1} = 1|\tilde{r}_{t-1})$. To this end, we apply *Bayes' Formula* yielding

$$\begin{aligned} \Pr(S_{t-1} = 1|\tilde{r}_t) &= \Pr(S_{t-1} = 1|r_t, \tilde{r}_{t-1}) \\ &= \frac{f(r_t|S_{t-1} = 1, \tilde{r}_{t-1}) \Pr(S_{t-1} = 1, \tilde{r}_{t-1})}{\sum_{i=1}^{2} f(r_t|S_{t-1} = i, \tilde{r}_{t-1}) \Pr(S_{t-1} = i, \tilde{r}_{t-1})}. \end{aligned} \tag{A.3.5}$$

Expanding the ratio on the right-hand side of Eq. (A.3.5), we obtain

$$\begin{aligned} \Pr(S_{t-1} = 1|\tilde{r}_t) &= \frac{f(r_t|S_{t-1} = 1, \tilde{r}_{t-1}) \Pr(S_{t-1} = 1|\tilde{r}_{t-1})}{\sum_{i=1}^{2} f(r_t|S_{t-1} = i, \tilde{r}_{t-1}) \Pr(S_{t-1} = i|\tilde{r}_{t-1})} \\ &= \frac{f(r_t|S_{t-1} = 1, \tilde{r}_{t-1}) p_{1,t-1}}{\sum_{i=1}^{2} f(r_t|S_{t-1} = i, \tilde{r}_{t-1}) p_{i,t-1}} \\ &= \frac{g_{1,t-1} \cdot p_{1,t-1}}{\sum_{i=1}^{2} g_{i,t-1} \cdot p_{i,t-1}}, \end{aligned} \tag{A.3.6}$$

where, for ease of notation, we have defined $g_{i,t-1} \equiv f(r_t|S_{t-1} = i, \tilde{r}_{t-1}) = f(r_t|S_{t-1} = i, \phi_{t-1})$. Using Eq. (A.3.6), we are now able to calculate the *ex-ante* probability $p_{1,t}$ by inserting Eq. (A.3.6) in Eq. (A.3.3):

$$\begin{aligned} p_{1,t} &= P_t \cdot \frac{g_{1,t-1} p_{1,t-1}}{g_{1,t-1} p_{1,t-1} + g_{2,t-1}(1 - p_{1,t-1})} + (1 - Q_t) \cdot \left[1 - \frac{g_{1,t-1} p_{1,t-1}}{g_{1,t-1} p_{1,t-1} + g_{2,t-1}(1 - p_{1,t-1})}\right] \\ &= P_t \cdot \frac{g_{1,t-1} p_{1,t-1}}{g_{1,t-1} p_{1,t-1} + g_{2,t-1}(1 - p_{1,t-1})} + (1 - Q_t) \cdot \frac{g_{2,t-1}(1 - p_{1,t-1})}{g_{1,t-1} p_{1,t-1} + g_{2,t-1}(1 - p_{1,t-1})} \end{aligned} \tag{A.3.7}$$

Next, we address the exact form of the conditional density f appearing in the Eqs. (A.3.1) and (A.3.7). As we are assuming conditional normality f is given as follows:

$$f(r_{t+1}|S_t = i, \phi_t) = \frac{1}{\sqrt{2\pi h_{i,t}}} \exp\left\{-\frac{\left[r_{t+1} - (\lambda_i + \gamma_i\sqrt{h_{i,t}})\right]^2}{2h_{i,t}}\right\}. \qquad \text{(A.3.8)}$$

The variance $h_{i,t}$ depends on the explicit functional form of the GARCH equation. It is easy to check from Eq. (3.8) that it can be written as

$$h_{i,t} = \begin{cases} \left[\omega_i + \alpha_i\sqrt{h_{t-1}^{(i)}}^{\mu_i} f_i^{\nu_i}(\delta_t^{(i)}) + \beta_i\sqrt{h_{t-1}^{(i)}}^{\mu_i}\right]^{2/\mu_i} & \text{for} \quad \mu_i > 0 \\ \left[\exp\left\{\omega_i + \alpha_i f_i^{\nu_i}(\delta_t^{(i)}) + \beta_i \ln\left(\sqrt{h_{t-1}^{(i)}}\right)\right\}\right]^2 & \text{for} \quad \mu_i = 0 \end{cases} \qquad \text{(A.3.9)}$$

with appropiately defined parameters $\omega_i, \alpha_i, \beta_i$.

It is obvious from Eq. (A.3.9) that for the calculation of regime-specific variances $h_{i,t}$ we need the aggregated variances and shock terms $h_{t-1}^{(i)}$ and $\delta_t^{(i)}$ the calculation of which we base on the Klaassen (2002) probabilities $p_{1,t-1}^{(i)}$ as described in the main text. Using *Bayes' Formula* again, we obtain the Klaassen probabilities as

$$\begin{aligned} p_{1,t-1}^{(i)} &= \Pr(S_{t-1} = 1|\tilde{r}_{t-1}, S_t = i) \\ &= \frac{\Pr(S_t = i|\tilde{r}_{t-1}, S_{t-1} = 1)\Pr(S_{t-1} = 1|\tilde{r}_{t-1})}{\Pr(S_t = i|\tilde{r}_{t-1})} \\ &= \frac{\Pr(S_t = i|\tilde{r}_{t-1}, S_{t-1} = 1) \cdot p_{1,t-1}}{\Pr(S_t = i|\tilde{r}_{t-1})} \end{aligned} \qquad \text{(A.3.10)}$$

with $p_{1,t-1}$ as given in Eq. (A.3.7). Applying the *Theorem of Total Probabilities* once more, we write the denominator in Eq. (A.3.10) as

$$\begin{aligned} \Pr(S_t = i|\tilde{r}_{t-1}) &= \Pr(S_t = i|\tilde{r}_{t-1}, S_{t-1} = 1) \cdot p_{1,t-1} \\ &\quad + \Pr(S_t = i|\tilde{r}_{t-1}, S_{t-1} = 2) \cdot (1 - p_{1,t-1}). \end{aligned} \qquad \text{(A.3.11)}$$

To calculate the probability on the left-hand of Eq. (A.3.11) we need the two probabilities $\Pr(S_t = i|\tilde{r}_{t-1}, S_{t-1} = 1)$ and $\Pr(S_t = i|\tilde{r}_{t-1}, S_{t-1} = 2)$. To be consistent with the specifications (3.9) and (3.10) for the time-varying transition probabilities, we have to choose appropriate forecasts of the return r_t conditional on either $\tilde{r}_{t-1}, S_{t-1} = 1$ or $\tilde{r}_{t-1}, S_{t-1} = 2$. In what follows we use the conditional expectations $E(r_t|\tilde{r}_{t-1}, S_{t-1} = 1) = \lambda_1 + \gamma_1\sqrt{h_{1,t-1}}$ and $E(r_t|\tilde{r}_{t-1}, S_{t-1} = 2) = \lambda_2 + \gamma_2\sqrt{h_{2,t-1}}$ which are known to be optimal forecasts with respect to the mean squared error (MSE). Thus, we obtain

$$\begin{aligned}
\Pr(S_t = 1|\tilde{r}_{t-1}, S_{t-1} = 1) &= \Phi\left(d_1 + e_1 \cdot \left[\lambda_1 + \gamma_1\sqrt{h_{1,t-1}}\,\right]\right), \\
\Pr(S_t = 2|\tilde{r}_{t-1}, S_{t-1} = 1) &= 1 - \Phi\left(d_1 + e_1 \cdot \left[\lambda_1 + \gamma_1\sqrt{h_{1,t-1}}\,\right]\right), \\
\Pr(S_t = 1|\tilde{r}_{t-1}, S_{t-1} = 2) &= 1 - \Phi\left(d_2 + e_2 \cdot \left[\lambda_2 + \gamma_2\sqrt{h_{2,t-1}}\,\right]\right), \\
\Pr(S_t = 2|\tilde{r}_{t-1}, S_{t-1} = 2) &= \Phi\left(d_2 + e_2 \cdot \left[\lambda_2 + \gamma_2\sqrt{h_{2,t-1}}\,\right]\right).
\end{aligned} \tag{A.3.12}$$

Now, inserting the Eqs. (A.3.12) and (A.3.11) in Eq. (A.3.10) we obtain

$$\begin{aligned}
p_{1,t-1}^{(1)} &= \frac{\Phi(d_1 + e_1[\lambda_1 + \gamma_1\sqrt{h_{1,t-1}}])p_{1,t-1}}{\Phi(d_1 + e_1[\lambda_1 + \gamma_1\sqrt{h_{1,t-1}}])p_{1,t-1} + \{1 - \Phi(d_2 + e_2[\lambda_2 + \gamma_2\sqrt{h_{2,t-1}}])\}p_{2,t-1}}, \\
p_{2,t-1}^{(1)} &= 1 - p_{1,t-1}^{(1)}, \\
p_{1,t-1}^{(2)} &= \frac{\{1 - \Phi(d_1 + e_1[\lambda_1 + \gamma_1\sqrt{h_{1,t-1}}])\}p_{1,t-1}}{\{1 - \Phi(d_1 + e_1[\lambda_1 + \gamma_1\sqrt{h_{1,t-1}}])\}p_{1,t-1} + \Phi(d_2 + e_2[\lambda_2 + \gamma_2\sqrt{h_{2,t-1}}])p_{2,t-1}}, \\
p_{2,t-1}^{(2)} &= 1 - p_{1,t-1}^{(2)}.
\end{aligned} \tag{A.3.13}$$

Finally, we use the recursive structures developed so far to construct the log-likelihood function of our flexible Markov-switching model defined in the Eqs. (3.1) to (3.10). The general form of the likelihood function is

$$L(\boldsymbol{\Theta}) = f(r_t, \ldots, r_1; \boldsymbol{\Theta})$$

with the vector $\boldsymbol{\Theta}$ containing all model parameters. Writing this joint distribution of the returns as a product of conditional densities, we obtain

$$L(\boldsymbol{\Theta}) = \prod_{t=1}^{T} f(r_t|\tilde{r}_{t-1}; \boldsymbol{\Theta})$$

for which we define the starting term as $f(r_1|\tilde{r}_0;\mathbf{\Theta}) \equiv f(r_1;\mathbf{\Theta})$. Taking the logarithm of $L(\mathbf{\Theta})$ and inserting (the lagged form of) Eq. (A.3.1), we obtain the log-likelihood function as

$$\begin{aligned}\ell(\mathbf{\Theta}) \equiv \log[L(\mathbf{\Theta})] &= \sum_{t=1}^{T} \log\left[f(r_t|\tilde{r}_{t-1};\mathbf{\Theta})\right] \\ &= \sum_{t=1}^{T} \log\left[\sum_{j=1}^{2} f(r_t|S_{t-1}=j,\tilde{r}_{t-1};\mathbf{\Theta}) \cdot p_{j,t-1}\right]. \qquad \text{(A.3.14)}\end{aligned}$$

Figures

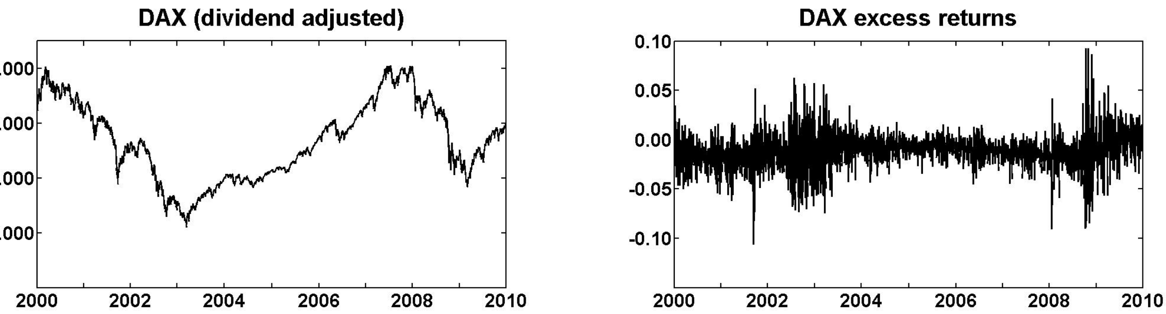

Figure 3.1: Dividend adjusted DAX and DAX excess returns (2000 - 2009)

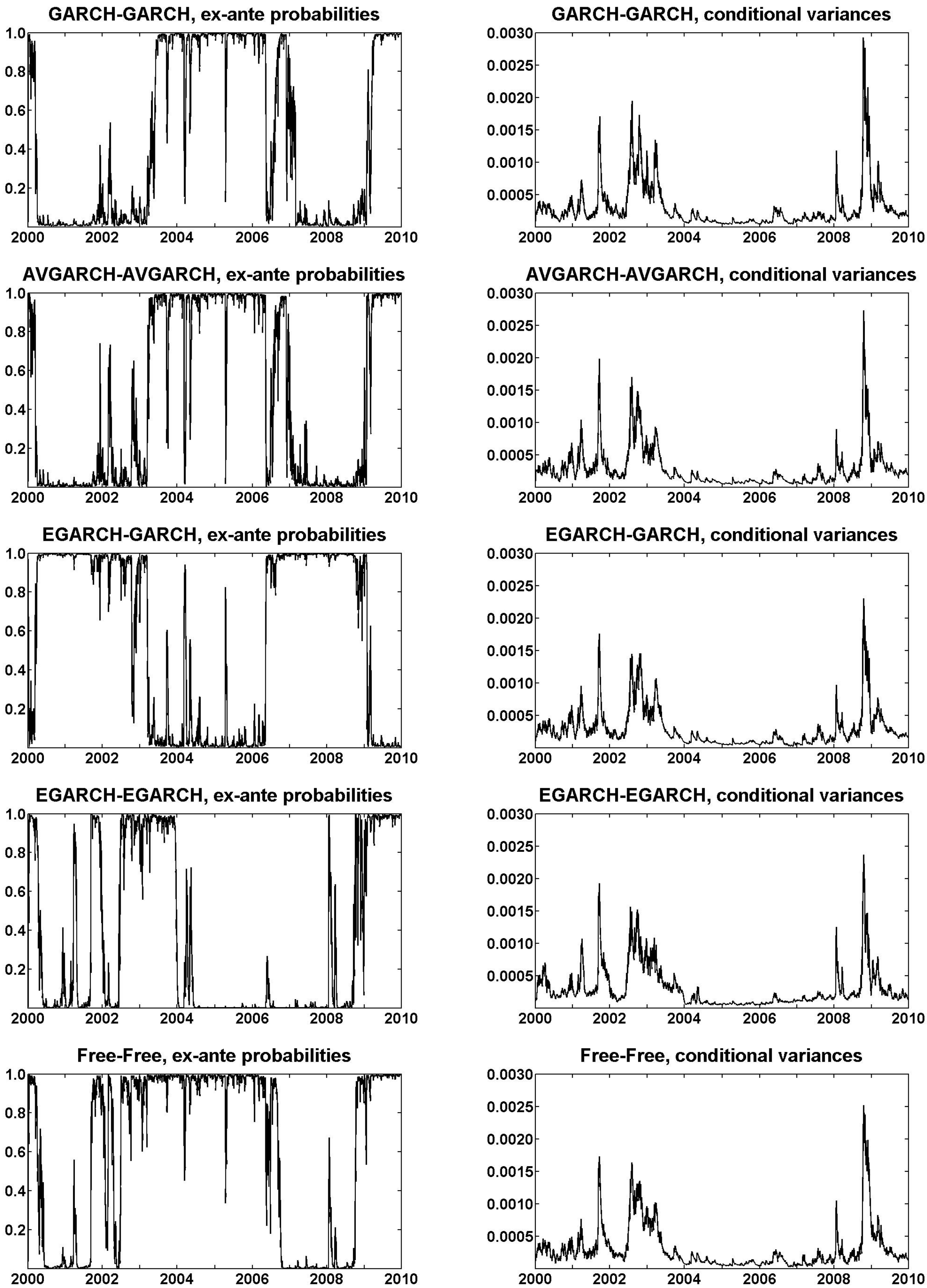

Figure 3.2: *Ex-ante* probabilities and conditional variances of five Markov-switching GARCH models

Tables

μ_i	ν_i	b_i	c_i	Model	Reference
0	1	0	free	Exponential GARCH (EGARCH)	Nelson (1991)
1	1	0	$\lvert c_i \rvert \leq 1$	Threshold GARCH (TGARCH)	Zakoian (1994)
1	1	free	$\lvert c_i \rvert \leq 1$	Absolute Value GARCH (AVGARCH)	Hentschel (1995)
2	2	0	0	Standard GARCH (GARCH)	Bollerslev (1986)
2	2	free	0	Nonlinear-asymmetric GARCH	Engle and Ng (1993)
2	2	0	free	Glosten-Jagannathan-Runkle GARCH	Glosten et al. (1993)
free	μ_i s	0	0	Nonlinear ARCH	Higgins and Bera (1992)
free	μ_i	0	$\lvert c_i \rvert \leq 1$	Asymmetric power ARCH	Ding et al. (1993)

Note: Table compiled from Hentschel (1995, Table 1).

Table 3.1: Nested GARCH models (within regime i)

	GARCH–GARCH	AVGARCH–AVGARCH	EGARCH–GARCH	EGARCH–EGARCH	Free–Free
μ_1	2.0000	1.0000	0.0000	0.0000	0.7597***
					(0.0156)
μ_2	2.0000	1.0000	2.0000	0.0000	0.0024**
					(0.0011)
ν_1	2.0000	1.0000	1.0000	1.0000	2.0126***
					(0.0053)
ν_2	2.0000	1.0000	2.0000	1.0000	1.4421***
					(0.0364)
λ_1	−0.0113***	−0.0120***	−0.0119***	0.0043***	−0.0071***
	(0.0006)	(0.0001)	(0.0001)	(0.0000)	(0.0007)
λ_2	−0.0176***	−0.0156***	−0.0108***	0.0028***	−0.0091***
	(0.0001)	(0.0001)	(0.0000)	(0.0000)	(0.0003)
γ_1	0.3853***	0.4403***	−0.2871***	−0.5740***	−0.0564***
	(0.0009)	(0.0011)	(0.0001)	(0.0000)	(0.0003)
γ_2	0.1285***	−0.0739***	0.3998***	−1.3937***	−0.6234***
	(0.0015)	(0.0001)	(0.0013)	(0.0000)	(0.0004)
ω_1	0.0000***	0.0001*	−0.1252***	−0.0989***	0.0003**
	(0.0000)	(0.0001)	(0.0002)	(0.0000)	(0.0002)
ω_2	0.0000*	0.0004***	0.0000***	−0.0600***	0.0250***
	(0.0000)	(0.0001)	(0.0000)	(0.0000)	(0.0003)
α_1	0.0594***	0.1081***	0.0741***	0.0365***	0.0259***
	(0.0008)	(0.0003)	(0.0001)	(0.0000)	(0.0005)
α_2	0.1219***	0.0896***	0.0464***	0.0108***	0.0001**
	(0.0024)	(0.0004)	(0.0001)	(0.0000)	(0.0000)
β_1	0.9263***	0.9647***	0.9839***	0.9819***	0.9624***
	(0.0043)	(0.0061)	(0.0005)	(0.0005)	(0.0056)
β_2	0.8654***	0.9082***	0.9420***	0.9886***	0.9743***
	(0.0208)	(0.0036)	(0.0029)	(0.0002)	(0.0006)
d_1	2.8417***	2.7771***	3.3509***	3.0705***	2.7773***
	(0.0473)	(0.0100)	(0.0036)	(0.0018)	(0.0687)
d_2	2.9047***	2.5439***	2.7844***	5.0867***	4.2702***
	(0.0368)	(0.0205)	(0.0052)	(0.0048)	(0.0645)
e_1	2.1697***	9.1372***	18.5418***	−36.9247***	−5.5716***
	(0.0215)	(0.0544)	(0.1223)	(0.0000)	(0.0011)
e_2	6.9612***	−11.8090***	−5.7178***	76.5404***	46.1943***
	(0.0440)	(0.0001)	(0.0001)	(0.0937)	(0.7260)
b_1	0.0000	1.2469***	0.0000	0.0000	0.6021***
		(0.0063)			(0.0071)
b_2	0.0000	0.0098***	0.0000	0.0000	2.5977***
		(0.0001)			0.0742)
c_1	0.0000	−0.9086***	0.7998***	1.4149***	−0.1538***
		(0.0001)	(0.0019)	(0.0011)	(0.0001)
c_2	0.0000	0.7351***	0.0000	3.8618***	−0.0319***
		(0.0018)		(0.0038)	(0.0001)

Note: Estimates for parameters from the Eqs. (3.1) to (3.10). Standard errors are in parentheses. *,** and *** denote statistical significance at 10, 5 and 1% levels, respectively.

Table 3.2: Estimates of alternative the Markov-switching GARCH specifications

	GARCH–GARCH	AVGARCH–AVGARCH	EGARCH–GARCH	EGARCH–EGARCH	Free–Free
Log-likelihood	7290.4170	7311.4635	7323.5311	7308.1564	7331.2173
LRT statistic vs Free-Free model	81.6006^{***} [0.0000]	39.5076^{***} [0.0000]	15.3724^{**} [0.0315]	46.1218^{***} [0.0000]	
χ^2-df	8.0000	4.0000	7.0000	6.0000	

Note: The LRT statistic of the testing problem 'H_0: the considered two-regime specification versus H_1: the two-regime Free-Free specification' is computed as twice the difference in the log-likelihoods of the Free-Free specification and the two-regime specification under the null hypothesis. The LRT statistics are asymptotically χ^2-distributed under the respective null hypotheses with degree-of-freedom parameters as given in the row 'χ^2-df'. p-values are in squared brackets. *,** and *** denote statistical significance at 10, 5 and 1% levels, respectively.

Table 3.3: Log-likelihood values and likelihood ratio tests

Chapter 4

Short-selling constraints and stock-return volatility: empirical evidence from the German stock market

4.1 Introduction

During the financial crisis 2008/2009 and the Greek crisis 2009/2010 governments in many countries imposed limitations on short-selling activities to displace short sellers and prevent further declines in stock prices. While governments, regulators, and the media blame short sellers for reinforcing stock market downturns, academic research mostly finds distortions of short-selling restrictions on market efficiency, liquidity and pricing. Surprisingly little is known about the impact of short selling-restrictions on stock returns volatility. We expect an increase in volatility due to short-selling restrictions because they limit the ability of investors to find the fundamental price. Consequently, short-selling bans contribute to a destabilization of stock prices during periods of market downturns and even exaggerate stock-price declines. Hence, short-selling bans are counterproductive.

We apply a version of an asymmetric Markov-switching GARCH model to the recent short-selling bans from September 2008 to January 2010 and their re-installment from May 2010 onwards on stocks of financial firms in Germany. The main advantage of the Markov-switching GARCH method is that it does not require an exogenously pre-determined date for the shift in stock returns volatility. Instead, these models allow for endogenous specifications of volatility regime shifts and thus let the data speak for themselves.

The investigation of the short-selling restrictions in Germany is motivated by the length of the ban period. Given that short-selling bans are often imposed for relatively brief periods the majority of relevant analyses relies on cross-sectional regressions or resorts to event-type studies. The recent German experience has the advantage that it permits us to exploit time-series variation in the volatility of stock returns, thereby allowing the testing of implications of short-selling restrictions on higher moments in the distribution of stock returns. To our best knowledge, we are the first to specify and estimate an asymmetric Markov-switching GARCH model to test hypotheses surrounding the effects of restrictions on short sales for the data under investigation.

The remainder of this chapter is structured as follows. Section 4.2 considers the Markov-switching GARCH framework. Section 4.3 provides the data and the empirical findings, while Section 4.4 concludes.

4.2 A Markov-switching GARCH framework

In this section we specify our econometric model. Since we aim to investigate as to what extent the restrictions of naked short sales may have a stabilizing impact on the stock market we focus on measuring stock-return volatility before, during and after restriction periods. Movements of volatility through time are typically well-captured by some sort of GARCH model. However, the fact that we expect short-selling constraints to induce a change in the behavior of market participants suggests estimating of a Markov-switching GARCH model.

In our empirical analysis below we estimate two distinct Markov-switching GARCH specifications which are special cases of a very general two-regime Markov-switching GARCH framework developed by Reher and Wilfling (2010). In our first specification one Markov-regime is governed by an EGARCH process (see Nelson, 1991) and the second Markov-regime is governed by a TGARCH process (see Zakoian, 1994). While we refer to this specification as a (Markov-switching) EGARCH-TGARCH model, our second specification assumes distinct TGARCH processes in both Markov-regimes suggesting the term TGARCH-TGARCH for this model.

To build up our econometric specifications we assume that the data generating process (DGP) of the stock-price return r_t is affected by an unobserved latent random variable S_t representing the regime the DGP is in at time t. As mentioned above, we assume only two distinct regimes 1 and 2 at any point in time, that is we assume either $S_t = 1$ or $S_t = 2$ for all $t = 1, 2, \ldots$. Since we focus on the volatility effects of short-selling restrictions, we model a constant, non-switching mean λ across both Markov-regimes and only let the volatility processes switch between the regimes.

To this end we let the return dynamics depend on the regime indicator $S_t = i, i = 1, 2$ and specify

$$r_{t+1} = \lambda + \sqrt{h_{i,t}}\epsilon_{t+1}. \tag{4.1}$$

In Eq. (4.1) $\{\epsilon_t\}$ is an i.i.d. process of standard normal variates, while $h_{i,t}$ represents the conditional variance in regime i. Following Reher and Wilfling (2010), we define our variance equation in its most general form as

$$h_{i,t} = \begin{cases} \left[\omega_i + \alpha_i \sqrt{h_{t-1}^{(i)}}^{\mu_i} f_i^{\nu_i}(\delta_t^{(i)}) + \beta_i \sqrt{h_{t-1}^{(i)}}^{\mu_i}\right]^{2/\mu_i} & \text{for} \quad \mu_i > 0 \\ \left[\exp\left\{\omega_i + \alpha_i f_i^{\nu_i}(\delta_t^{(i)}) + \beta_i \ln\left(\sqrt{h_{t-1}^{(i)}}\right)\right\}\right]^2 & \text{for} \quad \mu_i = 0 \end{cases}, \tag{4.2}$$

with

$$f_i(\delta_t^{(i)}) = |\delta_t^{(i)}| - c_i\delta_t^{(i)}. \tag{4.3}$$

In the Eqs. (4.2) and (4.3) the quantities $\omega_i, \alpha_i, \beta_i, \mu_i$ and ν_i are parameters which are typically estimated from the data. The lagged variance term $h_{t-1}^{(i)}$ is an average value of the regime-dependent lagged value $h_{i,t-1}$, while $\delta_t^{(i)}$ is an average value of the regime-dependent estimates of ϵ_t. Both quantities will be explained below. The parameters μ_i and ν_i are shape parameters which may be predetermined in order to preselect alternative distinct and well-known GARCH models.[1] In our empirical analysis below, for example, we estimate two different Markov-switching specifications. In our first specification, a Markov-switching EGARCH-TGARCH model, we predetermine $\mu_1 = 0$ and $\nu_1 = 1$ in order to obtain an EGARCH equation in regime 1, while we set $\mu_2 = 1$ and $\nu_2 = 1$ to obtain a TGARCH equation in regime 2. Via this predetermination Eq. (4.2) becomes

$$h_{i,t} = \begin{cases} \left[\exp\left\{\omega_1 + \alpha_1 f_1(\delta_t^{(1)}) + \beta_1 \ln\left(\sqrt{h_{t-1}^{(1)}}\right)\right\}\right]^2 & \text{for} \quad S_t = 1 \\ \left[\omega_2 + \alpha_2\sqrt{h_{t-1}^{(2)}} f_2(\delta_t^{(2)}) + \beta_2\sqrt{h_{t-1}^{(2)}}\right]^2 & \text{for} \quad S_t = 2 \end{cases}. \tag{4.4}$$

In our second empirical specification, a Markov-switching TGARCH-TGARCH model, we predetermine $\mu_1 = \mu_2 = 1$ and $\nu_1 = \nu_2 = 1$ so that $h_{i,t}$ adapts accordingly.

The quantities $h_{t-1}^{(i)}$ and $\delta_t^{(i)}$ have been introduced in the above volatility equations in order to circumvent the problem of *path dependence* which is typically inherent in Markov-switching GARCH models. This collapsing procedure is originally due to Gray (1996) and was later refined by Klaassen (2002). Reher and Wilfling (2010) present a detailed description of how to calculate both quantities in the general Markov-switching GARCH setup stated above. For the purpose of this chapter, it is sufficient to mention that for expressing $h_{t-1}^{(i)}$ and $\delta_t^{(i)}$ we need certain regime probabilities, denoted by $p_{1,t-1}^{(i)}$, which represent the probability that the DGP was in regime 1 at date $t-1$ when the DGP

[1]This general parameterization of the GARCH equation in a single-regime framework is due to Hentschel (1995).

currently (i.e. at date t) is regime i. Via these probabilities, which can be computed from Eq. (A.4.13) in the appendix to this chapter, we have

$$h_{t-1}^{(i)} = p_{1,t-1}^{(i)} h_{1,t-1} + (1 - p_{1,t-1}^{(i)}) h_{2,t-1} \tag{4.5}$$

and

$$\delta_t^{(i)} = p_{1,t-1}^{(i)} \frac{r_t - \lambda}{\sqrt{h_{1,t-1}}} + (1 - p_{1,t-1}^{(i)}) \frac{r_t - \lambda}{\sqrt{h_{2,t-1}}}. \tag{4.6}$$

Finally, we close our econometric model by specifying the probabilistic nature of the regime indicator S_t. To keep the analysis tractable, we model S_t as a first-order Markov process with constant transition probabilities π_1 and π_2:

$$\begin{aligned} \Pr\left(S_t = 1 | S_{t-1} = 1\right) &= \pi_1, \\ \Pr\left(S_t = 2 | S_{t-1} = 1\right) &= 1 - \pi_1, \\ \Pr\left(S_t = 1 | S_{t-1} = 2\right) &= 1 - \pi_2, \\ \Pr\left(S_t = 2 | S_{t-1} = 2\right) &= \pi_2. \end{aligned} \tag{4.7}$$

Our Markov-switching GARCH model established in the Eqs. (4.1) to (4.7) can now be estimated using (quasi) maximum likelihood techniques. The log-likelihood function is constructed recursively and we present its exact form in the Eqs. (A.4.14) of the appendix.

4.3 Data and estimation results

4.3.1 Data

On 19 September 2008 the German *Federal Financial Supervisory Authority* (BaFin) announced the prohibition of naked short sales of the shares for the following 11 enterprises: Aareal Bank, Allianz, AMB Generali Holding, Commerzbank, Deutsche Bank, Deutsche Börse, Deutsche Postbank, Hannover Rückversicherung, Hypo Real Estate Holding, MLP, and Münchener Rückversicherungsgesellschaft. Initially, the BaFin envisaged to sustain this short-selling restriction until 31 December 2008, but then prolonged it in three steps, namely (1) until 31 March 2009, (2) until 31 May 2009, and finally (3) until 31 January 2010. Overall, this amounts to a period of 343 trading days in succession during which the restriction was in force. From 1 February 2010 onwards the BaFin lifted the restriction, but reinstalled it on 18 May 2010 for the same group of enterprises except for the Hypo Real Estate Holding which had ceased being listed in the meantime.

In this chapter we aim at assessing the overall volatility effects of these short-selling constraints, but refrain from analyzing the volatility effects on a single share. For this purpose we construct an index from the stock returns of the above-mentioned enterprises, but exclude the Hypo Real Estate Holding which was delisted during the sampling period. In what follows we refer to these 10 enterprises as our sample group. As our index weights we use daily market values which we observe between 2 January 2006 and 23 June 2010 (1136 observations).

In order to compare our volatility results with a control group we constructed a second index consisting of all DAX enterprises that were not subject to short-selling constraints (i.e. which are not among the enterprises of the sample group). Additionally, we excluded Volkswagen from the control group since take-over speculation during the sampling period caused abnormal stock-price behavior that might interfere with our research question.

Figure 4.1 about here

Figure 4.1 displays the daily index returns for both groups during the sampling period. In each panel we mark the two periods without and with short-selling restrictions by differently colored lines with the dark line highlighting the periods subject to short-selling restrictions. Both trajectories clearly exhibit volatility clustering, a prominent feature well-documented in the financial literature on asset-return dynamics. We now turn to analyzing the volatility structures in both return series within our Markov-switching GARCH framework from Section 4.2.

Table 4.1 about here

4.3.2 Estimation results

Although the general Markov-switching GARCH framework established in Section 4.2 covers a broad range of distinct regime-specific GARCH equations, we restrict attention in this chapter to two alternative Markov-switching GARCH specifications, namely an EGARCH-TGARCH and a TGARCH-TGARCH model.[2] Table 4.1 displays the maximum likelihood (ML) estimates of both Markov-switching GARCH models—as represented by the Eqs. (4.1) to (4.7)—for the index returns of our sample and control groups. We numerically maximized the log-likelihood functions from the Eqs. (A.4.1) to (A.4.14) by the use of the BFGS-algorithm as implemented in the FMINCON module of the software package MATLAB. Our estimation results are robust to different starting values. To circumvent numerical problems stemming from the absolute value function appearing on the right-hand side of Eq. (4.3), we follow Hentschel (1995) and replace the argument of the absolute value function by a hyperbolic approximation.[3] Standard errors were computed from the diagonal of the heteroskedasticity-consistent (White-robust) covariance matrix.

The parameter estimates and standard errors for both Markov-switching GARCH specifications reported in Table 4.1 can be used to assess the statistical significance of the respective model parameters. To this end, we consider the conventional t-statistic the exact finite-sample distribution of which is generally unknown under our estimation setup. However, we can make asymptotic inference by noticing (1) that our ML estimators are asymptotically normally distributed, and (2) that our standard errors constitute (weakly) consistent estimates of the true standard deviations of the ML estimators. Consequently, under the null hypothesis of a single parameter being equal to 0, our t-statistics should converge in distribution towards a standard normal variate implying critical values of 1.6449, 1.9600 and 2.5758 at the 10, 5, and 1% levels, respectively, for the absolute value

[2]In a preliminary analysis we estimated several other Markov-switching GARCH specifications. However, we found that the EGARCH-TGARCH and the TGARCH-TGARCH specifications provide the best fit to our data.

[3]Technical details on the estimation procedure are available upon request.

of the t-statistic (see Greene 2008, Appendix D). Following this reasoning, we find that 35 out of 44 parameters are significant at the 1% level.

An important issue concerns the persistence of volatility shocks. In a conventional single-regime TGARCH(1,1) equation the persistence of volatility shocks is typically measured by the term $\alpha^2(1+c^2)+\beta^2+4\alpha\beta\phi(0)$ (with $\phi(\cdot)$ denoting the probability density function of the standard normal distribution), while in a single-regime EGARCH model the persistence of volatility shocks is simply measured by the parameter β. Generally speaking, we can state that the higher the value of these terms, the longer it takes until a volatility shock dies out. In particular, when the terms are equal to 1 volatility shocks have a permanent effect and the unconditional variances of the respective processes get infinitely large.

In view of these considerations within a single-regime framework, it appears natural to measure the persistence of volatility shocks in our two-regime Markov-switching EGARCH-TGARCH model by the corresponding regime-specific persistence conditions. Unfortunately, matters turn out to be more complicated, since in general it is the interaction between the regime-specific volatility parameters and the (constant) transition probabilities π_1 and π_2 of the regime indicator S_t which determines the variance-stability of a Markov-switching GARCH model.[4]

However, since exact mathematical conditions covering the variance-stability even of the structurally simplest Markov-switching GARCH models are not available in the literature, we are restricted to analyzing the persistence of volatility shocks within each Markov-regime (neglecting the potential impact of the interaction between the regime-specific GARCH parameters and the transition probabilities on the variance stability of the entire Markov-switching model). In Table 4.1 these volatility persistence measures are given in the rows 'Volatility persistence in regime 1' and 'Volatility persistence in regime 2'. Obviously, all these measures are smaller than 1 indicating that shocks do not have a permanent impact on regime-specific unconditional variances.

[4]See for example, Wilfling (2009) and the literature cited there

As a final diagnostic check of the validity of both Markov-switching GARCH specifications we may test the squared (standardized) residuals for first- and higher-order serial correlation by the Ljung-Box-Q-statistic. We performed Ljung-Box-tests for serial correlation of the squared residuals out to various lags for both Markov-switching GARCH specifications and did not find any statistical evidence of serial correlation[5]

Figure 4.2 about here

Volatility and regime-switching results for the EGARCH-TGARCH model

Figure 4.2 displays the conditional variances and regime-1 probabilities for the sample group (left panels) and the control group (right) panels as estimated by our Markov-switching EGARCH-TGARCH specification. We depict two distinct regime-1 probabilities for both groups, namely the *ex ante* regime-1 probabilities $p_{1,t} = \Pr\{S_t = 1|\phi_t\}$ as estimated according to Eq. (A.4.8) of the appendix (thin lines) and the so-called *smoothed regime-1 probabilities* $\Pr\{S_t = 1|\phi_T\}$ which we computed as described in Hamilton (1994).

Figure 4.2 gives evidence that the short-selling restrictions have rather a destabilizing impact than a stabilizing impact on return volatility. The conditional variances have a distinctive peak in both the sample and the control group after the short-selling restrictions have come into operation. The peak in the sample group is even higher in absolute value than in the control group, but the overall volatility level has already been higher in the sample group prior to the prohibition of naked short sales. For the sample group we observe a second clear-cut peak in the conditional variances in March/April 2009. This large volatility increase has no counterpart in the control group. From this we might conjecture (1) that the first peak may have been caused by the outbreak of the financial crisis and (2) that there is no evidence that the short selling restrictions may have stabilized return volatility in our sample group. The second volatility peak in March/April

[5]Details of the autocorrelation tests are available upon request.

2009 rather appears to indicate that the return variances of the sample group have been destabilized.

The regime-1 probabilities in Figure 4.2 indicate two substantial changes in the return dynamics for both the sample and the control group. The first striking regime-switch from regime 2 to regime 1 occurs in September 2006 for the sample group, when international market participants became aware of the first signals of the subprime crisis, and in January/February 2007 for the control group. The second switch from regime 2 to regime 1 occurred in August 2008 for the sample group and in January 2009 for the control group. Obviously, the regime switches in the control group occur later than their counterparts for the sample group. A plausible explanation of this phenomenon is that the financial sector—represented by our sample group—was immediately affected by the financial crisis whereas the nonfinancial enterprises in our control group were affected with a delay. Apart from the switches from regime 2 to regime 1, the regime-1 probabilities for the sample group exhibit a switch back from regime 1 to regime 2 at the beginning of the restriction period. This switch happened precisely on 18 December 2008 when the BaFin announced the first prolongation of the short-selling restrictions.

There are two justifications as to why the aforementioned regime switches are likely to have stemmed from other effects than the short-selling restrictions. First, we observe analogous regime-switches also for the control group, and second we observe an analogous regime switch for the sample group during a period long before the short-selling restrictions came into operation.

There is, however, a clear-cut empirical argument in favor of the hypothesis that the short-selling restrictions might have had a rather destabilizing than a stabilizing effect on the return volatility of the sample group. We base our reasoning on the volatility persistence in both Markov-regimes as estimated in the lower part of Table 4.1. We interpret a high (low) value of the regime-specific volatility persistence as an indicator of low (high) variance stability. For the sample group the volatility persistence in regime 1 is 0.9951 exceeding the corresponding regime-1 volatility persistence of the control group which is 0.9824. In contrast to this, the volatility persistence in regime 2 for the sample group is 0.9063 and thus lower than the regime-2 volatility persistence for the control group which is 0.9185. Consequently, for our sample group the regime switch from regime 2 to regime 1 entails a higher increase in volatility persistence than for our control group indicating a relatively stronger variance destabilization in regime 1 for the sample group as compared to the control group.

Figure 4.3 about here

Volatility and regime-switching results for the TGARCH-TGARCH model

Figure 4.3 displays the regime-1 probabilities and conditional variances of our Markov-Switching TGARCH-TGARCH model. As within our EGARCH-TGARCH framework from Section 4.3.2.1, Figure 4.3 again does not provide any evidence that the short-selling restrictions may have had a stabilizing effect on return volatility. The conditional variances are qualitatively similar to those of the EGARCH-TGARCH model with a clear-cut peak for both groups after the short-selling restrictions have come into operation. The volatility peak in the sample group is again higher than in the control group and the relative increase in the conditional variances of the sample group as compared to the variance increase of the control group is even larger than within the EGARCH-TGARCH framework. Consequently, we are inclined to draw the same conclusions as in Section 4.3.2.1.

For the sample group the regime-1 probabilities in Figure 3 indicate a switch from regime 2 to regime 1 during the period of the short-selling restrictions. There is a switch back to regime 2 shortly before the short-selling restrictions expire. This might indicate that market participants anticipate that there will be no further prolongation of the restriction period. By contrast, the regime probabilities of the control group do not exhibit such distinctive switches. However, during the period of the short-selling restrictions the control group is also in regime 1 most of the time.

Finally, we consider again the volatility persistence in both Markov-regimes which we interpret as a measure of variance stability. According to the lower part of Table 4.1 the regime-1 volatility persistence for the sample group is 0.9852 again (slightly) exceeding the corresponding regime-1 volatility persistence of the control group which is 0.9831. In regime 2 the volatility persistence of the sample group is 0.8733 and hence is substantially lower than the volatility persistence for the control group which is 0.9313. Consequently, the switch from regime 2 to regime 1 during the period of short-selling restrictions again constitutes a transition into a highly destabilized volatility regime. This is particularly true for our sample group for which we find an extreme increase in volatility persistence from 0.8733 to 0.9852.

4.4 Conclusions

In this chapter we analyze the impact of short-selling restrictions on stock-return volatility for the German stock market. To this end, we construct an index from those stocks on which the German Federal Financial Supervisory Authority (BaFin) imposed short-selling constraints during the years 2008 until 2010. As a control group we consider an index constructed from all DAX enterprises that were not subject to short-selling constraints. Estimating two distinct versions of a two-regime Markov-switching asymmetric GARCH model (an EGARCH-TGARCH and a TGARCH-TGARCH specification) for the stock returns of both indices, we observe an overall increase in the conditional variances for the whole German stock market after the collapse of Lehman Brothers. However, we find econometric evidence that the financial crisis is accompanied by an increase in *volatility persistence* and that this effect is particularly pronounced for those stocks that were subject to short-selling constraints. We interpret this finding as evidence of a destabilizing impact of short-selling constraints on stock-return volatility.

A natural line of future research could be an investigation on the impact of short-selling constraints on the trading volume. An analysis of this issue could possibly reveal two aspects. First, market participants are likely to circumvent the short-selling restriction through the evasion in derivatives and a switch to alternative market places in which no short-selling restrictions are in force. Second, short-selling constraints might entail substantial changes in liquidity. Apart from that a quantitative analysis as to what extent short-selling constraints on selected shares can erode investors' confidence in the affected enterprises and thus induce stock-price declines appears to be indicated.

Appendix

In this appendix we construct the log-likelihood function for the Markov-switching EGARCH-TGARCH model established in Section 2. The log-likelihood function for the TGARCH-TGARCH model may be similarly derived by inserting the corresponding functional form in the GARCH equation (A.4.10) below.

The conditional probability distribution of r_{t+1} is given by

$$r_{t+1}|\phi_t \sim \begin{cases} N(\lambda, h_{1,t}) & \text{with probability} \quad p_{1,t} \\ N(\lambda, h_{2,t}) & \text{with probability} \quad (1-p_{1,t}) \end{cases}, \tag{A.4.1}$$

where ϕ_t defines the information set as of date t and $p_{i,t} \equiv \Pr\{S_t = i|\phi_t\}$ denotes the so-called *ex-ante* probability of being in regime i at date t. It is instructive to note that the information set ϕ_t basically coincides with the return path $\tilde{r}_t = \{r_t, r_{t-1}, \ldots\}$, but does not contain the path of the unobservable regime indicator S_t. The corresponding probability density function has the form

$$\begin{aligned} f(r_{t+1}|\phi_t) &= \sum_{i=1}^{2} f(r_{t+1}, S_t = i|\phi_t) \\ &= \sum_{i=1}^{2} \Pr(S_t = i|\phi_t) \cdot f(r_{t+1}|S_t = i, \phi_t) \\ &= \sum_{i=1}^{2} p_{i,t} \cdot f(r_{t+1}|S_t = i, \phi_t). \end{aligned} \tag{A.4.2}$$

Since the regime indicator S_t follows a first-order Markov process the *ex-ante* probability $p_{i,t}$ depends only on S_{t-1}. Using the *Theorem of Total Probabilities*, we obtain

$$p_{i,t} = \sum_{j=1}^{2} \Pr(S_t = i|S_{t-1} = j, \tilde{r}_t) \Pr(S_{t-1} = j|\tilde{r}_t). \tag{A.4.3}$$

The first probability $\Pr(S_t = i|S_{t-1} = j, \tilde{r}_t)$ on the right-hand side of (A.4.3) does not depend on $\tilde{r}_t$. Thus, we can insert the probabilities specified in Eq. (4.7) in Eq. (A.4.3) and obtain

$$\begin{aligned} p_{1,t} &= \pi_1 \cdot \Pr(S_{t-1} = 1|\tilde{r}_t) + (1 - \pi_2) \cdot \Pr(S_{t-1} = 2|\tilde{r}_t) \\ &= \pi_1 \cdot \Pr(S_{t-1} = 1|\tilde{r}_t) + (1 - \pi_2) \cdot (1 - \Pr(S_{t-1} = 1|\tilde{r}_t)), \end{aligned} \tag{A.4.4}$$

and analogously

$$p_{2,t} = \pi_2 \cdot (1 - \Pr(S_{t-1} = 1|\tilde{r}_t)) + (1 - \pi_1) \cdot \Pr(S_{t-1} = 1|\tilde{r}_t). \tag{A.4.5}$$

The remaining probability $\Pr(S_{t-1} = 1|\tilde{r}_t)$ in the Eqs. (A.4.4) and (A.4.5) can be written as a function of $p_{1,t-1} = \Pr(S_{t-1} = 1|\tilde{r}_{t-1})$. To this end, we apply *Bayes' Formula* yielding

$$\begin{aligned} \Pr(S_{t-1} = 1|\tilde{r}_t) &= \Pr(S_{t-1} = 1|r_t, \tilde{r}_{t-1}) \\ &= \frac{f(r_t|S_{t-1} = 1, \tilde{r}_{t-1}) \Pr(S_{t-1} = 1, \tilde{r}_{t-1})}{\sum_{i=1}^{2} f(r_t|S_{t-1} = i, \tilde{r}_{t-1}) \Pr(S_{t-1} = i, \tilde{r}_{t-1})}. \end{aligned} \tag{A.4.6}$$

Expanding the ratio on the right-hand side of Eq. (A.4.6), we obtain

$$\begin{aligned}\Pr(S_{t-1}=1|\tilde{r}_t) &= \frac{f(r_t|S_{t-1}=1,\tilde{r}_{t-1})\Pr(S_{t-1}=1|\tilde{r}_{t-1})}{\sum_{i=1}^{2} f(r_t|S_{t-1}=i,\tilde{r}_{t-1})\Pr(S_{t-1}=i|\tilde{r}_{t-1})} \\ &= \frac{f(r_t|S_{t-1}=1,\tilde{r}_{t-1})p_{1,t-1}}{\sum_{i=1}^{2} f(r_t|S_{t-1}=i,\tilde{r}_{t-1})p_{i,t-1}} \\ &= \frac{g_{1,t-1}\cdot p_{1,t-1}}{\sum_{i=1}^{2} g_{i,t-1}\cdot p_{i,t-1}}, \end{aligned} \tag{A.4.7}$$

where, for ease of notation, we have defined $g_{i,t-1} \equiv f(r_t|S_{t-1}=i,\tilde{r}_{t-1}) = f(r_t|S_{t-1}=i,\phi_{t-1})$. Using Eq. (A.4.7), we are now able to calculate the *ex-ante* probability $p_{1,t}$ by inserting Eq. (A.4.7) in Eq. (A.4.4):

$$\begin{aligned}p_{1,t} &= \pi_1\cdot\frac{g_{1,t-1}p_{1,t-1}}{g_{1,t-1}p_{1,t-1}+g_{2,t-1}(1-p_{1,t-1})} + (1-\pi_2)\cdot\left[1-\frac{g_{1,t-1}p_{1,t-1}}{g_{1,t-1}p_{1,t-1}+g_{2,t-1}(1-p_{1,t-1})}\right] \\ &= \pi_1\cdot\frac{g_{1,t-1}p_{1,t-1}}{g_{1,t-1}p_{1,t-1}+g_{2,t-1}(1-p_{1,t-1})} + (1-\pi_2)\cdot\frac{g_{2,t-1}(1-p_{1,t-1})}{g_{1,t-1}p_{1,t-1}+g_{2,t-1}(1-p_{1,t-1})}\end{aligned} \tag{A.4.8}$$

Next, we address the exact form of the conditional density f appearing in the Eqs. (A.4.2) and (A.4.8). Since we assume conditional normality f is given as

$$f(r_{t+1}|S_t=i,\phi_t) = \frac{1}{\sqrt{2\pi h_{i,t}}}\exp\left\{-\frac{(r_{t+1}-\lambda)^2}{2h_{i,t}}\right\}. \tag{A.4.9}$$

The variance $h_{i,t}$ depends on the explicit functional form of the GARCH equation. In the case of our EGARCH-TGARCH specification we set $\mu_1 = 0, \nu_1 = 1, \mu_2 = 1$ and $\nu_2 = 1$ in Eq. (4.2) of the main text and obtain

$$h_{i,t} = \begin{cases} \left[\exp\left\{\omega_1+\alpha_1 f_1(\delta_t^{(1)})+\beta_1\ln\left(\sqrt{h_{t-1}^{(1)}}\right)\right\}\right]^2 & \text{for} \quad S_t=1 \\ \left[\omega_2+\alpha_2\sqrt{h_{t-1}^{(2)}}f_2(\delta_t^{(2)})+\beta_2\sqrt{h_{t-1}^{(2)}}\right]^2 & \text{for} \quad S_t=2 \end{cases} \tag{A.4.10}$$

with appropiately defined parameters $\omega_i, \alpha_i, \beta_i$.

It is obvious from Eq. (A.4.10) that for the calculation of regime-specific variances $h_{i,t}$ we need the aggregated variances and shock terms $h_{t-1}^{(i)}$ and $\delta_t^{(i)}$ the calculation of which we base on the probabilities $p_{1,t-1}^{(i)}$ as described in the main text. Using *Bayes' Formula* once more, we obtain these probabilities as

$$\begin{aligned} p_{1,t-1}^{(i)} &= \Pr(S_{t-1} = 1 | \tilde{r}_{t-1}, S_t = i) \\ &= \frac{\Pr(S_t = i | \tilde{r}_{t-1}, S_{t-1} = 1) \Pr(S_{t-1} = 1 | \tilde{r}_{t-1})}{\Pr(S_t = i | \tilde{r}_{t-1})} \\ &= \frac{\Pr(S_t = i | \tilde{r}_{t-1}, S_{t-1} = 1) \cdot p_{1,t-1}}{\Pr(S_t = i | \tilde{r}_{t-1})} \end{aligned} \tag{A.4.11}$$

with $p_{1,t-1}$ as given in Eq. (A.4.8). Using the *Theorem of Total Probabilities*, we write the denominator in Eq. (A.4.11) as

$$\begin{aligned} \Pr(S_t = i | \tilde{r}_{t-1}) &= \Pr(S_t = i | \tilde{r}_{t-1}, S_{t-1} = 1) \cdot p_{1,t-1} \\ &\quad + \Pr(S_t = i | \tilde{r}_{t-1}, S_{t-1} = 2) \cdot (1 - p_{1,t-1}). \end{aligned} \tag{A.4.12}$$

To calculate the probability on the left-hand of Eq. (A.4.11) we need the two probabilities $\Pr(S_t = i | \tilde{r}_{t-1}, S_{t-1} = 1)$ and $\Pr(S_t = i | \tilde{r}_{t-1}, S_{t-1} = 2)$. These probabilities coincide with those defined in Eq. (4.7) of the main text, since the Markov process is independent of the return process. Now, inserting the Eqs. (4.7) and (A.4.12) in Eq. (A.4.11), we obtain

$$\begin{aligned} p_{1,t-1}^{(1)} &= \frac{\pi_1 \cdot p_{1,t-1}}{\pi_1 \cdot p_{1,t-1} + (1 - \pi_2) \cdot p_{2,t-1}}, \\ p_{2,t-1}^{(1)} &= 1 - p_{1,t-1}^{(1)}, \\ p_{1,t-1}^{(2)} &= \frac{(1 - \pi_1) \cdot p_{1,t-1}}{(1 - \pi_1) \cdot p_{1,t-1} + \pi_2 \cdot p_{2,t-1}}, \\ p_{2,t-1}^{(2)} &= 1 - p_{1,t-1}^{(2)}. \end{aligned} \tag{A.4.13}$$

Finally, we use the recursive structures developed so far to construct the log-likelihood function of our Markov-switching model defined in the Eqs. (4.1) to (4.7). The general form of the likelihood function is

$$L(\mathbf{\Theta}) = f(r_t, \ldots, r_1; \mathbf{\Theta})$$

with the vector $\mathbf{\Theta}$ containing all model parameters. Writing this joint distribution of the returns as a product of conditional densities, we obtain

$$L(\mathbf{\Theta}) = \prod_{t=1}^{T} f(r_t|\tilde{r}_{t-1}; \mathbf{\Theta})$$

for which we define the starting term as $f(r_1|\tilde{r}_0; \mathbf{\Theta}) \equiv f(r_1; \mathbf{\Theta})$. Taking the logarithm of $L(\mathbf{\Theta})$ and inserting (the lagged form of) Eq. (A.4.1), we obtain the log-likelihood function as

$$\begin{aligned} \ell(\mathbf{\Theta}) \equiv \log[L(\mathbf{\Theta})] &= \sum_{t=1}^{T} \log\left[f(r_t|\tilde{r}_{t-1}; \mathbf{\Theta})\right] \\ &= \sum_{t=1}^{T} \log\left[\sum_{j=1}^{2} f(r_t|S_{t-1} = j, \tilde{r}_{t-1}; \mathbf{\Theta}) \cdot p_{j,t-1}\right]. \end{aligned} \tag{A.4.14}$$

Figures

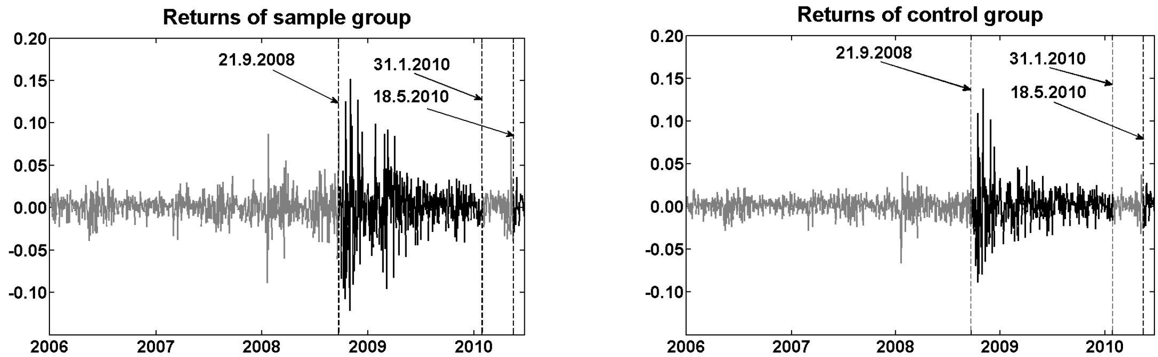

Figure 4.1: Daily index returns for the sample and the control group

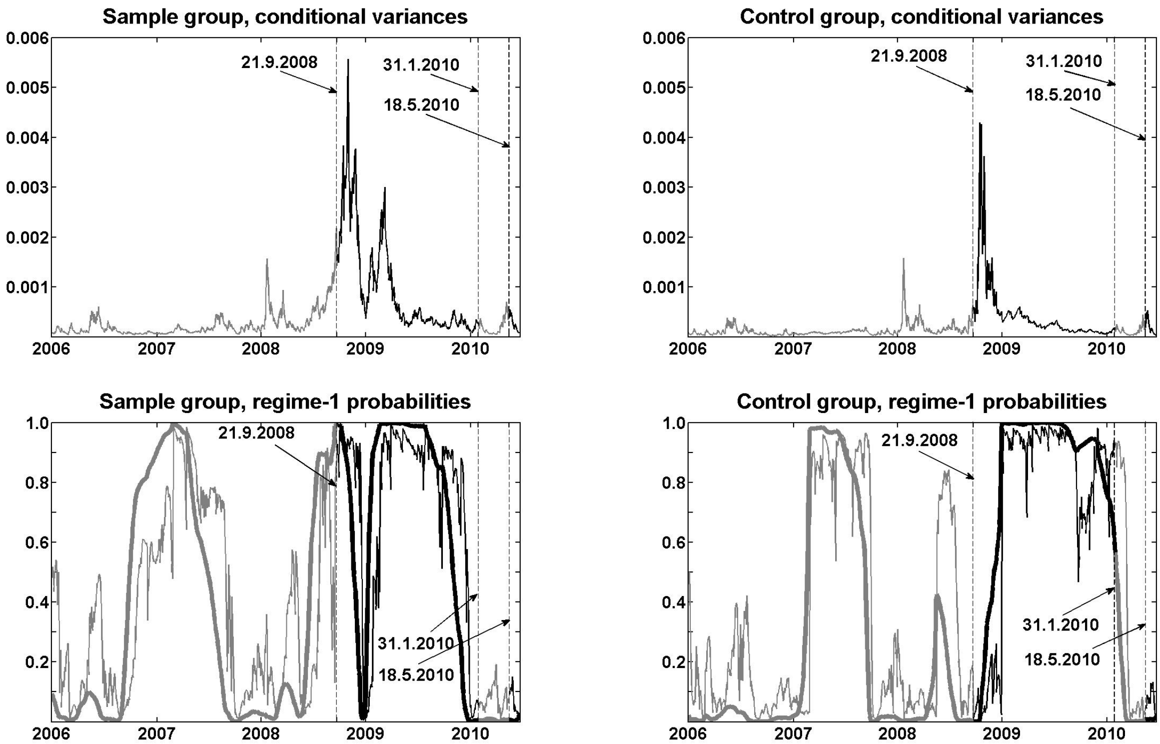

Figure 4.2: Conditional variances and regime-1 probabilities for the sample and the control group estimated by the Markov-switching EGARCH-TGARCH specification

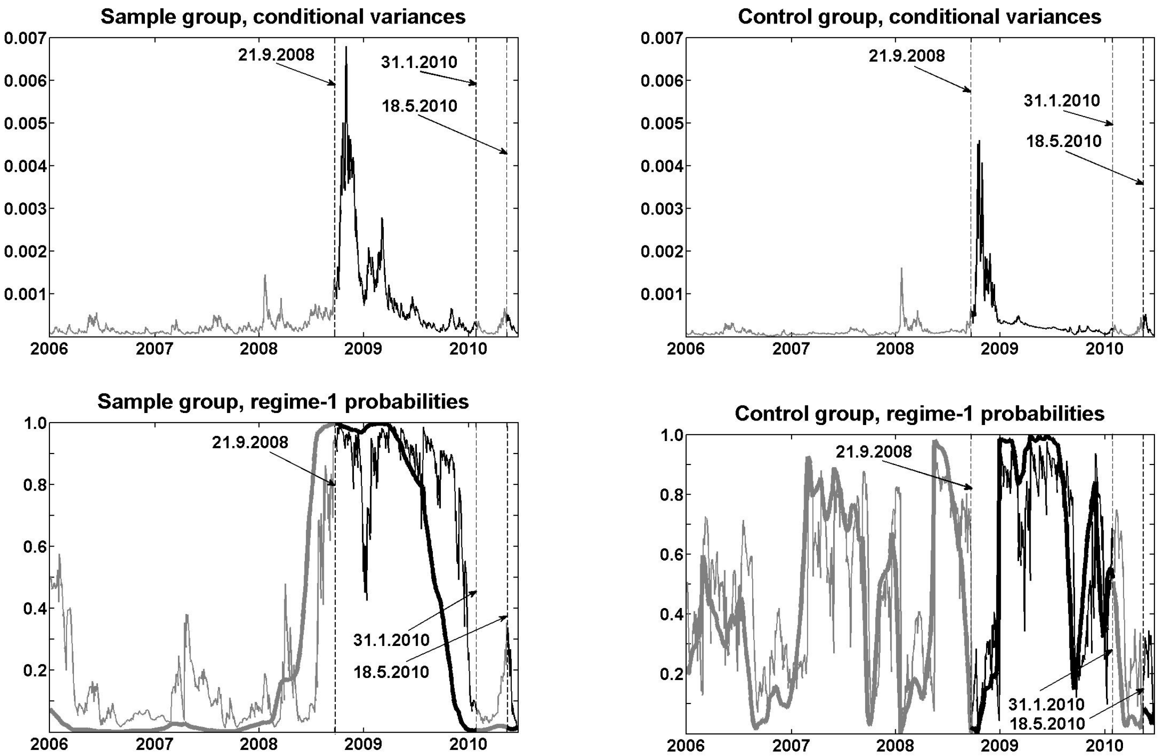

Figure 4.3: Conditional variances and regime-1 probabilities for the sample and the control group estimated by the Markov-switching TGARCH-TGARCH specification

Table

	Sample group EGARCH-TGARCH	Control group EGARCH-TGARCH	Sample group TGARCH-TGARCH	Control group TGARCH-TGARCH
μ_1	0.0000	0.0000	1.0000	1.0000
μ_2	1.0000	1.0000	1.0000	1.0000
ν_1	1.0000	1.0000	1.0000	1.0000
ν_2	1.0000	1.0000	1.0000	1.0000
λ	−0.0000	0.0002	0.0000	0.0004
	(0.0015)	(0.0067)	(0.0001)	(0.0015)
ω_1	0.0158***	−0.0336	0.0004***	0.0001
	(0.0012)	(0.0234)	(0.0000)	(0.0002)
ω_2	0.0007	0.0006	0.0010***	0.0006
	(0.0010)	(0.0010)	(0.0000)	(0.0004)
α_1	−0.0409***	−0.0528***	0.0672***	0.0118***
	(0.0014)	(0.0171)	(0.0002)	(0.0006)
α_2	0.0991***	0.1418***	0.0935***	0.1443***
	(0.0050)	(0.0420)	(0.0003)	(0.0041)
β_1	0.9951***	0.9824***	0.9358***	0.9820***
	(0.0000)	(0.0000)	(0.0023)	(0.0134)
β_2	0.8661***	0.8309***	0.8535***	0.8350***
	(0.0559)	(0.0214)	(0.0040)	(0.0491)
π_1	0.9941***	0.9956***	0.9958***	0.9845***
	(0.0001)	(0.0238)	(0.0000)	(0.0001)
π_2	0.9946***	0.9963***	0.9982***	0.9851***
	(0.0001)	(0.0299)	(0.0000)	(0.0053)
c_1	−1.7014***	−0.6443***	1.0000***	1.0000***
	(0.0009)	(0.0444)	(0.0027)	(0.0721)
c_2	0.9778***	0.9975***	1.0000***	1.0000***
	(0.1598)	(0.3074)	(0.0035)	(0.0202)
Volatility persistence in regime 1	0.9951	0.9824	0.9852	0.9831
Volatility persistence in regime 2	0.9063	0.9185	0.8733	0.9313

Notes: Estimates for parameters from the Eqs. (1) to (10). Standard errors are in parentheses. *,** and *** denote statistical significance at 10, 5 and 1% levels, respectively.

Table 4.1: Estimates of EGARCH-TGARCH model for the sample group and the control group

Chapter 5

Conclusions

The three essays in this thesis investigate policy shifts and Markov-switching in financial markets. Along these lines we (a) analyze a change of the exchange-rate system and the impact on interest rate derivatives, (b) develop a class of two-regime Markov-switching heteroskedasticity models, and (c) study short-selling constraints as a measure to stabilize financial markets.

In chapter two we valuate options on zero-coupon bonds in a continuous-time modeling in the run-up to a currency union. For this purpose we establish interest rate dynamics in a situation based on (a) a Vasicek-dynamic of the foreign short rate, (b) a dynamic of the interest rate differentials relying on a Brownian-motion of the exchange-rate fundamental, and (c) a schedule that is similar to Stage III of the EMU. Taking into account the specific interest-rate dynamics, we obtain closed-form pricing formulas for zero-coupon bonds and options on zero-coupon bonds in this situation. To accomplish this we employ the equivalent martingale measure, a classical option-pricing technique. In our simulation study we illustrate that neglecting previously described interest rate dynamics can generate substantial option-pricing errors. We compute the prices of eleven correctly priced products and use these prices to calibrate the correct interest-rate dynamic as well as the Vasicek-dynamic, as this is the correct dynamic if the currency union would never happen. Based on these calibrated dynamics we compare prices of currently issued options. The observed prices can differ remarkably, depending on the remaining time until the exchange-rate becomes fixed, the time to maturity of the option, and the time to maturity of the underlying bond of the option. Obviously, our pricing formulas may be used to price more complex contingent claims like interest-rate floors (caps), which are among the most traded of all interest-rate derivatives, so that our results should be of high value for traders in all sorts of financial markets located in the upcoming EMU accession countries. A line of future research is the impact of modifications of our model for our option-valuation dynamics. Possible modifications are a different interest rate model for the foreign country or active policy regimes modeled by Ornstein-Uhlenbeck processes and Brownian-bridge specifications for the fundamental process.

In chapter three we extend the existing regime-switching literature for conditional heteroskedasticity models. We combine Klaasens' (2002) improved version of Grays (1996) method to circumvent the problem of path-dependency with Hentschels (1995) approach of nesting various well-known single-regime GARCH models to achieve a very flexible two-regime Markov-switching model class. This model class is able to describe time series data with different specifications in both regimes such as EGARCH-GARCH models. In addition to the model features implied by a Markov-switching approach and the properties of symmetric and asymmetric autoregressive conditional heteroskedasticity model, our new model class covers GARCH-in-Mean specifications as well as time-varying transition probabilities in its modeling approach. We apply our model to daily excess returns of the DAX for the period between January 2000 and December 2009. In this dataset we find broad evidence for Markov-switching in the German stock market with substantially differing volatility structures across both Markov-regimes. Furthermore, we illustrate that for this dataset our model unambiguously outperforms all alternative Markov-switching GARCH models existing in the literature so far. In the technical appendix of chapter three we reveal the maximum likelihood estimation algorithm for our model. The corresponding program codes are given in the programming appendix B of this thesis. Discovering stationarity conditions for regime-switching GARCH models in general and for our model in particular is a possible line of future research. Furthermore, alternative estimation techniques like Bayesian Markov-Chain Monte-Carlo (MCMC) algorithms may be another road to circumvent the problem of path dependency (see Bauwens et al., 2010).

In chapter four we analyze the impact of short-selling restrictions as a measure to stabilize financial markets. In order to accomplish this we apply special versions of our model from chapter three to specific indices of stocks. The stocks in the first index are subject to short-selling restrictions, whereas the stocks in the second index are not restricted. We use data from Germany, since the German Federal Financial Supervisory Authority (BaFin) enacted short-selling constraints for a sufficiently long duration to achieve reliable results. As expected, we observe an overall increase in the conditional variances for the whole German stock market after the collapse of Lehman Brothers. Interestingly, we find evidence that the financial crisis is accompanied by an increase in volatility persistence, particulary for those stocks that are affected by short-selling constraints. These results cast doubts on whether short-selling constraints are a suitable measure to stabilize financial markets. An interesting path of future research is an investigation on the impact of short-selling constraints on trading volume. This could indicate circumvention of regulated markets through evasion in derivatives. Additionally, extent short-selling constraints on selected shares might erode the confidence of investors for the concerned enterprises and thus induce stock price declines.

In this thesis we give evidence that models should be capable to incorporate fundamental changes in the underlying market. Neglecting fundamental changes lead to mispricing or improper regulatory decisions. Particularly in financial markets, as they are crucial for the whole economy, reliable models are essential.

References

Bauwens, L., Preminger, A., Rombouts, J.V.K., 2010. Theory and inference for a Markov switching GARCH model. Econometrics Journal 13, 218-244.

Björk, T., 2004. Arbitrage Theory in Continuous Time. Oxford University Press, New York.

Black, F., 1976. Studies of stock price volatility changes. Proceedings of the 1976 Meetings of the Business and Economics Statistics Section, American Statistical Association, 177-181.

Bohl, M.T., Salm, C.A., Wilfling, B., 2010. Do individual index futures investors destabilize the underlying spot market? Journal of Futures Markets, forthcoming.

Bollerslev, T., 1986. Generalized autoregressive conditional heteroskedasticity. Journal of Econometrics 31, 307-328.

Box, G.E.P., Cox, D.R., 1964. An analysis of transformations. Journal of the Royal Statistical Society, Series B 26, 211-252.

Cai, J., 1994. A Markov model of unconditional variance in ARCH. Journal of Business and Economic Statistics 12, 309-316.

Christie, A.A., 1982. The stochastic behavior of common stock variances: value, leverage and interest rate effects. Journal of Financial Economics 10, 407-432.

Cox, J.C., Ingersoll, J.E., Ross, S.A., 1985. A Theory of the Term Structure of Interest Rates. Econometrica 53, 385-407.

Davies, R.B., 1987. Hypothesis testing when a nuisance parameter is present only under the alternative. Biometrika 64, 247-254.

De Grauwe, P., Dewachter, H., Veestraeten, D., 1999. Price Dynamics Under Stochastic Process Switching: Some Extensions and an Application to EMU. Journal of International Money and Finance 18 (2), 195-224.

Ding, Z., Granger, C.W.J., Engle, R.F., 1993. A long memory property of stock market returns and a new model. Journal of Empirical Finance 1, 83-106.

Elliot, R.J., Kopp, P.E., 2005. Mathematics of Financial Markets, 2nd Edition. Springer Science+Business Media Inc., New York.

Engle, R.F., 1982. Autoregressive conditional heteroskedasticity with estimates of the variance of United Kingdom inflation. Econometrica 50, 987-1007.

Engle, R.F., Lilien, D.M., Robins, R.P., 1987. Estimating time varying risk premia in the term structure: the ARCH-M model. Econometrica 55, 391-407.

Engle, R.F., Ng, V.K., 1993. Measuring and testing the impact of news volatility. Journal of Finance 48, 1749-1778.

Gelman, S., Wilfling, B., 2009. Markov-switching in target stocks during takeover bids. Journal of Empirical Finance 16, 745-758.

Glosten, L.R., Jagannathan, R., Runkle, D.E., 1993. On the relation between expected value and the volatility of the nominal excess return on stocks. Journal of Finance 48, 1779-1801.

Gray, S.F., 1996. Modeling the conditional distribution of interest-rates as a regime-switching process. Journal of Financial Economics 3, 59-71.

Greene, W.H., 2008. Econometric Analysis. Pearson Education, New Jersey.

Hamilton, J.D., 1994. Time Series Analysis. Princeton University Press, Princeton.

Hamilton, J.D., Susmel, R., 1994. Autoregressive conditional heteroskedasticity and changes in regime. Journal of Econometrics 64, 307-333.

Henry, Ó.T., 2009. Regime switching in the relationship between equity returns and short-term interest rates in the UK. Journal of Banking and Finance 33, 405-414.

Hentschel, L., 1995. All in the family: Nesting Symmetric and Asymmetric GARCH models. Journal of Financial Economics 39, 71-104.

Higgins, M.L., Bera, A.K., 1992. A class of nonlinear ARCH models. International Economic Review 33, 137-158.

Ho, T.S.Y., Lee, S.-B., 1986. Term Structure Movements and Pricing Interest Rate Contingent Claims. Journal of Finance 41, 1011-1029.

Hull, J., White, A., 1994. Numerical Procedures for Implementing Term Structure Models II: Two-Factor Models. Journal of Derivatives 2, 37-48.

Jamshidian, F., 1989. An Exact Bond Option Formula. Journal of Finance 44, 205-209.

Klaassen, F., 2002. Improving GARCH volatility forecasts with regime-switching GARCH. Empirical Economics 27, 363-394.

Krugman, P., 1991. Target Zones and Exchange Rate Dynamics. Quarterly Journal of Economics 106, 669-682.

Nelson, D.B., 1990. Stationarity and persistence in the GARCH(1,1) model. Econometric Theory 6, 318-334.

Nelson, D.B., 1991. Conditional heteroskedasticity in asset returns: a new approach. Econometrica 59, 347-370.

Reher, G., Wilfling, B., 2010. Markov-switching in conditional heteroskedasticity models: a unifying approach with an application to the German stock market. Working Paper, Department of Economics, University of Münster.

Schwert, G.W., 1989. Why does stock market volatility change over time? Journal of Finance 44, 1115-1153.

Sondermann, D., Trede, M, Wilfling, B., 2010. Estimating the Degree of Interventionist Policies in the Run-Up to EMU. Applied Economics, forthcoming.

Sutherland, A., 1995. State- and Time-Contingent Switches of Exchange Rate Regime. Journal of International Economics 38, 361-374.

Svensson, L.E.O., 1991a. Target Zones and Interest Rate Variability. Journal of International Economics 31, 27-54.

Svensson, L.E.O., 1991b. The Term Structure of Interest Rate Differentials in a Target Zone: Theory and Swedish Data. Journal of Monetary Economics 28, 87-116.

Vasicek, O., 1977. An Equilibrium Characterization of the Term Structure. Journal of Financial Economics 5, 177-188.

Wilfling, B., 2003. Interest Rate Volatility Prior to Monetary Union under Alternative Pre-Switch Regimes. German Economic Review 4, 433-457.

Wilfling, B., 2009. Volatility regime-switching in European exchange rates prior to monetary unification. Journal of International Money and Finance 28, 240-270.

Wilfling, B., Maennig, W., 2001. Exchange Rate Dynamics in Anticipation of Time-Contingent Regime Switching: Modelling the Effects of a Possible Delay. Journal of International Money and Finance 20 (1), 91-113.

Zakoian, J.M., 1994. Threshold heteroscedastic models. Journal of Economic Dynamics and Control 18, 931-955.

Appendix A

Programming Codes for Chapter 2

The Monte-Carlo simulation in chapter 2 represents a very significant proportion of the academic work made in chapter 2. To make this part of the work transparent and understandable we give the MATLAB-Codes for a specific parameter setup. The programming code of main.m has to be manually changed for other parameter setups in the way the comments in the code show.

At first we give the programming code of the main script-file. This file has to be started to run the Monte-Carlo simulation. The other m-files follow below.

main.m:

```
%%%%% GENERAL SETUP

%Number of iterations in the Monte-Carlo simulation.
mcanzahl=10000;

%Set model parameters for interest rate process (small country,
%Ornstein-Uhlenbeck process).
b=1;
c=0.05;
rsternnull=0.05;
sigma=0.015;

%Set model parameters for interest rate differential process.
sigmaschlange=0.025;
alpha=2.0;
```

```
%Correlation parameter, must be zero. Algorithm in this version is not able
%to work with other values (numerical integration needed, computing time
%for Monte-Carlo simulation would increase significantly).
beta=0.0;

%Fineness of the grid (support points of the time series).
delta=0.01;

%Announcement date of the currency union.
ta=0.5;

%Starting time of the currency union (must be greater than ta).
ts=1.5;

%End of simulation period.
T=2;

%Logarithm of fixed exchange rate.
s=log(1.00);

%Starting value of fundamental process (at announcement date).
knull=0;

%Preparing the grid for the interest rate process (Vasicek model in the big
%economy, gitterou=Grid Ornstein-Uhlenbeck process). We simulate the
%Ornstein Uhlenbeck process via a brownian motion in non-linear time.
gitterlinear=(delta:delta:T);
gitterou=exp(2*b*gitterlinear)-1;

%Preparing the grid for the macroeconomic fundamental process (the interest
%rate differential process).
gitterlinear2=(ta:delta:ts-delta);

%To prepare the algorithm for a correlation between the interest rate process and
%the interest rate differential process we prepare a common grid for both
%processes (not necessary for beta=0, but we want to prepare the model for
%this modification).
%To simulate the brownian motion on the non-linear grid, we need the distance
%between the support points of the process (diffgitter).
gitter=sort([gitterou 0 gitterlinear2]);
diffgitter=diff(gitter);

%Now we assure the identification which support points belong to which
%process (interest rate process or interest rate differential process).
[gitter,index]=sort([gitterou gitterlinear2]);
```

```
indexou=(index<=T/delta).*(1:length(index));
indexou=unique(indexou); indexou=indexou(2:length(indexou));
indexwiener=(index>T/delta).*(1:length(index));
indexwiener=unique(indexwiener);
indexwiener=indexwiener(2:length(indexwiener));

%Number of support points of the whole model.
stuetzstellen=length(diffgitter);

%Number of support points after ts (if our simulation scenario exceeds ts).
if T>ts
    stuetzstellenamende=(T-ts)/delta;
else
    stuetzstellenamende=0;
end

%%%%% SETUP OF OPTIONS AND BONDS

%We model six bond options and five bonds (used to calibrate the wrongly specified
%model).
%Strike prices of  six bond-options (the bond options should be approximately at the money).
K1=0.99;
K2=0.98;
K3=0.96;
K4=0.99;
K5=0.97;
K6=0.98;

%ATTENTION: The following values and the time to maturity of the related bond
%must be manually changed in the option price formulas below!

%Time to maturity of the bonds.
laufzeit1=1/12;
laufzeit2=1/4;
laufzeit3=1/2;
laufzeit4=1;
laufzeit5=2;

%Time to maturity of the options.
optionslaufzeit1=1/12;
optionslaufzeit2=1/12;
optionslaufzeit3=1/12;
optionslaufzeit4=1/4;
optionslaufzeit5=1/4;
optionslaufzeit6=1/2;
```

```
%We use the bonds with time to maturity from above as underlying bonds of
%the options. We now calculate the remaining time to maturity of the bonds at the
%expiration date of the option.

%Time to maturity of the option: 1 month. Time to maturity of the
%underlying bond: 3 month.
restlaufzeit1=1/4-1/12;

%Time to maturity of the option: 1 month. Time to maturity of the
%underlying bond: 6 month.
restlaufzeit2=1/2-1/12;

%Time to maturity of the option: 1 month. Time to maturity of the
%underlying bond: 12 month.
restlaufzeit3=1-1/12;

%Time to maturity of the option: 3 month. Time to maturity of the
%underlying bond: 6 month.
restlaufzeit4=1/2-1/4;

%Time to maturity of the option: 3 month. Time to maturity of the
%underlying bond: 12 month.
restlaufzeit5=1-1/4;

%Time to maturity of the option: 6 month. Time to maturity of the
%underlying bond: 12 month.
restlaufzeit6=1-1/2;

%ATTENTION: The following values and the time to maturity of the related bond
%must be manually changed in the option price formulas below!

%Strike prices of the options that are newly issued to the market.
vergleichsK1=0.915;
vergleichsK2=0.92;
vergleichsK3=0.925;
vergleichsK4=0.93;
vergleichsK5=0.935;
vergleichsK6=0.94;

%Time to maturity of the newly issued options (is held constant to analyze
%the ceteris paribus effect of different strike prices).
vergleichsoptionslaufzeit1=2/12;
vergleichsoptionslaufzeit2=2/12;
vergleichsoptionslaufzeit3=2/12;
vergleichsoptionslaufzeit4=2/12;
vergleichsoptionslaufzeit5=2/12;
```

```
vergleichsoptionslaufzeit6=2/12;

%Time to maturity of the underlying bond of the newly issued options (just
%for information).
%vergleichsbondlaufzeit1=14/12;
%vergleichsbondlaufzeit2=14/12;
%vergleichsbondlaufzeit3=14/12;
%vergleichsbondlaufzeit4=14/12;
%vergleichsbondlaufzeit5=14/12;
%vergleichsbondlaufzeit6=14/12;

%Remaining time to maturity of the bonds at the
%expiration date of the option.
vergleichsrestlaufzeit1=12/12;
vergleichsrestlaufzeit2=12/12;
vergleichsrestlaufzeit3=12/12;
vergleichsrestlaufzeit4=12/12;
vergleichsrestlaufzeit5=12/12;
vergleichsrestlaufzeit6=12/12;

%%%%% STARTING THE CALCULATION (Changes after this point are only necessary
%%%%% if time to maturities were changed or the assumption beta=0 should be
%%%%% cancelled).

%To accelerate the Monte-Carlo simulation we build basic frameworks for
%bond and option price formulas. These frameworks include the parts of the
%formulas that are independent of stochastic processes.

%Basic framework for the bond prices.
Prohlaufzeit1=bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,laufzeit1,ta,ts,T,delta);
Prohlaufzeit2=bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,laufzeit2,ta,ts,T,delta);
Prohlaufzeit3=bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,laufzeit3,ta,ts,T,delta);
Prohlaufzeit4=bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,laufzeit4,ta,ts,T,delta);
Prohlaufzeit5=bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,laufzeit5,ta,ts,T,delta);

%Basic framework for the fundamental process (for all 3 cases: t<ta,
%ta<t<ts, ts<t). Combined with the stochastic elements it will complete the
%bond prices below.
krohmaske1=fallmaske(laufzeit1,ta,ts,T,delta,alpha);
krohmaske2=fallmaske(laufzeit2,ta,ts,T,delta,alpha);
krohmaske3=fallmaske(laufzeit3,ta,ts,T,delta,alpha);
krohmaske4=fallmaske(laufzeit4,ta,ts,T,delta,alpha);
krohmaske5=fallmaske(laufzeit5,ta,ts,T,delta,alpha);

%Basic framework for the critical value of y_0 (conditional on time to
%maturity of the option and remaining time to maturity of the bond on the
```

```
%expiration date). This value is needed to avoid the "max"-term in the
%integration.
krit1=kritmaske(optionslaufzeit1,restlaufzeit1,ta,ts,
                T,delta,alpha,b,K1,sigma,c,s,sigmaschlange);
krit2=kritmaske(optionslaufzeit2,restlaufzeit2,ta,ts,
                T,delta,alpha,b,K2,sigma,c,s,sigmaschlange);
krit3=kritmaske(optionslaufzeit3,restlaufzeit3,ta,ts,
                T,delta,alpha,b,K3,sigma,c,s,sigmaschlange);
krit4=kritmaske(optionslaufzeit4,restlaufzeit4,ta,ts,
                T,delta,alpha,b,K4,sigma,c,s,sigmaschlange);
krit5=kritmaske(optionslaufzeit5,restlaufzeit5,ta,ts,
                T,delta,alpha,b,K5,sigma,c,s,sigmaschlange);
krit6=kritmaske(optionslaufzeit6,restlaufzeit6,ta,ts,
                T,delta,alpha,b,K6,sigma,c,s,sigmaschlange);

%Basic framework for mu_2.
mu_2_1hilf=mumaske(optionslaufzeit1,restlaufzeit1,ta,ts,T,delta,b,c);
mu_2_2hilf=mumaske(optionslaufzeit2,restlaufzeit2,ta,ts,T,delta,b,c);
mu_2_3hilf=mumaske(optionslaufzeit3,restlaufzeit3,ta,ts,T,delta,b,c);
mu_2_4hilf=mumaske(optionslaufzeit4,restlaufzeit4,ta,ts,T,delta,b,c);
mu_2_5hilf=mumaske(optionslaufzeit5,restlaufzeit5,ta,ts,T,delta,b,c);
mu_2_6hilf=mumaske(optionslaufzeit6,restlaufzeit6,ta,ts,T,delta,b,c);

%Additional framework for mu_2. This part will be combined with the
%stochastic component of the fundamental process (K-process).
mu_2_hilfe_bei_k_1=fallmaske2(restlaufzeit1,optionslaufzeit1,ta,ts,T,delta,alpha,b);
mu_2_hilfe_bei_k_2=fallmaske2(restlaufzeit2,optionslaufzeit2,ta,ts,T,delta,alpha,b);
mu_2_hilfe_bei_k_3=fallmaske2(restlaufzeit3,optionslaufzeit3,ta,ts,T,delta,alpha,b);
mu_2_hilfe_bei_k_4=fallmaske2(restlaufzeit4,optionslaufzeit4,ta,ts,T,delta,alpha,b);
mu_2_hilfe_bei_k_5=fallmaske2(restlaufzeit5,optionslaufzeit5,ta,ts,T,delta,alpha,b);
mu_2_hilfe_bei_k_6=fallmaske2(restlaufzeit6,optionslaufzeit6,ta,ts,T,delta,alpha,b);

%Additional framework for mu_2. This part will be combined with the
%stochastic component of the Ornstein-Uhlenbeck process
%(interest rate process in the big economy).
mu_2_hilfe_bei_m_1=fallmaske3(restlaufzeit1,optionslaufzeit1,ta,ts,T,delta,b);
mu_2_hilfe_bei_m_2=fallmaske3(restlaufzeit2,optionslaufzeit2,ta,ts,T,delta,b);
mu_2_hilfe_bei_m_3=fallmaske3(restlaufzeit3,optionslaufzeit3,ta,ts,T,delta,b);
mu_2_hilfe_bei_m_4=fallmaske3(restlaufzeit4,optionslaufzeit4,ta,ts,T,delta,b);
mu_2_hilfe_bei_m_5=fallmaske3(restlaufzeit5,optionslaufzeit5,ta,ts,T,delta,b);
mu_2_hilfe_bei_m_6=fallmaske3(restlaufzeit6,optionslaufzeit6,ta,ts,T,delta,b);

%Values of sigma_2.
sigma_2_1=sqrt(varianz2maske(optionslaufzeit1,restlaufzeit1,ta,ts,
                T,delta,b,alpha,sigma,sigmaschlange));
sigma_2_2=sqrt(varianz2maske(optionslaufzeit2,restlaufzeit2,ta,ts,
```

```
                T,delta,b,alpha,sigma,sigmaschlange));
sigma_2_3=sqrt(varianz2maske(optionslaufzeit3,restlaufzeit3,ta,ts,
                T,delta,b,alpha,sigma,sigmaschlange));
sigma_2_4=sqrt(varianz2maske(optionslaufzeit4,restlaufzeit4,ta,ts,
                T,delta,b,alpha,sigma,sigmaschlange));
sigma_2_5=sqrt(varianz2maske(optionslaufzeit5,restlaufzeit5,ta,ts,
                T,delta,b,alpha,sigma,sigmaschlange));
sigma_2_6=sqrt(varianz2maske(optionslaufzeit6,restlaufzeit6,ta,ts,
                T,delta,b,alpha,sigma,sigmaschlange));

%Values of the covariances. (ATTENTION: If beta is allowed to differ from zero,
%a modification is needed!)
rho_sigma_1_sigma_2_1=kovarianzmaske(optionslaufzeit1,restlaufzeit1,ta,ts,
                        T,delta,b,sigma,sigmaschlange,alpha);
rho_sigma_1_sigma_2_2=kovarianzmaske(optionslaufzeit2,restlaufzeit2,ta,ts,
                        T,delta,b,sigma,sigmaschlange,alpha);
rho_sigma_1_sigma_2_3=kovarianzmaske(optionslaufzeit3,restlaufzeit3,ta,ts,
                        T,delta,b,sigma,sigmaschlange,alpha);
rho_sigma_1_sigma_2_4=kovarianzmaske(optionslaufzeit4,restlaufzeit4,ta,ts,
                        T,delta,b,sigma,sigmaschlange,alpha);
rho_sigma_1_sigma_2_5=kovarianzmaske(optionslaufzeit5,restlaufzeit5,ta,ts,
                        T,delta,b,sigma,sigmaschlange,alpha);
rho_sigma_1_sigma_2_6=kovarianzmaske(optionslaufzeit6,restlaufzeit6,ta,ts,
                        T,delta,b,sigma,sigmaschlange,alpha);

%Define an auxiliary variable d_1. This variable is the quantile of the
%first normal distribution in the option price formula. A basic framework
%is created here (deterministic components for all cases).
d_1hilf1=d1maske(restlaufzeit1,optionslaufzeit1,ta,ts,T,delta,b);
d_1hilf2=d1maske(restlaufzeit2,optionslaufzeit2,ta,ts,T,delta,b);
d_1hilf3=d1maske(restlaufzeit3,optionslaufzeit3,ta,ts,T,delta,b);
d_1hilf4=d1maske(restlaufzeit4,optionslaufzeit4,ta,ts,T,delta,b);
d_1hilf5=d1maske(restlaufzeit5,optionslaufzeit5,ta,ts,T,delta,b);
d_1hilf6=d1maske(restlaufzeit6,optionslaufzeit6,ta,ts,T,delta,b);

%Basic framework for the bond prices that are needed for the newly issued options.
vergleichsProhlaufzeit1=bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,
                                          2/12,ta,ts,T,delta);
vergleichsProhlaufzeit2=bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,
                                          14/12,ta,ts,T,delta);

%Basic framework for the fundamental process. Combined with the stochastic
%elements it will complete the bond prices (for the newly issued options) below.
vergleichskrohmaske1=fallmaske(2/12,ta,ts,T,delta,alpha);
vergleichskrohmaske2=fallmaske(14/12,ta,ts,T,delta,alpha);
```

```
%Basic framework for the critical values of y_0 for the newly issued options.
vergleichskrit1=kritmaske(vergleichsoptionslaufzeit1,vergleichsrestlaufzeit1,ta,ts,
                          T,delta,alpha,b,vergleichsK1,sigma,c,s,sigmaschlange);
vergleichskrit2=kritmaske(vergleichsoptionslaufzeit2,vergleichsrestlaufzeit2,ta,ts,
                          T,delta,alpha,b,vergleichsK2,sigma,c,s,sigmaschlange);
vergleichskrit3=kritmaske(vergleichsoptionslaufzeit3,vergleichsrestlaufzeit3,ta,ts,
                          T,delta,alpha,b,vergleichsK3,sigma,c,s,sigmaschlange);
vergleichskrit4=kritmaske(vergleichsoptionslaufzeit4,vergleichsrestlaufzeit4,ta,ts,
                          T,delta,alpha,b,vergleichsK4,sigma,c,s,sigmaschlange);
vergleichskrit5=kritmaske(vergleichsoptionslaufzeit5,vergleichsrestlaufzeit5,ta,ts,
                          T,delta,alpha,b,vergleichsK5,sigma,c,s,sigmaschlange);
vergleichskrit6=kritmaske(vergleichsoptionslaufzeit6,vergleichsrestlaufzeit6,ta,ts,
                          T,delta,alpha,b,vergleichsK6,sigma,c,s,sigmaschlange);

%Basic framework for mu_2.
vergleichsmu_2_1hilf=mumaske(vergleichsoptionslaufzeit1,vergleichsrestlaufzeit1,ta,ts,
                             T,delta,b,c);
vergleichsmu_2_2hilf=mumaske(vergleichsoptionslaufzeit2,vergleichsrestlaufzeit2,ta,ts,
                             T,delta,b,c);
vergleichsmu_2_3hilf=mumaske(vergleichsoptionslaufzeit3,vergleichsrestlaufzeit3,ta,ts,
                             T,delta,b,c);
vergleichsmu_2_4hilf=mumaske(vergleichsoptionslaufzeit4,vergleichsrestlaufzeit4,ta,ts,
                             T,delta,b,c);
vergleichsmu_2_5hilf=mumaske(vergleichsoptionslaufzeit5,vergleichsrestlaufzeit5,ta,ts,
                             T,delta,b,c);
vergleichsmu_2_6hilf=mumaske(vergleichsoptionslaufzeit6,vergleichsrestlaufzeit6,ta,ts,
                             T,delta,b,c);

%Additional framework for mu_2 (for the newly issued options). This part will be
%combined with the stochastic component of the fundamental process (K-process).
vergleichsmu_2_hilfe_bei_k_1=fallmaske2(vergleichsrestlaufzeit1,
                             vergleichsoptionslaufzeit1,ta,ts,T,delta,alpha,b);
vergleichsmu_2_hilfe_bei_k_2=fallmaske2(vergleichsrestlaufzeit2,
                             vergleichsoptionslaufzeit2,ta,ts,T,delta,alpha,b);
vergleichsmu_2_hilfe_bei_k_3=fallmaske2(vergleichsrestlaufzeit3,
                             vergleichsoptionslaufzeit3,ta,ts,T,delta,alpha,b);
vergleichsmu_2_hilfe_bei_k_4=fallmaske2(vergleichsrestlaufzeit4,
                             vergleichsoptionslaufzeit4,ta,ts,T,delta,alpha,b);
vergleichsmu_2_hilfe_bei_k_5=fallmaske2(vergleichsrestlaufzeit5,
                             vergleichsoptionslaufzeit5,ta,ts,T,delta,alpha,b);
vergleichsmu_2_hilfe_bei_k_6=fallmaske2(vergleichsrestlaufzeit6,
                             vergleichsoptionslaufzeit6,ta,ts,T,delta,alpha,b);

%Additional framework for mu_2 (for the newly issued options). This part will be
%combined with the stochastic component of the Ornstein-Uhlenbeck process
%(interest rate process in the big economy).
```

```
vergleichsmu_2_hilfe_bei_m_1=fallmaske3(vergleichsrestlaufzeit1,
                            vergleichsoptionslaufzeit1,ta,ts,T,delta,b);
vergleichsmu_2_hilfe_bei_m_2=fallmaske3(vergleichsrestlaufzeit2,
                            vergleichsoptionslaufzeit2,ta,ts,T,delta,b);
vergleichsmu_2_hilfe_bei_m_3=fallmaske3(vergleichsrestlaufzeit3,
                            vergleichsoptionslaufzeit3,ta,ts,T,delta,b);
vergleichsmu_2_hilfe_bei_m_4=fallmaske3(vergleichsrestlaufzeit4,
                            vergleichsoptionslaufzeit4,ta,ts,T,delta,b);
vergleichsmu_2_hilfe_bei_m_5=fallmaske3(vergleichsrestlaufzeit5,
                            vergleichsoptionslaufzeit5,ta,ts,T,delta,b);
vergleichsmu_2_hilfe_bei_m_6=fallmaske3(vergleichsrestlaufzeit6,
                            vergleichsoptionslaufzeit6,ta,ts,T,delta,b);

%Values of sigma_2 (for the newly issued options).
vergleichssigma_2_1=sqrt(varianz2maske(vergleichsoptionslaufzeit1,
                vergleichsrestlaufzeit1,ta,ts,T,delta,b,alpha,sigma,sigmaschlange));
vergleichssigma_2_2=sqrt(varianz2maske(vergleichsoptionslaufzeit2,
                vergleichsrestlaufzeit2,ta,ts,T,delta,b,alpha,sigma,sigmaschlange));
vergleichssigma_2_3=sqrt(varianz2maske(vergleichsoptionslaufzeit3,
                vergleichsrestlaufzeit3,ta,ts,T,delta,b,alpha,sigma,sigmaschlange));
vergleichssigma_2_4=sqrt(varianz2maske(vergleichsoptionslaufzeit4,
                vergleichsrestlaufzeit4,ta,ts,T,delta,b,alpha,sigma,sigmaschlange));
vergleichssigma_2_5=sqrt(varianz2maske(vergleichsoptionslaufzeit5,
                vergleichsrestlaufzeit5,ta,ts,T,delta,b,alpha,sigma,sigmaschlange));
vergleichssigma_2_6=sqrt(varianz2maske(vergleichsoptionslaufzeit6,
                vergleichsrestlaufzeit6,ta,ts,T,delta,b,alpha,sigma,sigmaschlange));

%Values of the covariances (for the newly issued options). (ATTENTION: If beta is
%allowed to differ from zero, a modification is needed!)
vergleichsrho_sigma_1_sigma_2_1=kovarianzmaske(vergleichsoptionslaufzeit1,
                vergleichsrestlaufzeit1,ta,ts,T,delta,b,sigma,sigmaschlange,alpha);
vergleichsrho_sigma_1_sigma_2_2=kovarianzmaske(vergleichsoptionslaufzeit2,
                vergleichsrestlaufzeit2,ta,ts,T,delta,b,sigma,sigmaschlange,alpha);
vergleichsrho_sigma_1_sigma_2_3=kovarianzmaske(vergleichsoptionslaufzeit3,
                vergleichsrestlaufzeit3,ta,ts,T,delta,b,sigma,sigmaschlange,alpha);
vergleichsrho_sigma_1_sigma_2_4=kovarianzmaske(vergleichsoptionslaufzeit4,
                vergleichsrestlaufzeit4,ta,ts,T,delta,b,sigma,sigmaschlange,alpha);
vergleichsrho_sigma_1_sigma_2_5=kovarianzmaske(vergleichsoptionslaufzeit5,
                vergleichsrestlaufzeit5,ta,ts,T,delta,b,sigma,sigmaschlange,alpha);
vergleichsrho_sigma_1_sigma_2_6=kovarianzmaske(vergleichsoptionslaufzeit6,
                vergleichsrestlaufzeit6,ta,ts,T,delta,b,sigma,sigmaschlange,alpha);

%Values of the deterministic components of d_1 (for the newly issued options).
vergleichsd_1hilf1=d1maske(vergleichsrestlaufzeit1,vergleichsoptionslaufzeit1,
                            ta,ts,T,delta,b);
vergleichsd_1hilf2=d1maske(vergleichsrestlaufzeit2,vergleichsoptionslaufzeit2,
```

```
                             ta,ts,T,delta,b);
vergleichsd_1hilf3=d1maske(vergleichsrestlaufzeit3,vergleichsoptionslaufzeit3,
                             ta,ts,T,delta,b);
vergleichsd_1hilf4=d1maske(vergleichsrestlaufzeit4,vergleichsoptionslaufzeit4,
                             ta,ts,T,delta,b);
vergleichsd_1hilf5=d1maske(vergleichsrestlaufzeit5,vergleichsoptionslaufzeit5,
                             ta,ts,T,delta,b);
vergleichsd_1hilf6=d1maske(vergleichsrestlaufzeit6,vergleichsoptionslaufzeit6,
                             ta,ts,T,delta,b);

%Start of the Monte-Carlo simulation.
for durchlaeufe=1:mcanzahl

%Simulate two brownian motions. The first one is needed for the Ornstein-Uhlenbeck
%process (interest rate process in the big economy). The second one is needed
%for the fundamental process respective the interest rate differential. The
%second process is simulated between ta and ts.
wiener1=cumsum(sqrt(diffgitter).*normrnd(0,1,1,stuetzstellen));
wiener2=wienerprozess(delta,1,ts-ta-delta);

%Interest rate process in the big economy. The command wiener1(indexou)
%gets the correct point of the time transformed brownian motion.
rstern=(rsternnull-c)*exp(-b*gitterlinear)+c+sigma/sqrt(2*b)*
        exp(-b*gitterlinear).*wiener1(indexou);

%Calculation of the fundamental process from 0 to T (the process is different
%from zero only from ta to ts).
k=[knull (sigmaschlange*(beta*(wiener1(indexwiener(2:length(indexwiener)))
           -wiener1(indexwiener(1)))+sqrt(1-beta^2)*wiener2))];
khilf=[zeros(1,ta/delta-1) k zeros(1,stuetzstellenamende+1)];

%Bubble free solution of the interest rate differential process.
ID=[zeros(1,ta/delta-1) (exp((gitterlinear2-ts)/alpha).*(s-k)/alpha)
zeros(1,stuetzstellenamende+1)];

%Interest rate process in the small economy (sum of interest rate in the
%big economy and interest rate differential).
r=rstern+ID;

%Put the frameworks above and the random processes together. First we
%calculate the price pathes of the bonds (at every point in time the time
%to maturity stays constant).
Plaufzeit1=Prohlaufzeit1.*exp(-1/b*rstern*(1-exp(-b*laufzeit1))+khilf.*krohmaske1);
Plaufzeit2=Prohlaufzeit2.*exp(-1/b*rstern*(1-exp(-b*laufzeit2))+khilf.*krohmaske2);
Plaufzeit3=Prohlaufzeit3.*exp(-1/b*rstern*(1-exp(-b*laufzeit3))+khilf.*krohmaske3);
Plaufzeit4=Prohlaufzeit4.*exp(-1/b*rstern*(1-exp(-b*laufzeit4))+khilf.*krohmaske4);
```

```
Plaufzeit5=Prohlaufzeit5.*exp(-1/b*rstern*(1-exp(-b*laufzeit5))+khilf.*krohmaske5);

%Second we need some precalculations for the options price formula. We calculate mu_2.
mu_2_1=mu_2_1hilf+rstern.*mu_2_hilfe_bei_m_1+khilf.*mu_2_hilfe_bei_k_1;
mu_2_2=mu_2_2hilf+rstern.*mu_2_hilfe_bei_m_2+khilf.*mu_2_hilfe_bei_k_2;
mu_2_3=mu_2_3hilf+rstern.*mu_2_hilfe_bei_m_3+khilf.*mu_2_hilfe_bei_k_3;
mu_2_4=mu_2_4hilf+rstern.*mu_2_hilfe_bei_m_4+khilf.*mu_2_hilfe_bei_k_4;
mu_2_5=mu_2_5hilf+rstern.*mu_2_hilfe_bei_m_5+khilf.*mu_2_hilfe_bei_k_5;
mu_2_6=mu_2_6hilf+rstern.*mu_2_hilfe_bei_m_6+khilf.*mu_2_hilfe_bei_k_6;

%Third we calculate the values of the two quantiles in the option price
%formula. The second quantile is d_2.
d_2_1=(krit1-mu_2_1+rho_sigma_1_sigma_2_1)./sigma_2_1;
d_2_2=(krit2-mu_2_2+rho_sigma_1_sigma_2_2)./sigma_2_2;
d_2_3=(krit3-mu_2_3+rho_sigma_1_sigma_2_3)./sigma_2_3;
d_2_4=(krit4-mu_2_4+rho_sigma_1_sigma_2_4)./sigma_2_4;
d_2_5=(krit5-mu_2_5+rho_sigma_1_sigma_2_5)./sigma_2_5;
d_2_6=(krit6-mu_2_6+rho_sigma_1_sigma_2_6)./sigma_2_6;

%The first quantile is d_1. This is a slight transformation of d_2.
d_1_1=d_2_1+d_1hilf1.*sigma_2_1; d_1_2=d_2_2+d_1hilf2.*sigma_2_2;
d_1_3=d_2_3+d_1hilf3.*sigma_2_3; d_1_4=d_2_4+d_1hilf4.*sigma_2_4;
d_1_5=d_2_5+d_1hilf5.*sigma_2_5; d_1_6=d_2_6+d_1hilf6.*sigma_2_6;

%Price pathes of the option prices (ATTENTION: If time to maturity of
%bonds and options is changed, these formulas need to be readjusted).
optionspreis1=Plaufzeit2.*cdf('Normal',d_1_1,0,1)-K1*Plaufzeit1.*cdf('Normal',d_2_1,0,1);
optionspreis2=Plaufzeit3.*cdf('Normal',d_1_2,0,1)-K2*Plaufzeit1.*cdf('Normal',d_2_2,0,1);
optionspreis3=Plaufzeit4.*cdf('Normal',d_1_3,0,1)-K3*Plaufzeit1.*cdf('Normal',d_2_3,0,1);
optionspreis4=Plaufzeit3.*cdf('Normal',d_1_4,0,1)-K4*Plaufzeit2.*cdf('Normal',d_2_4,0,1);
optionspreis5=Plaufzeit4.*cdf('Normal',d_1_5,0,1)-K5*Plaufzeit2.*cdf('Normal',d_2_5,0,1);
optionspreis6=Plaufzeit4.*cdf('Normal',d_1_6,0,1)-K6*Plaufzeit3.*cdf('Normal',d_2_6,0,1);

%Calculating the components for the pricing of the newly issued option. We
%put the deterministic framework and stochastic elements together.
%We start with mu_2:
vergleichsmu_2_1=vergleichsmu_2_1hilf+rstern.*vergleichsmu_2_hilfe_bei_m_1
                    +khilf.*vergleichsmu_2_hilfe_bei_k_1;
vergleichsmu_2_2=vergleichsmu_2_2hilf+rstern.*vergleichsmu_2_hilfe_bei_m_2
                    +khilf.*vergleichsmu_2_hilfe_bei_k_2;
vergleichsmu_2_3=vergleichsmu_2_3hilf+rstern.*vergleichsmu_2_hilfe_bei_m_3
                    +khilf.*vergleichsmu_2_hilfe_bei_k_3;
vergleichsmu_2_4=vergleichsmu_2_4hilf+rstern.*vergleichsmu_2_hilfe_bei_m_4
                    +khilf.*vergleichsmu_2_hilfe_bei_k_4;
vergleichsmu_2_5=vergleichsmu_2_5hilf+rstern.*vergleichsmu_2_hilfe_bei_m_5
                    +khilf.*vergleichsmu_2_hilfe_bei_k_5;
```

```
vergleichsmu_2_6=vergleichsmu_2_6hilf+rstern.*vergleichsmu_2_hilfe_bei_m_6
                    +khilf.*vergleichsmu_2_hilfe_bei_k_6;

%Now we calculate the values for the two quantiles in the options price
%formula (for the newly issued option).
%At first d_2:
vergleichsd_2_1=(vergleichskrit1-vergleichsmu_2_1+vergleichsrho_sigma_1_sigma_2_1)
                    ./vergleichssigma_2_1;
vergleichsd_2_2=(vergleichskrit2-vergleichsmu_2_2+vergleichsrho_sigma_1_sigma_2_2)
                    ./vergleichssigma_2_2;
vergleichsd_2_3=(vergleichskrit3-vergleichsmu_2_3+vergleichsrho_sigma_1_sigma_2_3)
                    ./vergleichssigma_2_3;
vergleichsd_2_4=(vergleichskrit4-vergleichsmu_2_4+vergleichsrho_sigma_1_sigma_2_4)
                    ./vergleichssigma_2_4;
vergleichsd_2_5=(vergleichskrit5-vergleichsmu_2_5+vergleichsrho_sigma_1_sigma_2_5)
                    ./vergleichssigma_2_5;
vergleichsd_2_6=(vergleichskrit6-vergleichsmu_2_6+vergleichsrho_sigma_1_sigma_2_6)
                    ./vergleichssigma_2_6;

%Second d_1:
vergleichsd_1_1=vergleichsd_2_1+vergleichsd_1hilf1.*vergleichssigma_2_1;
vergleichsd_1_2=vergleichsd_2_2+vergleichsd_1hilf2.*vergleichssigma_2_2;
vergleichsd_1_3=vergleichsd_2_3+vergleichsd_1hilf3.*vergleichssigma_2_3;
vergleichsd_1_4=vergleichsd_2_4+vergleichsd_1hilf4.*vergleichssigma_2_4;
vergleichsd_1_5=vergleichsd_2_5+vergleichsd_1hilf5.*vergleichssigma_2_5;
vergleichsd_1_6=vergleichsd_2_6+vergleichsd_1hilf6.*vergleichssigma_2_6;

%Price pathes of the bonds that are needed to calculate the option price
%for the newly issued options (ATTENTION: If time to maturity of
%bonds and options is changed, these formulas need to be readjusted).
vergleichsPlaufzeit1=vergleichsProhlaufzeit1.*exp(-1/b*rstern*(1-exp(-b*2/12))
                        +khilf.*vergleichskrohmaske1);
vergleichsPlaufzeit2=vergleichsProhlaufzeit2.*exp(-1/b*rstern*(1-exp(-b*14/12))
                        +khilf.*vergleichskrohmaske2);

%Options price pathes for the newly issued options (ATTENTION: If time to maturity
%of bonds and options is changed, these formulas need to be readjusted).
option1=vergleichsPlaufzeit2.*cdf('Normal',vergleichsd_1_1,0,1)
        -vergleichsK1*vergleichsPlaufzeit1.*cdf('Normal',vergleichsd_2_1,0,1);
option2=vergleichsPlaufzeit2.*cdf('Normal',vergleichsd_1_2,0,1)
        -vergleichsK2*vergleichsPlaufzeit1.*cdf('Normal',vergleichsd_2_2,0,1);
option3=vergleichsPlaufzeit2.*cdf('Normal',vergleichsd_1_3,0,1)
        -vergleichsK3*vergleichsPlaufzeit1.*cdf('Normal',vergleichsd_2_3,0,1);
option4=vergleichsPlaufzeit2.*cdf('Normal',vergleichsd_1_4,0,1)
        -vergleichsK4*vergleichsPlaufzeit1.*cdf('Normal',vergleichsd_2_4,0,1);
option5=vergleichsPlaufzeit2.*cdf('Normal',vergleichsd_1_5,0,1)
```

```
        -vergleichsK5*vergleichsPlaufzeit1.*cdf('Normal',vergleichsd_2_5,0,1);
option6=vergleichsPlaufzeit2.*cdf('Normal',vergleichsd_1_6,0,1)
        -vergleichsK6*vergleichsPlaufzeit1.*cdf('Normal',vergleichsd_2_6,0,1);

%Starting values for the calibration of the Vasicek dynamic.
ystart=[b c sigma];
%Calibration of the Vasicek model on the eleven market prices of bonds and
%options. We use a least squares approach to calibrate the model parameters.
%We set bounds for the 3 parameter: 0<=b<=20, 0<=c<=0.5, 0<=sigma<=0.5. We
%do this separately for every point in time.
for j=1:(T/delta)
y=lsqnonlin(@(y)[vasipreisoption(y,optionslaufzeit1,restlaufzeit1,Plaufzeit2(j),
                Plaufzeit1(j),K1)-optionspreis1(j);
vasipreisoption(y,optionslaufzeit2,restlaufzeit2,Plaufzeit3(j),Plaufzeit1(j),K2)
                -optionspreis2(j);
vasipreisoption(y,optionslaufzeit3,restlaufzeit3,Plaufzeit4(j),Plaufzeit1(j),K3)
                -optionspreis3(j);
vasipreisoption(y,optionslaufzeit4,restlaufzeit4,Plaufzeit3(j),Plaufzeit2(j),K4)
                -optionspreis4(j);
vasipreisoption(y,optionslaufzeit5,restlaufzeit5,Plaufzeit4(j),Plaufzeit2(j),K5)
                -optionspreis5(j);
vasipreisoption(y,optionslaufzeit6,restlaufzeit6,Plaufzeit4(j),Plaufzeit3(j),K6)
                -optionspreis6(j);
vasipreisbond(y,laufzeit1,r(j))-Plaufzeit1(j);
vasipreisbond(y,laufzeit2,r(j))-Plaufzeit2(j);
vasipreisbond(y,laufzeit3,r(j))-Plaufzeit3(j);
vasipreisbond(y,laufzeit4,r(j))-Plaufzeit4(j);
vasipreisbond(y,laufzeit5,r(j))-Plaufzeit5(j);],ystart,[0 0 0],[20 0.5 0.5],
                optimset('Display','off'));

%Save the calibrated parameter for every point in time.
y_ergebnis(j,1:3)=y;

%Calculate the resulting price path of the bond prices needed for the option
%price formula.
vergleichsvasibond1(j)=vasipreisbond(y,vergleichsoptionslaufzeit1,r(j));
vergleichsvasibond2(j)=vasipreisbond(y,vergleichsoptionslaufzeit1+vergleichsrestlaufzeit1,
                        r(j));
end


%Calculate the price of the newly issued options if the incorrect dynamic is used.
%(Vasicek dynamic).
vasi1=vasipreisoption2(y_ergebnis,vergleichsoptionslaufzeit1,vergleichsrestlaufzeit1,
                        vergleichsvasibond2,vergleichsvasibond1,vergleichsK1);
vasi2=vasipreisoption2(y_ergebnis,vergleichsoptionslaufzeit2,vergleichsrestlaufzeit2,
```

```
                        vergleichsvasibond2,vergleichsvasibond1,vergleichsK2);
vasi3=vasipreisoption2(y_ergebnis,vergleichsoptionslaufzeit3,vergleichsrestlaufzeit3,
                        vergleichsvasibond2,vergleichsvasibond1,vergleichsK3);
vasi4=vasipreisoption2(y_ergebnis,vergleichsoptionslaufzeit4,vergleichsrestlaufzeit4,
                        vergleichsvasibond2,vergleichsvasibond1,vergleichsK4);
vasi5=vasipreisoption2(y_ergebnis,vergleichsoptionslaufzeit5,vergleichsrestlaufzeit5,
                        vergleichsvasibond2,vergleichsvasibond1,vergleichsK5);
vasi6=vasipreisoption2(y_ergebnis,vergleichsoptionslaufzeit6,vergleichsrestlaufzeit6,
                        vergleichsvasibond2,vergleichsvasibond1,vergleichsK6);

%Write the price pathes in a matrix. Each row stands for the number
%of the Monte-Carlo iteration.
A(durchlaeufe,1:3400)=[vasi1  option1 vasi2 option2 vasi3 option3
vasi4 option4 vasi5 option5 vasi6 option6 r ID
transpose(y_ergebnis(:,1)) transpose(y_ergebnis(:,2))
transpose(y_ergebnis(:,3))]; end
```

To run the code above there are some m-files needed. The program code of this functions is given below in alphabetical order:

bondpreisrohmaske.m:

```
%This function helps to calculate the bond prices. It contains nearly
%all deterministic components for all cases depending on the current point
%in time and the time to maturity.
function P = bondpreisrohmaske(alpha,sigma,sigmaschlange,b,c,s,laufzeit,ta,ts,T,delta)
for i=1:T/delta
    if (i*delta<ta) || (i*delta>=ts)
        P(i)=exp(-c*laufzeit-c/b*(exp(-b*laufzeit)-1)+sigma^2/(4*b^3)
                *(2*b*laufzeit-3+4*exp(-b*laufzeit)-exp(-2*b*laufzeit)));
    elseif (i*delta+laufzeit<=ts)
        P(i)=exp(-c*laufzeit-c/b*(exp(-b*laufzeit)-1)
                -s*(exp((laufzeit-(ts-i*delta))/alpha)-exp(-(ts-i*delta)/alpha))
                +sigma^2/(4*b^3)*(2*b*laufzeit-3+4*exp(-b*laufzeit)-exp(-2*b*laufzeit))
                +sigmaschlange^2*alpha/4*exp((2*laufzeit-2*(ts-i*delta))/alpha)
                *(2*laufzeit/alpha-3+4*exp(-laufzeit/alpha)-exp(-2*laufzeit/alpha)));
    else
        P(i)=exp(-c*laufzeit-c/b*(exp(-b*laufzeit)-1)-s*(1-exp(-(ts-i*delta)/alpha))
                +sigma^2/(4*b^3)*(2*b*laufzeit-3+4*exp(-b*laufzeit)-exp(-2*b*laufzeit))
                +sigmaschlange^2*alpha/4*(2*(ts-i*delta)/alpha-3
                +4*exp(-(ts-i*delta)/alpha)-exp(-2*(ts-i*delta)/alpha)));
    end
end
```

d1maske.m

```
%This function helps to calculate d_1. All values depend on
%the current point in time, the time to maturity of the bond and the time
%to maturity of the option.
function k = d1maske(restlaufzeit,optionslaufzeit,ta,ts,T,delta,b) for i=1:T/delta
    if (i*delta<ta) || (i*delta+optionslaufzeit>=ts)
        k(i)=1/b*(1-exp(-b*restlaufzeit));
    elseif (i*delta+optionslaufzeit+restlaufzeit<=ts)
        k(i)=1/b;
    else
        k(i)=1/b;
    end
end
```

fallmaske.m:

```
%This function helps to calculate the part of the bond price formula
%that depends on the fundamental process k. All values depend on the
%current point in time and the time to maturity.
function k = fallmaske(laufzeit,ta,ts,T,delta,alpha) for i=1:T/delta
    if (i*delta<ta) || (i*delta>=ts)
        k(i)=0;
    elseif (i*delta+laufzeit<=ts)
        k(i)=exp((laufzeit-(ts-i*delta))/alpha)-exp(-(ts-i*delta)/alpha);
    else
        k(i)=1-exp(-(ts-i*delta)/alpha);
    end
end
```

fallmaske2.m:

```
%This function helps to calculate mu_2. All values depend on
%the current point in time, the time to maturity of the bond and the time
%to maturity of the option. This part will be combined with the
%stochastic component of the fundamental process (K-process).
function k = fallmaske2(restlaufzeit,optionslaufzeit,ta,ts,T,delta,alpha,b) for
i=1:T/delta
    if (i*delta<ta) || (i*delta+optionslaufzeit>=ts)
        k(i)=0;
    elseif (i*delta+optionslaufzeit+restlaufzeit<=ts)
        k(i)=b*(exp((optionslaufzeit-(ts-i*delta))/alpha)
            -exp((optionslaufzeit+restlaufzeit-(ts-i*delta))/alpha));
```

```
    else
        k(i)=b*(exp((optionslaufzeit-(ts-i*delta))/alpha)-1);
    end
end
```

fallmaske3.m:

```
%This function helps to calculate mu_2. All values depend on
%the current point in time, the time to maturity of the bond and the time
%to maturity of the option. This part will be combined with the
%stochastic component of the Ornstein-Uhlenbeck process (interest rate
%process in the big economy).
function k = fallmaske3(restlaufzeit,optionslaufzeit,ta,ts,T,delta,b) for i=1:T/delta
    if (i*delta<ta) || (i*delta+optionslaufzeit>=ts)
        k(i)=exp(-b*optionslaufzeit);
    elseif (i*delta+optionslaufzeit+restlaufzeit<=ts)
        k(i)=exp(-b*optionslaufzeit)*(1-exp(-b*restlaufzeit));
    else
        k(i)=exp(-b*optionslaufzeit)*(1-exp(-b*restlaufzeit));
    end
end
```

kovarianzmaske.m:

```
%This function calculates the value of the covariance in the option price
%formula. All values depend on the current point in time, the time to
%maturity of the bond and the time to maturity of the option.
function k =
kovarianzmaske(optionslaufzeit,restlaufzeit,ta,ts,T,delta,b,sigma,sigmaschlange,alpha)
for i=1:T/delta
    if (i*delta<ta) || (i*delta+optionslaufzeit>=ts)
        k(i)=sigma^2/(2*b^2)*(1-exp(-b*optionslaufzeit))^2;
    elseif (i*delta+optionslaufzeit+restlaufzeit<=ts)
        k(i)=sigma^2/(2*b^2)*(1-exp(-b*restlaufzeit))*(1-exp(-b*optionslaufzeit))^2
        +b*alpha*sigmaschlange^2*exp((optionslaufzeit-2*(ts-i*delta))/alpha)
        *(exp(restlaufzeit/alpha)-1-optionslaufzeit/alpha*exp(optionslaufzeit/alpha)
        +exp(optionslaufzeit/alpha)+optionslaufzeit/alpha
        *exp((optionslaufzeit+restlaufzeit)/alpha)
        -exp((optionslaufzeit+restlaufzeit)/alpha));
    else
        k(i)=sigma^2/(2*b^2)*(1-exp(-b*restlaufzeit))*(1-exp(-b*optionslaufzeit))^2
        -b*alpha*sigmaschlange^2*exp((optionslaufzeit-(ts-i*delta))/alpha)
        *(1-optionslaufzeit/alpha-exp(-optionslaufzeit/alpha)+exp(-(ts-i*delta)/alpha)
        +optionslaufzeit/alpha*exp((optionslaufzeit-(ts-i*delta))/alpha)
```

```
        -exp((optionslaufzeit-(ts-i*delta))/alpha));
    end
end
```

kritmaske.m:

```
%This function helps to calculate the critical value of y_0. This value
%is needed to avoid the "max"-term in the integration. All values depend on
%the current point in time, the time to maturity of the bond and the time
%to maturity of the option.
function k =
kritmaske(optionslaufzeit,restlaufzeit,ta,ts,T,delta,alpha,b,K,sigma,c,s,sigmaschlange)
for i=1:T/delta
    if (i*delta<ta) || (i*delta+optionslaufzeit>=ts)
        k(i)=(-b*log(K)-b*c*restlaufzeit+c*(1-exp(-b*restlaufzeit))
            +sigma^2/(4*b^2)*(2*b*restlaufzeit-3+4*exp(-b*restlaufzeit)
            -exp(-2*b*restlaufzeit)))/(1-exp(-b*restlaufzeit));
    elseif (i*delta+optionslaufzeit+restlaufzeit<=ts)
        k(i)=-b*log(K)-b*c*restlaufzeit+c*(1-exp(-b*restlaufzeit))
        +sigma^2/(4*b^2)*(2*b*restlaufzeit-3+4*exp(-b*restlaufzeit)
        -exp(-2*b*restlaufzeit))+sigmaschlange^2*b*alpha/4
        *exp(2*(optionslaufzeit+restlaufzeit-(ts-i*delta))/alpha)
        *(2*restlaufzeit/alpha-3+4*exp(-restlaufzeit/alpha)
        -exp(-2*restlaufzeit/alpha))
        -b*s*(exp((optionslaufzeit+restlaufzeit-(ts-i*delta))/alpha)
        -exp((optionslaufzeit-(ts-i*delta))/alpha));
    else
        k(i)=-b*log(K)-b*c*restlaufzeit+c*(1-exp(-b*restlaufzeit))
        +sigma^2/(4*b^2)*(2*b*restlaufzeit-3+4*exp(-b*restlaufzeit)
        -exp(-2*b*restlaufzeit))+sigmaschlange^2*b*alpha/4
        *(2*((ts-i*delta)-optionslaufzeit)/alpha-3
        -exp(-2*(ts-i*delta-optionslaufzeit)/alpha)
        +4*exp(-(ts-i*delta-optionslaufzeit)/alpha))
        -b*s*(1-exp((optionslaufzeit-(ts-i*delta))/alpha));
    end
end
```

mumaske.m

```
%This function helps to calculate the deterministic components of mu_2.
%All values depend on the current point in time, the time to maturity of the
%bond and the time to maturity of the option.
function k = mumaske(optionslaufzeit,restlaufzeit,ta,ts,T,delta,b,c) for i=1:T/delta
    if (i*delta<ta) || (i*delta+optionslaufzeit>=ts)
```

```
        k(i)=c*(1-exp(-b*optionslaufzeit));
    elseif (i*delta+optionslaufzeit+restlaufzeit<=ts)
        k(i)=c*(1-exp(-b*optionslaufzeit))*(1-exp(-b*restlaufzeit));
    else
        k(i)=c*(1-exp(-b*optionslaufzeit))*(1-exp(-b*restlaufzeit));
    end
end
```

varianz2maske.m

```
%This function calculates the value of sigma_2. All values depend on
%the current point in time, the time to maturity of the bond and the time
%to maturity of the option.
function k =
varianz2maske(optionslaufzeit,restlaufzeit,ta,ts,T,delta,b,alpha,sigma,sigmaschlange) for
i=1:T/delta
    if (i*delta<ta) || (i*delta+optionslaufzeit>=ts)
        k(i)=sigma^2/(2*b)*(1-exp(-2*b*optionslaufzeit));
    elseif (i*delta+optionslaufzeit+restlaufzeit<=ts)
        k(i)=(1-exp(-b*restlaufzeit))^2*sigma^2/(2*b)*(1-exp(-2*b*optionslaufzeit))
            +sigmaschlange^2*optionslaufzeit*b^2*(exp((optionslaufzeit-(ts-i*delta))/alpha)
            -exp((optionslaufzeit+restlaufzeit-(ts-i*delta))/alpha))^2;
    else
        k(i)=(1-exp(-b*restlaufzeit))^2*sigma^2/(2*b)*(1-exp(-2*b*optionslaufzeit))
        +sigmaschlange^2*optionslaufzeit*b^2*(exp((optionslaufzeit-(ts-i*delta))/alpha)-1)^2;
    end
end
```

vasipreisbond.m

```
%Calculates the price of a bond (with Vasicek dynamik).
function P = vasipreisbond(y,bondlaufzeit,zins) if (y(1)>0) && (y(2)>0) && (y(3)>0)
    P=exp(-y(2)*bondlaufzeit+(zins-y(2))/y(1)*(exp(-y(1)*bondlaufzeit)-1)
        +y(3)^2/(4*y(1)^3)*(2*y(1)*bondlaufzeit-3
        +4*exp(-y(1)*bondlaufzeit)-exp(-2*y(1)*bondlaufzeit)));
else
    P=100;
end if y(3)>=1
    P=100;
end if y(2)>=1
    P=100;
end
```

vasipreisoption.m

```
%Calculates the price of a bond option with a Vasicek dynamik.
function P = vasipreisoption(y,optionslaufzeit,restlaufzeit,underlying,abzinsung,K) if
(y(1)>0) && (y(2)>0) && (y(3)>0)
    tau=y(3)./y(1).*(1-exp(-y(1)*restlaufzeit))
        .*sqrt((1-exp(-2*y(1)*optionslaufzeit))./(2*y(1)));
    h=1./tau.*log(underlying./(abzinsung*K))+tau/2;
    P=underlying.*cdf('Normal',h,0,1)-K*abzinsung.*cdf('Normal',h-tau,0,1);
else
    P=100;
end if y(3)>1
    P=100;
end if y(2)>1
    P=100;
end
```

vasipreisoption2.m

```
%Calculates the whole price path of a bond option with a Vasicek dynamik.
function P = vasipreisoption2(y,optionslaufzeit,restlaufzeit,underlying,abzinsung,K)
    tau=y(:,3)./y(:,1).*(1-exp(-y(:,1)*restlaufzeit))
        .*sqrt((1-exp(-2*y(:,1)*optionslaufzeit))./(2*y(:,1)));
    tau=transpose(tau);
    h=1./tau.*log(underlying./(abzinsung.*K))+tau/2;
    P=underlying.*cdf('Normal',h,0,1)-K*abzinsung.*cdf('Normal',h-tau,0,1);
end
```

Appendix B

Programming Codes for Chapter 3

In the following we give the MATLAB-Code for the Maximum-Likelihood estimation of the model introduced in chapter 3. The file main.m is a script file. At first, starting values for the parameter estimation have to be found if one does not chose them manually. After this, the Maximum-Likelihood estimation has to be done. One has to modify the file maximierung_likelihood.m in order to estimate a specific kind of regime-switching conditional heteroskedasticity model. An instruction is given in the comments of the code. If one has the final estimator, the file covariancematrix.m will calculate the corresponding standard deviations. Therefore, the package DERIVESTsuite is needed (as mentioned in the comments of main.m). Again, the file has to be modified in the same way as maximierung_likelihood.m. The functions that compute the value of the likelihood function for a specific kind of regime-switching conditional heteroskedasticity model must be modified manually. An instruction is given in likelihoodfrei.m. To run the code below this step has to be done for all kinds of models that are mentioned in maximierung_likelihood.m. The code is not given here because it is very similar to likelihoodfrei.m (and also very intuitive to create when you have likelihoodfrei.m). Read the comments in likelihoodfrei.m before you start your routine and create all files that are needed.

At first we give the programming code of main.m. After that the other files needed are given in alphabetical order.

main.m:

```
%To start the maximum likelihood estimation one has to load the package
%DERIVESTsuite. If the package do not work properly, change the fixedStep value
%in line 82 of gradest.m to a very small number (we choose 0.0000001). Do the
%same with line 61 of hessdiag.m (we choose 0.0001). After this, create the
%likelihoodfunctions for those models you want to estimate. An instruction
%is given in likelihoodfrei.m. Then modify the file startvalues.m and
%maximierung_likelihood.m to work with these likelihoodfunctions (an
%instruction is given in those files). After this, one can start the
%program.

%Get starting values for different model setups.
startvalues;

%Calculate the ML-estimator for different starting values.
maximierung_likelihood;

%Select the best maximization and get the related standard errors. For this step,
%the above mentioned package is needed. The final matrices consist of the
%parameter vector in the first row and the standard deviation in the second
%row. The last column is the exit flag of the maximization. This value must
%be non-zero.
covariancematrix;
```

covariancematrix.m:

```
%Calculation of the standard deviation of the (quasi maximum likelihood)
%parameter estimations. The procedure is the same for all possible combinations of
%conditional heteroskedasticity models. If only a specific standard deviation of
%one model is needed, copy the corresponding part or make the other parts a comment.

%%%%%%%%%%%%%%%%%%%%%EGARCH-EGARCH
%Get the maximum value of all possible ML-estimations for the EGARCH-EGARCH
%model.
[C,I]=max(ergebnismatrixegarchegarch(:,23));

%Get the vector of those variables that are not fix.
x0=[ergebnismatrixegarchegarch(I,1:14) ergebnismatrixegarchegarch(I,21:22)];

%Estimate the first derivation for all parameters.
[jac,err] = jacobianest(@(x) likelihoodegarchegarchqml(rt,T,x),x0);

%Estimate the covariance matrix via the outer product.
covop=zeros(16,16);
```

```
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);

%Estimate the hessian matrix.
[hess,err] = hessian(@(x) likelihoodegarchegarch(rt,T,x),x0);

%Estimate the covariance matrix via the second derivates.
cov2d=-hess/T;

%Estimate the variance-covarianve matrix as White(1982) does.
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;

%Build a matrix with parameters of the model in the first row. The last two
%entries are the log-likelihood value and a control variable that is not
%zero if the maximization has converged. In the second row are the standard
%deviations of the flexible parameters. Finally save this matrix.
endergebnisegarchegarch(1,1:24)=[ergebnismatrixegarchegarch(I,1:24)];
endergebnisegarchegarch(2,1)=sqrt(qmlcov(1,1));
endergebnisegarchegarch(2,2)=sqrt(qmlcov(2,2));
endergebnisegarchegarch(2,3)=sqrt(qmlcov(3,3));
endergebnisegarchegarch(2,4)=sqrt(qmlcov(4,4));
endergebnisegarchegarch(2,5)=sqrt(qmlcov(5,5));
endergebnisegarchegarch(2,6)=sqrt(qmlcov(6,6));
endergebnisegarchegarch(2,7)=sqrt(qmlcov(7,7));
endergebnisegarchegarch(2,8)=sqrt(qmlcov(8,8));
endergebnisegarchegarch(2,9)=sqrt(qmlcov(9,9));
endergebnisegarchegarch(2,10)=sqrt(qmlcov(10,10));
endergebnisegarchegarch(2,11)=sqrt(qmlcov(11,11));
endergebnisegarchegarch(2,12)=sqrt(qmlcov(12,12));
endergebnisegarchegarch(2,13)=sqrt(qmlcov(13,13));
endergebnisegarchegarch(2,14)=sqrt(qmlcov(14,14));
endergebnisegarchegarch(2,15)=NaN;
endergebnisegarchegarch(2,16)=NaN;
endergebnisegarchegarch(2,17)=NaN;
endergebnisegarchegarch(2,18)=NaN;
endergebnisegarchegarch(2,19)=NaN;
endergebnisegarchegarch(2,20)=NaN;
endergebnisegarchegarch(2,21)=sqrt(qmlcov(15,15));
endergebnisegarchegarch(2,22)=sqrt(qmlcov(16,16));
endergebnisegarchegarch(2,23)=NaN;
endergebnisegarchegarch(2,24)=NaN;
save('endergebnisegarchegarch','endergebnisegarchegarch');
```

```
%%%%%%%%%%%%%%%%%%%%%EGARCH-GARCH
[C,I]=max(ergebnismatrixegarchgarch(:,23));
x0=[ergebnismatrixegarchgarch(I,1:14)ergebnismatrixegarchgarch(I,21)];
[jac,err] = jacobianest(@(x) likelihoodegarchgarchqml(rt,T,x),x0);
covop=zeros(15,15);
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodegarchgarch(rt,T,x),x0);
cov2d=-hess/T;
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;
endergebnisegarchgarch(1,1:24)=[ergebnismatrixegarchgarch(I,1:24)];
endergebnisegarchgarch(2,1)=sqrt(qmlcov(1,1));
endergebnisegarchgarch(2,2)=sqrt(qmlcov(2,2));
endergebnisegarchgarch(2,3)=sqrt(qmlcov(3,3));
endergebnisegarchgarch(2,4)=sqrt(qmlcov(4,4));
endergebnisegarchgarch(2,5)=sqrt(qmlcov(5,5));
endergebnisegarchgarch(2,6)=sqrt(qmlcov(6,6));
endergebnisegarchgarch(2,7)=sqrt(qmlcov(7,7));
endergebnisegarchgarch(2,8)=sqrt(qmlcov(8,8));
endergebnisegarchgarch(2,9)=sqrt(qmlcov(9,9));
endergebnisegarchgarch(2,10)=sqrt(qmlcov(10,10));
endergebnisegarchgarch(2,11)=sqrt(qmlcov(11,11));
endergebnisegarchgarch(2,12)=sqrt(qmlcov(12,12));
endergebnisegarchgarch(2,13)=sqrt(qmlcov(13,13));
endergebnisegarchgarch(2,14)=sqrt(qmlcov(14,14));
endergebnisegarchgarch(2,15)=NaN;
endergebnisegarchgarch(2,16)=NaN;
endergebnisegarchgarch(2,17)=NaN;
endergebnisegarchgarch(2,18)=NaN;
endergebnisegarchgarch(2,19)=NaN;
endergebnisegarchgarch(2,20)=NaN;
endergebnisegarchgarch(2,21)=sqrt(qmlcov(15,15));
endergebnisegarchgarch(2,22)=NaN;
endergebnisegarchgarch(2,23)=NaN;
endergebnisegarchgarch(2,24)=NaN;
save('endergebnisegarchgarch','endergebnisegarchgarch');

%%%%%%%%%%%%%%%%%%%%%GARCH-GARCH
[C,I]=max(ergebnismatrixgarchgarch(:,23));
x0=[ergebnismatrixgarchgarch(I,1:14)];
[jac,err] = jacobianest(@(x) likelihoodgarchgarchqml(rt,T,x),x0);
covop=zeros(14,14);
```

```
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodgarchgarch(rt,T,x),x0);
cov2d=-hess/T;
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;
endergebnisgarchgarch(1,1:24)=[ergebnismatrixgarchgarch(I,1:24)];
endergebnisgarchgarch(2,1)=sqrt(qmlcov(1,1));
endergebnisgarchgarch(2,2)=sqrt(qmlcov(2,2));
endergebnisgarchgarch(2,3)=sqrt(qmlcov(3,3));
endergebnisgarchgarch(2,4)=sqrt(qmlcov(4,4));
endergebnisgarchgarch(2,5)=sqrt(qmlcov(5,5));
endergebnisgarchgarch(2,6)=sqrt(qmlcov(6,6));
endergebnisgarchgarch(2,7)=sqrt(qmlcov(7,7));
endergebnisgarchgarch(2,8)=sqrt(qmlcov(8,8));
endergebnisgarchgarch(2,9)=sqrt(qmlcov(9,9));
endergebnisgarchgarch(2,10)=sqrt(qmlcov(10,10));
endergebnisgarchgarch(2,11)=sqrt(qmlcov(11,11));
endergebnisgarchgarch(2,12)=sqrt(qmlcov(12,12));
endergebnisgarchgarch(2,13)=sqrt(qmlcov(13,13));
endergebnisgarchgarch(2,14)=sqrt(qmlcov(14,14));
endergebnisgarchgarch(2,15)=NaN;
endergebnisgarchgarch(2,16)=NaN;
endergebnisgarchgarch(2,17)=NaN;
endergebnisgarchgarch(2,18)=NaN;
endergebnisgarchgarch(2,19)=NaN;
endergebnisgarchgarch(2,20)=NaN;
endergebnisgarchgarch(2,21)=NaN;
endergebnisgarchgarch(2,22)=NaN;
endergebnisgarchgarch(2,23)=NaN;
endergebnisgarchgarch(2,24)=NaN;
save('endergebnisgarchgarch','endergebnisgarchgarch');

%%%%%%%%%%%%%%%%%%%%AVGARCH-AVGARCH
[C,I]=max(ergebnismatrixavgarchavgarch(:,23));
x0=[ergebnismatrixavgarchavgarch(I,1:14) ergebnismatrixavgarchavgarch(I,19:22)];
[jac,err] = jacobianest(@(x) likelihoodavgarchavgarchqml(rt,T,x),x0);
covop=zeros(18,18);
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodavgarchavgarch(rt,T,x),x0);
cov2d=-hess/T;
```

```
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;
endergebnisavgarchavgarch(1,1:24)=[ergebnismatrixavgarchavgarch(I,1:24)];
endergebnisavgarchavgarch(2,1)=sqrt(qmlcov(1,1));
endergebnisavgarchavgarch(2,2)=sqrt(qmlcov(2,2));
endergebnisavgarchavgarch(2,3)=sqrt(qmlcov(3,3));
endergebnisavgarchavgarch(2,4)=sqrt(qmlcov(4,4));
endergebnisavgarchavgarch(2,5)=sqrt(qmlcov(5,5));
endergebnisavgarchavgarch(2,6)=sqrt(qmlcov(6,6));
endergebnisavgarchavgarch(2,7)=sqrt(qmlcov(7,7));
endergebnisavgarchavgarch(2,8)=sqrt(qmlcov(8,8));
endergebnisavgarchavgarch(2,9)=sqrt(qmlcov(9,9));
endergebnisavgarchavgarch(2,10)=sqrt(qmlcov(10,10));
endergebnisavgarchavgarch(2,11)=sqrt(qmlcov(11,11));
endergebnisavgarchavgarch(2,12)=sqrt(qmlcov(12,12));
endergebnisavgarchavgarch(2,13)=sqrt(qmlcov(13,13));
endergebnisavgarchavgarch(2,14)=sqrt(qmlcov(14,14));
endergebnisavgarchavgarch(2,15)=NaN;
endergebnisavgarchavgarch(2,16)=NaN;
endergebnisavgarchavgarch(2,17)=NaN;
endergebnisavgarchavgarch(2,18)=NaN;
endergebnisavgarchavgarch(2,19)=sqrt(qmlcov(15,15));
endergebnisavgarchavgarch(2,20)=sqrt(qmlcov(16,16));
endergebnisavgarchavgarch(2,21)=sqrt(qmlcov(17,17));
endergebnisavgarchavgarch(2,22)=sqrt(qmlcov(18,18));
endergebnisavgarchavgarch(2,23)=NaN;
endergebnisavgarchavgarch(2,24)=NaN;
save('endergebnisavgarchavgarch','endergebnisavgarchavgarch');

%%%%%%%%%%%%%%%%%%%%FREE-FREE
[C,I]=max(ergebnismatrixfrei(:,23));
x0=[ergebnismatrixfrei(I,1:22)];
[jac,err] = jacobianest(@(x) likelihoodfreiqml(rt,T,x),x0);
covop=zeros(22,22);
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodfrei(rt,T,x),x0);
cov2d=-hess/T;
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;
endergebnisfrei(1,1:24)=[ergebnismatrixfrei(I,1:24)];
endergebnisfrei(2,1)=sqrt(qmlcov(1,1));
endergebnisfrei(2,2)=sqrt(qmlcov(2,2));
endergebnisfrei(2,3)=sqrt(qmlcov(3,3));
```

```
endergebnisfrei(2,4)=sqrt(qmlcov(4,4));
endergebnisfrei(2,5)=sqrt(qmlcov(5,5));
endergebnisfrei(2,6)=sqrt(qmlcov(6,6));
endergebnisfrei(2,7)=sqrt(qmlcov(7,7));
endergebnisfrei(2,8)=sqrt(qmlcov(8,8));
endergebnisfrei(2,9)=sqrt(qmlcov(9,9));
endergebnisfrei(2,10)=sqrt(qmlcov(10,10));
endergebnisfrei(2,11)=sqrt(qmlcov(11,11));
endergebnisfrei(2,12)=sqrt(qmlcov(12,12));
endergebnisfrei(2,13)=sqrt(qmlcov(13,13));
endergebnisfrei(2,14)=sqrt(qmlcov(14,14));
endergebnisfrei(2,15)=sqrt(qmlcov(15,15));
endergebnisfrei(2,16)=sqrt(qmlcov(16,16));
endergebnisfrei(2,17)=sqrt(qmlcov(17,17));
endergebnisfrei(2,18)=sqrt(qmlcov(18,18));
endergebnisfrei(2,19)=sqrt(qmlcov(19,19));
endergebnisfrei(2,20)=sqrt(qmlcov(20,20));
endergebnisfrei(2,21)=sqrt(qmlcov(21,21));
endergebnisfrei(2,22)=sqrt(qmlcov(22,22));
endergebnisfrei(2,23)=NaN; endergebnisfrei(2,24)=NaN;
save('endergebnisfrei','endergebnisfrei');
```

likelihoodfrei.m:

```
%If you need the function to calculate only the likelihood of the parameter
%vector, the last line has to be an uncommented line. If you need the likelihood
%at every single point of time for the qml-estimator, comment the last line out.

%This function calculates the value of the log-likelihood function for the
%process rt with length T and given parameter vector x. To get the
%log-likelihood of a special member of our model class, substitute some
%parts of the vector x in the whole function by the specific value (e.g.
%mu1=0, nu1=1 and b1=0 if regime 1 should have an EGARCH specification).
%For the log-likelihood maximization of different model setups replace
%the parameters by the accordant value and save the function under the
%specific name, e.g. likelihoodegarchavgarch for a model with
%EGARCH in the first regime and AVGARCH in the second regime. To get the
%QML-estimator for the variance-covariance matrix comment the last line out
%and add a qml to the function name (e.g. likelihoodegarchavgarchqml).

%ATTENTION: If you replace e.g. only c1=0, you shorten the vector x and all
%following variables rise in rank. Thus, you have to replace x(22) by x(21)
%and the new vector x has a length of 21.

function P=likelihoodfrei(rt,T,x)
```

```
%1(2) after the name of a variable indicates that this variable belongs to
%regime 1(2). The variables in vector x are:
%lambda1=x(1)
%lambda2=x(2)
%gamma1=x(3)
%gamma2=x(4)
%omega1=x(5)
%omega2=x(6)
%alpha1=x(7)
%alpha2=x(8)
%beta1=x(9)
%beta2=x(10)
%d1=x(11)
%d2=x(12)
%e1=x(13)
%e2=x(14)
%mu1=x(15)
%mu2=x(16)
%nu1=x(17)
%nu2=x(18)
%b1=x(19)
%b2=x(20)
%c1=x(21)
%c2=x(22)

%"a" is the auxiliary variable to transform the absolute value expression in
%a differentiable function. It should be chosen very small. (If one gets
%numerical problems rise a.)
a=0.000001;

%Variance of the process in t=0.
startvarianz=0.0001;

%Initializing of all vectors needed.
%Ex-ante probability to be in regime 1 or 2 at time t.
p1t=NaN(1,T-1);
p2t=NaN(1,T-1);

%Probability to be in regime 1 or 2 at time t. The probability is
%calculated under the condition that the process is at time t+1 in regime 1
%or 2. pit_j is the probability to be in regime i at time t when the
%process is in regime j at time t+1.
p1t_1=NaN(1,T-1);
p1t_2=NaN(1,T-1);
p2t_1=NaN(1,T-1);
p2t_2=NaN(1,T-1);
```

```
%Value of delta at time t if the current regime (at time t) is 1 or 2.
deltat_1=NaN(1,T-1); deltat_2=NaN(1,T-1);

%Asymmetric transformation of delta (news-impact curve).
asymdeltat_1=NaN(1,T-1); asymdeltat_2=NaN(1,T-1);

%Variance at time t if the current regime (at time t) is 1.
h1t=NaN(1,T-1);

%Variance at time t if the current regime (at time t) is 2.
h2t=NaN(1,T-1);

%Averaged variance for time t if the process is in regime 1 at time t+1.
ht_1=NaN(1,T-1);

%Averaged variance for time t if the process is in regime 2 at time t+1.
ht_2=NaN(1,T-1);

%Starting values for the vectors:
%With the assumption p1t(0)=0.5 we get the starting probabilities for both
%regimes.
p1t(1)=normcdf(x(11)+x(13)*rt(1))*(normpdf(rt(1),x(1)+x(3)
        *sqrt(startvarianz),sqrt(startvarianz))*0.5
        /(normpdf(rt(1),x(1)+x(3)*sqrt(startvarianz),sqrt(startvarianz))
        *0.5+normpdf(rt(1),x(2)+x(4)*sqrt(startvarianz),sqrt(startvarianz))*0.5))
        +(1-normcdf(x(12)+x(14)*rt(1)))*((normpdf(rt(1),x(2)
        +x(4)*sqrt(startvarianz),sqrt(startvarianz))
        *0.5/(normpdf(rt(1),x(1)+x(3)*sqrt(startvarianz),sqrt(startvarianz))*0.5
        +normpdf(rt(1),x(2)+x(4)*sqrt(startvarianz),sqrt(startvarianz))*0.5)));
p2t(1)=1-p1t(1);

%We set the error in the first period to zero.
deltat_1(1)=0;
deltat_2(1)=0;

%The starting value for the asymmetric transformation of delta is given by:
asymdeltat_1(1)=sqrt(a^2+(deltat_1(1)-x(19))^2)-x(21)*(deltat_1(1)-x(19));
asymdeltat_2(1)=sqrt(a^2+(deltat_2(1)-x(20))^2)-x(22)*(deltat_2(1)-x(20));

%Variance in regime 1 with lagged variance as assumed above (ht_1(0)=startvarianz).
if x(15)~=0
    h1t(1)=(x(5)+x(7)*sqrt(startvarianz)^x(15)*asymdeltat_1(1)^x(17)
            +x(9)*sqrt(startvarianz)^x(15))^(2/x(15));
else
    %For mu1=0 the box-cox transformation converges to this:
```

```
    h1t(1)=(exp(x(5)+x(7)*asymdeltat_1(1)^x(17)+x(9)*log(sqrt(startvarianz))))^2;
end

%Variance in regime 2 with lagged variance as assumed above (ht_2(0)=startvarianz).
if x(16)~=0
    h2t(1)=(x(6)+x(8)*sqrt(startvarianz)^x(16)*asymdeltat_2(1)^x(18)
            +x(10)*sqrt(startvarianz)^x(16))^(2/x(16));
else
    %For mu2=0 the box-cox transformation converges to this:
    h2t(1)=(exp(x(6)+x(8)*asymdeltat_2(1)^x(18)+x(10)*log(sqrt(startvarianz))))^2;
end

%Plausibility check for value of the variance.
if ~isreal(h1t(1)) || h1t(1)<=0
    h1t(1)=NaN;
end if ~isreal(h2t(1)) || h2t(1)<=0
    h2t(1)=NaN;
end

%pit_j(1) is the probability to be at time t=1 in regime i when the process
%is in t+1 in regime j.
p1t_1(1)=normcdf(x(11)+x(13)*(x(1)+x(3)*sqrt(h1t(1))))*p1t(1)/(normcdf(x(11)
        +x(13)*(x(1)+x(3)*sqrt(h1t(1))))*p1t(1)+(1-normcdf(x(12)+x(14)*(x(2)
        +x(4)*sqrt(h2t(1)))))*p2t(1));
p1t_2(1)=(1-normcdf(x(11)+x(13)*(x(1)+x(3)*sqrt(h1t(1)))))*p1t(1)
        /((1-normcdf(x(11)+x(13)*(x(1)+x(3)*sqrt(h1t(1)))))*p1t(1)
        +normcdf(x(12)+x(14)*(x(2)+x(4)*sqrt(h2t(1))))*p2t(1));
p2t_1(1)=1-p1t_1(1);
p2t_2(1)=1-p1t_2(1);

%Averaged variance for time t=1 if the process is in regime 1 or 2 at time t+1.
ht_1(1)=p1t_1(1)*h1t(1)+(1-p1t_1(1))*h2t(1)+p1t_1(1)*(1-p1t_1(1))*(x(1)
        +x(3)*sqrt(h1t(1))-(x(2)+x(4)*sqrt(h2t(1))))^2;
ht_2(1)=p1t_2(1)*h1t(1)+(1-p1t_2(1))*h2t(1)+p1t_2(1)*(1-p1t_2(1))*(x(1)
        +x(3)*sqrt(h1t(1))-(x(2)+x(4)*sqrt(h2t(1))))^2;

%The vectors initialized above can now be calculated recursively.
for i=2:T-1
    %Ex-ante regime probabilities
    p1t(i)=normcdf(x(11)+x(13)*rt(i))*(normpdf(rt(i),x(1)+x(3)*sqrt(h1t(i-1)),
            sqrt(h1t(i-1)))*p1t(i-1)/(normpdf(rt(i),x(1)+x(3)*sqrt(h1t(i-1)),
            sqrt(h1t(i-1)))*p1t(i-1)+normpdf(rt(i),x(2)+x(4)*sqrt(h2t(i-1)),
            sqrt(h2t(i-1)))*p2t(i-1)))+(1-normcdf(x(12)+x(14)*rt(i)))
            *((normpdf(rt(i),x(2)+x(4)*sqrt(h2t(i-1)),sqrt(h2t(i-1)))*p2t(i-1)
            /(normpdf(rt(i),x(1)+x(3)*sqrt(h1t(i-1)),sqrt(h1t(i-1)))*p1t(i-1)
            +normpdf(rt(i),x(2)+x(4)*sqrt(h2t(i-1)),sqrt(h2t(i-1)))*p2t(i-1))));
```

```
p2t(i)=1-p1t(i);

%Calculate the error terms and their asymmetric transformation.
deltat_1(i)=p1t_1(i-1)*(rt(i)-(x(1)+x(3)*sqrt(h1t(i-1))))/(sqrt(h1t(i-1)))
            +p2t_1(i-1)*(rt(i)-(x(2)+x(4)*sqrt(h2t(i-1))))/(sqrt(h2t(i-1)));
deltat_2(i)=p1t_2(i-1)*(rt(i)-(x(1)+x(3)*sqrt(h1t(i-1))))/(sqrt(h1t(i-1)))
            +p2t_2(i-1)*(rt(i)-(x(2)+x(4)*sqrt(h2t(i-1))))/(sqrt(h2t(i-1)));
asymdeltat_1(i)=sqrt(a^2+(deltat_1(i)-x(19))^2)-x(21)*(deltat_1(i)-x(19));
asymdeltat_2(i)=sqrt(a^2+(deltat_2(i)-x(20))^2)-x(22)*(deltat_2(i)-x(20));

%Calculate the variance process for regime 1.
if x(15)~=0
    h1t(i)=(x(5)+x(7)*sqrt(ht_1(i-1))^x(15)*asymdeltat_1(i)^x(17)
            +x(9)*sqrt(ht_1(i-1))^x(15))^(2/x(15));
else
    h1t(i)=(exp(x(5)+x(7)*asymdeltat_1(i)^x(17)+x(9)*log(sqrt(ht_1(i-1)))))^2;
end

%Calculate the variance process for regime 2.
if x(16)~=0
    h2t(i)=(x(6)+x(8)*sqrt(ht_2(i-1))^x(16)*asymdeltat_2(i)^x(18)
            +x(10)*sqrt(ht_2(i-1))^x(16))^(2/x(16));
else
    h2t(i)=(exp(x(6)+x(8)*asymdeltat_2(i)^x(18)+x(10)*log(sqrt(ht_2(i-1)))))^2;
end

%Plausibility check for value of the variance.
if ~isreal(h1t(i)) || h1t(i)<=0
    h1t(i)=NaN;
end
if ~isreal(h2t(i)) || h2t(i)<=0
    h2t(i)=NaN;
end

%Calculate probabilities under the condition that the following
%regime is known.
p1t_1(i)=normcdf(x(11)+x(13)*(x(1)+x(3)*sqrt(h1t(i))))*p1t(i)
        /(normcdf(x(11)+x(13)*(x(1)+x(3)*sqrt(h1t(i))))*p1t(i)
        +(1-normcdf(x(12)+x(14)*(x(2)+x(4)*sqrt(h2t(i)))))*p2t(i));
p1t_2(i)=(1-normcdf(x(11)+x(13)*(x(1)+x(3)*sqrt(h1t(i)))))*p1t(i)
        /((1-normcdf(x(11)+x(13)*(x(1)+x(3)*sqrt(h1t(i)))))*p1t(i)
        +normcdf(x(12)+x(14)*(x(2)+x(4)*sqrt(h2t(i))))*p2t(i));
p2t_1(i)=1-p1t_1(i);
p2t_2(i)=1-p1t_2(i);

%Calculating variances under the condition that the following
```

```
    %regime is known.
    ht_1(i)=p1t_1(i)*h1t(i)+(1-p1t_1(i))*h2t(i)+p1t_1(i)*(1-p1t_1(i))
            *(x(1)+x(3)*sqrt(h1t(i))-(x(2)+x(4)*sqrt(h2t(i))))^2;
    ht_2(i)=p1t_2(i)*h1t(i)+(1-p1t_2(i))*h2t(i)+p1t_2(i)*(1-p1t_2(i))
            *(x(1)+x(3)*sqrt(h1t(i))-(x(2)+x(4)*sqrt(h2t(i))))^2;
end

%Calculate the value of the likelihood function.
P(1)=0; for i=2:T
    P(i-1)=log(normpdf(rt(i),x(1)+x(3)*sqrt(h1t(i-1)),sqrt(h1t(i-1)))*p1t(i-1)
            +normpdf(rt(i),x(2)+x(4)*sqrt(h2t(i-1)),sqrt(h2t(i-1)))*p2t(i-1));
    %Set likelihood equal to -infinity if the likelihood is not a real number
    %or the likelihood is not a number.
    if isnan(P(i-1)) || ~isreal(P)
        P(i-1)=-inf;
    end
end
%If you need the function to calculate only the likelihood of the parameter
%vector the next line has to be no comment. If you need the likelihood in
%every single point of time for the qml-estimator, comment the next
%line out.
P=sum(P);
end
```

maximierung_likelihood.m:

```
%ML estimation. At first define algorithm (BFGS) and some optimization
%conditions.
options=optimset('Algorithm','active-set','MaxIter',400,
                'MaxFunEvals',2200,'TolFun',1e-3);

%%%%%%%%%%%%%%%%%%%%%%%%Free-Free model
%Load the starting values for the free-free model (comment this line if the
%matrixes is still in work space).
load('ergebnismatrixfrei')

%Get the number of vectors with starting values.
n=size(ergebnismatrixfrei,1);

%Define constraints for the parameter vector. If there are no constraints set
%minus and plus infinity. This speeds up the estimation but be careful to
%set the constraints not to restrictive.
untergrenze=[-0.03 -0.03 -2 -2 0 0 0 0 0 0 1.8 1.8 -inf -inf
            0 0 0.25 0.25 -inf -inf -1 -1];
obergrenze=[0.03 0.03 2 2 0.4 0.4 1 1 1 1 6 6 inf inf
```

```
            4 4 4 4 inf inf 1 1];

%Start the maximization.
for i=1:n
    [x,fval,exitflag] = fmincon(@(x) -likelihoodfrei(rt,T,x),
    [ergebnismatrixfrei(i,1:22)],[],[],[],[],
    untergrenze,obergrenze,[],options);
    %Overwrite the value the optimization started with the results from
    %the maximization. After that write the likelihood value and at last
    %the exitflag. The exitflag shows if the parametervector is an
    %optimum or if e.g. the optimization stopped because the maximum
    %number of iteration steps has been reached. If the value is e.g.
    %zero just start the optimization again but use the final values as
    %new starting values.
    ergebnismatrixfrei(i,1:24)=[x(1:22) -fval exitflag];
end save('ergebnismatrixfrei','ergebnismatrixfrei')

%%%%%%%%%%%%%%%%%%%%%%%EGARCH-EGARCH model
load('ergebnismatrixegarchegarch')
n=size(ergebnismatrixegarchegarch,1);
untergrenze=[-0.03 -0.03 -2 -2 -inf -inf -inf -inf -inf -inf
            1.8 1.8 -inf -inf -inf -inf];
obergrenze=[0.03 0.03 2 2 inf inf inf inf inf inf 6 6 inf inf inf inf];
for i=1:n
    [x,fval,exitflag] = fmincon(@(x) -likelihoodegarchegarch(rt,T,x),
            [ergebnismatrixegarchegarch(i,1:14) ergebnismatrixegarchegarch(i,21:22)],
            [],[],[],[],untergrenze,obergrenze,[],options);
    ergebnismatrixegarchegarch(i,1:24)=[x(1:14) 0 0 1 1 0 0 x(15) x(16) -fval exitflag];
end
save('ergebnismatrixegarchegarch','ergebnismatrixegarchegarch')

%%%%%%%%%%%%%%%%%%%%%%%GARCH-GARCH model
load('ergebnismatrixgarchgarch')
n=size(ergebnismatrixgarchgarch,1);
untergrenze=[-0.03-0.03 -2 -2 0 0 0 0 0 0 1.8 1.8 -inf -inf];
obergrenze=[0.03 0.03 2 2 0.1 0.1 1 1 1 1 6 6 inf inf];
for i=1:n
    [x,fval,exitflag] = fmincon(@(x) -likelihoodgarchgarch(rt,T,x),
                    [ergebnismatrixgarchgarch(i,1:14)],[],[],[],[],
                    untergrenze,obergrenze,[],options);
    ergebnismatrixgarchgarch(i,1:24)=[x(1:14) 2 2 2 2 0 0 0 0 -fval exitflag];
end
save('ergebnismatrixgarchgarch','ergebnismatrixgarchgarch')

%%%%%%%%%%%%%%%%%%%%%%%%AVGARCH-AVGARCH model
load('ergebnismatrixavgarchavgarch')
```

```
n=size(ergebnismatrixavgarchavgarch,1);
untergrenze=[-0.03 -0.03 -2 -2 0 0 0 0 0 0 1.8 1.8 -inf -inf -inf -inf -1 -1];
obergrenze=[0.3 0.3 2 2 0.4 0.4 1 1 1 1 6 6 inf inf inf inf 1 1];
for i=1:n
    [x,fval,exitflag] = fmincon(@(x) -likelihoodavgarchavgarch(rt,T,x),
    [ergebnismatrixavgarchavgarch(i,1:14) ergebnismatrixavgarchavgarch(i,19:22)],
    [],[],[],[],untergrenze,obergrenze,[],options);
    ergebnismatrixavgarchavgarch(i,1:24)=[x(1:14) 1 1 1 1 x(15:18) -fval exitflag];
end
save('ergebnismatrixavgarchavgarch','ergebnismatrixavgarchavgarch')

%%%%%%%%%%%%%%%%%%%%%%%EGARCH-GARCH model
load('ergebnismatrixegarchgarch')
n=size(ergebnismatrixegarchgarch,1);
untergrenze=[-0.03 -0.03 -2 -2 -inf 0 -inf 0 -inf 0 1.8 1.8 -inf -inf -inf];
obergrenze=[0.03 0.03 2 2 inf 0.2 inf 1 inf 1 6 6 inf inf inf];
for i=1:n
    [x,fval,exitflag] = fmincon(@(x) -likelihoodegarchgarch(rt,T,x),
    [ergebnismatrixegarchgarch(i,1:14) ergebnismatrixegarchgarch(i,21)],
    [],[],[],[],untergrenze,obergrenze,[],options);
    ergebnismatrixegarchgarch(i,1:24)=[x(1:14) 0 2 1 2 0 0 x(15) 0 -fval exitflag];
end
save('ergebnismatrixegarchgarch','ergebnismatrixegarchgarch')
```

startvalues.m:

```
%This program looks for starting values for the parameters of the likelihood
%maximization from our regime switching model. Therefore, we draw a random
%sample out of suitable intervals and compare the likelihood
%functions of these random starting values. Output is a selection of this
%random start values.

%Number of random draws per model.
N=20000;

%Number of how many of the best results should be saved.
n=40;

%Process for which the model should be adjusted.
rt=renditeindex;

%Length of the process.
T=length(rt);

%In what follows the starting values for different kinds of models are
```

```
%calculated. Comments are only on the first one (all other are similar).

%%%%%%%%%%%%%%%%%%%%%%Free-Free model
%untergrenze is a vector with the lower limits for the parameters. To
%identify the position in the vector of all parameters see the function to
%calculate the specific likelihood value (likelihoodefrei.m in this case).
%obergrenze is the corresponding vector of the upper limits.
untergrenze=[-0.01 -0.01 -2 -2 0 0 0 0 0 0 1.8 1.8 0 0 0 0 0.25 0.25 0 0 -1 -1];
obergrenze=[0.01 0.01 2 2 0.4 0.4 1 1 1 1 6 6 0 0 4 4 4 4 0 0 1 1];

%Create a matrix that has the best n parameter choices. The matrix has a
%column more than there are parameters. In the last column the likelihood
%values are recorded (at the beginning the matrix is filled with NaN values
%and the likelihood values for these parameters is minus infinity).
zielmatrixfrei=NaN(n,23); for i=1:n
    zielmatrixfrei(i,23)=-inf;
end

%A variable is defined that is the likelihoodvalue that must be exceeded to
%belong to the n best parameter setups.
bester=-inf;

%A control variable that counts the number of valid parameter choices.
anzahl=0;

%N drawings of parameter vectors.
for i=1:N
    %Draw from the uniform distribution between the above defined upper and
    %lower limits.
    y=[random('unif',untergrenze,obergrenze)];

    %rating is the likelihood value corresponding to this parameter vector
    %(this row changes in the parts below).
    rating=likelihoodfrei(rt,T,y);

    %If the likelihoodvalue of this parameter vector is better than at
    %least one vector of our best-of list, make a new best-of list of the
    %parameter choices. The value "23" depends on the length of the
    %parameter vector and needs to be changed for other model setups (see below).
    if rating>bester
        vergleich=zielmatrixfrei(n-1,23);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),23);
```

```
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,23);
    end

    %Count the number of valid parameter choices tested (invalid choices
    %get a likelihoodvalue of minus infinity).
    anzahl=anzahl+isfinite(rating);
end

%Save the best-of list of parameter choices. Add a column with zeros to the
%matrix. This is an auxiliary step for our maximization. Furthermore, expand
%the parametervector with those values that are predetermined to the model
%choice (if the model is not totally free but e.g. an EGARCH or GARCH model
%in some regimes). Add these values at that place they are in the free-free
%model.
ergebnismatrixfrei(1:n,1:24)=[zielmatrixfrei(:,1:23) zeros(n,1)];
save('ergebnismatrixfrei','ergebnismatrixfrei')

%%%%%%%%%%%%%%%%%%%%%%%EGARCH-GARCH model
untergrenze=[-0.03 -0.03 -2 -2 -1 0 -1 0 -1 0 1.8 1.8 0 0 0];
obergrenze=[0.03 0.03 2 2 1 0.2 1 1 1 1 6 6 0 0 0];
zielmatrixfrei=NaN(n,16);
for i=1:n
    zielmatrixfrei(i,16)=-inf;
end
bester=-inf;
anzahl=0;
for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodegarchgarch(rt,T,y);
    if rating>bester
        vergleich=zielmatrixfrei(n-1,16);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),16);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,16);
    end
    anzahl=anzahl+isfinite(rating);
end
ergebnismatrixegarchgarch(1:n,1:24)=[zielmatrixfrei(:,1:14) zeros(n,1) ones(n,1)*2
            ones(n,1) ones(n,1)*2 zeros(n,1) zeros(n,1) zielmatrixfrei(:,15) zeros(n,1)
```

```
            zielmatrixfrei(:,16) zeros(n,1)];
save('ergebnismatrixegarchgarch','ergebnismatrixegarchgarch')

%%%%%%%%%%%%%%%%%%%%%GARCH-GARCH model
untergrenze=[-0.03 -0.03 -2 -2 0 0 0 0 0 0 1.8 1.8 0 0];
obergrenze=[0.03 0.03 2 2 0.1 0.1 1 1 1 1 6 6 0 0];
zielmatrixfrei=NaN(n,15);
for i=1:n
    zielmatrixfrei(i,15)=-inf;
end
bester=-inf;
anzahl=0;
for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodgarchgarch(rt,T,y);
    if rating>bester
        vergleich=zielmatrixfrei(n-1,15);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),15);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,15);
    end
    anzahl=anzahl+isfinite(rating);
end
ergebnismatrixgarchgarch(1:n,1:24)=[zielmatrixfrei(:,1:14) ones(n,1)*2 ones(n,1)*2
            ones(n,1)*2 ones(n,1)*2 zeros(n,1) zeros(n,1) zeros(n,1) zeros(n,1)
            zielmatrixfrei(:,15) zeros(n,1)];
save('ergebnismatrixgarchgarch','ergebnismatrixgarchgarch')

%%%%%%%%%%%%%%%%%%%%%AVGARCH-AVARCH model
untergrenze=[-0.03 -0.03 -2 -2 0 0 0 0 0 0 1.8 1.8 0 0 0 0 -1 -1];
obergrenze=[0.3 0.3 2 2 0.4 0.4 1 1 1 1 6 6 0 0 0 0 1 1];
zielmatrixfrei=NaN(n,19);
for i=1:n
    zielmatrixfrei(i,19)=-inf;
end
bester=-inf;
anzahl=0;
for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodavgarchavgarch(rt,T,y);
    if rating>bester
```

```
        vergleich=zielmatrixfrei(n-1,19);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),19);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,19);
    end
    anzahl=anzahl+isfinite(rating);
end
ergebnismatrixavgarchavgarch(1:n,1:24)=[zielmatrixfrei(:,1:14) ones(n,1) ones(n,1)
                    ones(n,1) ones(n,1) zielmatrixfrei(:,15:19) zeros(n,1)];
save('ergebnismatrixavgarchavgarch','ergebnismatrixavgarchavgarch')

%%%%%%%%%%%%%%%%%%%%%EGARCH-EGARCH model
untergrenze=[-0.03 -0.03 -2 -2 -1 -1 -1 -1 -1 -1 1.8 1.8 0 0 0 0];
obergrenze=[0.03 0.03 2 2 1 1 1 1 1 1 6 6 0 0 0 0];
zielmatrixfrei=NaN(n,17);
for i=1:n
    zielmatrixfrei(i,17)=-inf;
end
bester=-inf;
anzahl=0;
for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodegarchegarch(rt,T,y);
    if rating>bester
        vergleich=zielmatrixfrei(n-1,17);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),17);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,17);
    end
    anzahl=anzahl+isfinite(rating);
end ergebnismatrixegarchegarch(1:n,1:24)=[zielmatrixfrei(:,1:14) zeros(n,1) zeros(n,1)
        ones(n,1) ones(n,1) zeros(n,1) zeros(n,1) zielmatrixfrei(:,15:17) zeros(n,1)];
save('ergebnismatrixegarchegarch','ergebnismatrixegarchegarch')
```

Appendix C

Programming Codes for Chapter 4

In the following we give the MATLAB-Code for the Maximum-Likelihood estimation of the model in chapter 4. This is only a special case of the routine seen in Appendix B. Furthermore, this code enables to calculate the ex-post regime probabilities. The program main.m is just a script file. At first starting values for the parameter estimation have to be found if one does not chose them manually. After this the Maximum-Likelihood estimation has to be done. At last the file covariancematrix.m will calculate the corresponding standard deviations. Therefore, the package DERIVESTsuite is needed. The functions that give the value of the likelihood function for a specific kind of regime-switching conditional heteroskedasticity model must be created manually. An instruction is given in likelihoodfrei.m. To run the code below, this step has to be done for all kinds of model that are mentioned in maximierung_likelihood.m. The code is not given here because it is very similar to likelihoodfrei.m (and also very intuitive to create when one has likelihoodfrei.m). Read the comments in likelihoodfrei.m in the chapter above before starting the routine and create all files that are needed.

At first we give the programming code of main.m. After that the other files needed are given in alphabetical order.

main.m:

```
%To start the maximum likelihood estimation, load the package
%DERIVESTsuite. If the package does not work properly, in line 82 of gradest.m
%the fixedStep value to a very small number (we choose 0.0000001).
%Do the same with line 61 of hessdiag.m (we choose 0.0001). After this
%create the likelihoodfunctions for those models you want to estimate. An
%instruction is given in likelihoodfrei.m. Then modify the file
%startvalues.m and maximierung_likelihood.m to work with this
%likelihoodfunctions. An detailed instruction is given in the files of the
%general model. After this being done you can start the program.

%Get starting values for different model setups.
startvalues;

%Calculate the ML-estimator for different starting values.
maximierung_likelihood;

%Select the best maximization and get the corresponding standard errors. For this step
%you need the above mentioned package. The final matrixes have the
%parameter vector in the first row and the standard deviation in the second
%row. The last column is the exit flag of the maximization. This value must
%be non-zero.
covariancematrix;
```

covariancematrix.m:

```
%Calculation of the standard deviation of the (quasi maximum likelihood)
%parameter estimations. The procedure is a special case of the general
%model. Input are two time series, one sample group (indexbanken) and one
%control group (indexcontrol).

%TGARCH-TGARCH model.
rt=indexbanken;
T=length(rt);
[C,I]=max(ergebnismatrixtgarchtgarchbanken(:,18));
x0=[ergebnismatrixtgarchtgarchbanken(I,1:9) ergebnismatrixtgarchtgarchbanken(I,16:17)];
[jac,err] = jacobianest(@(x) likelihoodtgarchtgarchqml(rt,T,x),x0); covop=zeros(11,11);
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodtgarchtgarch(rt,T,x),x0);
cov2d=-hess/T;
cov=cov2d/covop*cov2d;
```

```
qmlcov=inv(cov)/T;
endergebnistgarchtgarchbanken(1,1:19)=[ergebnismatrixtgarchtgarchbanken(I,1:19)];
endergebnistgarchtgarchbanken(2,1)=sqrt(qmlcov(1,1));
endergebnistgarchtgarchbanken(2,2)=sqrt(qmlcov(2,2));
endergebnistgarchtgarchbanken(2,3)=sqrt(qmlcov(3,3));
endergebnistgarchtgarchbanken(2,4)=sqrt(qmlcov(4,4));
endergebnistgarchtgarchbanken(2,5)=sqrt(qmlcov(5,5));
endergebnistgarchtgarchbanken(2,6)=sqrt(qmlcov(6,6));
endergebnistgarchtgarchbanken(2,7)=sqrt(qmlcov(7,7));
endergebnistgarchtgarchbanken(2,8)=sqrt(qmlcov(8,8));
endergebnistgarchtgarchbanken(2,9)=sqrt(qmlcov(9,9));
endergebnistgarchtgarchbanken(2,10)=NaN; endergebnistgarchtgarchbanken(2,11)=NaN;
endergebnistgarchtgarchbanken(2,12)=NaN; endergebnistgarchtgarchbanken(2,13)=NaN;
endergebnistgarchtgarchbanken(2,14)=NaN; endergebnistgarchtgarchbanken(2,15)=NaN;
endergebnistgarchtgarchbanken(2,16)=sqrt(qmlcov(10,10));
endergebnistgarchtgarchbanken(2,17)=sqrt(qmlcov(11,11));
endergebnistgarchtgarchbanken(2,18)=NaN; endergebnistgarchtgarchbanken(2,19)=NaN;
save('endergebnistgarchtgarchbanken','endergebnistgarchtgarchbanken');

%EGARCH-TGARCH model.
[C,I]=max(ergebnismatrixegarchtgarchbanken(:,18));
x0=[ergebnismatrixegarchtgarchbanken(I,1:9) ergebnismatrixegarchtgarchbanken(I,16:17)];
[jac,err] = jacobianest(@(x) likelihoodegarchtgarchqml(rt,T,x),x0); covop=zeros(11,11);
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodegarchtgarch(rt,T,x),x0);
cov2d=-hess/T;
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;
endergebnisegarchtgarchbanken(1,1:19)=[ergebnismatrixegarchtgarchbanken(I,1:19)];
endergebnisegarchtgarchbanken(2,1)=sqrt(qmlcov(1,1));
endergebnisegarchtgarchbanken(2,2)=sqrt(qmlcov(2,2));
endergebnisegarchtgarchbanken(2,3)=sqrt(qmlcov(3,3));
endergebnisegarchtgarchbanken(2,4)=sqrt(qmlcov(4,4));
endergebnisegarchtgarchbanken(2,5)=sqrt(qmlcov(5,5));
endergebnisegarchtgarchbanken(2,6)=sqrt(qmlcov(6,6));
endergebnisegarchtgarchbanken(2,7)=sqrt(qmlcov(7,7));
endergebnisegarchtgarchbanken(2,8)=sqrt(qmlcov(8,8));
endergebnisegarchtgarchbanken(2,9)=sqrt(qmlcov(9,9));
endergebnisegarchtgarchbanken(2,10)=NaN; endergebnisegarchtgarchbanken(2,11)=NaN;
endergebnisegarchtgarchbanken(2,12)=NaN; endergebnisegarchtgarchbanken(2,13)=NaN;
endergebnisegarchtgarchbanken(2,14)=NaN; endergebnisegarchtgarchbanken(2,15)=NaN;
endergebnisegarchtgarchbanken(2,16)=sqrt(qmlcov(10,10));
endergebnisegarchtgarchbanken(2,17)=sqrt(qmlcov(11,11));
```

```
endergebnisegarchtgarchbanken(2,18)=NaN; endergebnisegarchtgarchbanken(2,19)=NaN;
save('endergebnisegarchtgarchbanken','endergebnisegarchtgarchbanken');

%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%Now calculate values for the control group.
%%%%%%%%%%%%%%%

%TGARCH-TGARCH model.
rt=indexcontrol;
T=length(rt);
[C,I]=max(ergebnismatrixtgarchtgarchcontrol(:,18));
x0=[ergebnismatrixtgarchtgarchcontrol(I,1:9) ergebnismatrixtgarchtgarchcontrol(I,16:17)];
[jac,err] = jacobianest(@(x) likelihoodtgarchtgarchqml(rt,T,x),x0); covop=zeros(11,11);
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodtgarchtgarch(rt,T,x),x0);
cov2d=-hess/T;
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;
endergebnistgarchtgarchcontrol(1,1:19)=[ergebnismatrixtgarchtgarchcontrol(I,1:19)];
endergebnistgarchtgarchcontrol(2,1)=sqrt(qmlcov(1,1));
endergebnistgarchtgarchcontrol(2,2)=sqrt(qmlcov(2,2));
endergebnistgarchtgarchcontrol(2,3)=sqrt(qmlcov(3,3));
endergebnistgarchtgarchcontrol(2,4)=sqrt(qmlcov(4,4));
endergebnistgarchtgarchcontrol(2,5)=sqrt(qmlcov(5,5));
endergebnistgarchtgarchcontrol(2,6)=sqrt(qmlcov(6,6));
endergebnistgarchtgarchcontrol(2,7)=sqrt(qmlcov(7,7));
endergebnistgarchtgarchcontrol(2,8)=sqrt(qmlcov(8,8));
endergebnistgarchtgarchcontrol(2,9)=sqrt(qmlcov(9,9));
endergebnistgarchtgarchcontrol(2,10)=NaN; endergebnistgarchtgarchcontrol(2,11)=NaN;
endergebnistgarchtgarchcontrol(2,12)=NaN; endergebnistgarchtgarchcontrol(2,13)=NaN;
endergebnistgarchtgarchcontrol(2,14)=NaN; endergebnistgarchtgarchcontrol(2,15)=NaN;
endergebnistgarchtgarchcontrol(2,16)=sqrt(qmlcov(10,10));
endergebnistgarchtgarchcontrol(2,17)=sqrt(qmlcov(11,11));
endergebnistgarchtgarchcontrol(2,18)=NaN; endergebnistgarchtgarchcontrol(2,19)=NaN;
save('endergebnistgarchtgarchcontrol','endergebnistgarchtgarchcontrol');

%EGARCH-TGARCH model.
[C,I]=max(ergebnismatrixegarchtgarchcontrol(:,18));
x0=[ergebnismatrixegarchtgarchcontrol(I,1:9) ergebnismatrixegarchtgarchcontrol(I,16:17)];
[jac,err] = jacobianest(@(x) likelihoodegarchtgarchqml(rt,T,x),x0); covop=zeros(11,11);
for i=1:T-1
    covop=jac(i,:)'*jac(i,:)+covop;
end
```

```
    covop=covop/(T-1);
[hess,err] = hessian(@(x) likelihoodegarchtgarch(rt,T,x),x0);
cov2d=-hess/T;
cov=cov2d/covop*cov2d;
qmlcov=inv(cov)/T;
endergebnisegarchtgarchcontrol(1,1:19)=[ergebnismatrixegarchtgarchcontrol(I,1:19)];
endergebnisegarchtgarchcontrol(2,1)=sqrt(qmlcov(1,1));
endergebnisegarchtgarchcontrol(2,2)=sqrt(qmlcov(2,2));
endergebnisegarchtgarchcontrol(2,3)=sqrt(qmlcov(3,3));
endergebnisegarchtgarchcontrol(2,4)=sqrt(qmlcov(4,4));
endergebnisegarchtgarchcontrol(2,5)=sqrt(qmlcov(5,5));
endergebnisegarchtgarchcontrol(2,6)=sqrt(qmlcov(6,6));
endergebnisegarchtgarchcontrol(2,7)=sqrt(qmlcov(7,7));
endergebnisegarchtgarchcontrol(2,8)=sqrt(qmlcov(8,8));
endergebnisegarchtgarchcontrol(2,9)=sqrt(qmlcov(9,9));
endergebnisegarchtgarchcontrol(2,10)=NaN; endergebnisegarchtgarchcontrol(2,11)=NaN;
endergebnisegarchtgarchcontrol(2,12)=NaN; endergebnisegarchtgarchcontrol(2,13)=NaN;
endergebnisegarchtgarchcontrol(2,14)=NaN; endergebnisegarchtgarchcontrol(2,15)=NaN;
endergebnisegarchtgarchcontrol(2,16)=sqrt(qmlcov(10,10));
endergebnisegarchtgarchcontrol(2,17)=sqrt(qmlcov(11,11));
endergebnisegarchtgarchcontrol(2,18)=NaN; endergebnisegarchtgarchcontrol(2,19)=NaN;
save('endergebnisegarchtgarchcontrol','endergebnisegarchtgarchcontrol');
```

likelihoodfrei.m:

```
%This is a special case of the general model.
%For the log-likelihood maximization of the different model setups,
%replace the parameters by the accordant values and save the function
%under the specific name, e.g. likelihoodegarchtgarch for a model with
%EGARCH in the first regime and TGARCH in the second regime. To get the
%QML-estimator for the variance-covariance matrix, comment the last line out
%and add a qml on the function name (e.g. likelihoodegarchtgarchqml). After
%this you can use covariancematrix.m. For further details see the comments
%in the general model.

%ATTENTION: Only in this version the filter and ex-post probabilities are
%calculated. There is no output of them, they have to be saved separately.
%See the lower part of this file.
function P=likelihoodfrei(rt,T,x)
%lambda1=x(1)
%lambda2=x(1)
%gamma1=0
%gamma2=0
%omega1=x(2)
%omega2=x(3)
```

```
%alpha1=x(4)
%alpha2=x(5)
%beta1=x(6)
%beta2=x(7)
%d1=x(8)
%d2=x(9)
%e1=0
%e2=0
%mu1=x(10)
%mu2=x(11)
%nu1=x(12)
%nu2=x(13)
%b1=0
%b2=0
%c1=x(14)
%c2=x(15)
a=0.000001;
startvarianz=0.0001;
p1t=NaN(1,T-1);
p2t=NaN(1,T-1);
p1t_1=NaN(1,T-1);
p1t_2=NaN(1,T-1);
p2t_1=NaN(1,T-1);
p2t_2=NaN(1,T-1);
deltat_1=NaN(1,T-1);
deltat_2=NaN(1,T-1);
asymdeltat_1=NaN(1,T-1);
asymdeltat_2=NaN(1,T-1);
h1t=NaN(1,T-1);
h2t=NaN(1,T-1);
ht_1=NaN(1,T-1);
ht_2=NaN(1,T-1);
p1t(1)=normcdf(x(8)+0*rt(1))*(normpdf(rt(1),x(1),sqrt(startvarianz))
        *0.5/(normpdf(rt(1),x(1),sqrt(startvarianz))*0.5
        +normpdf(rt(1),x(1),sqrt(startvarianz))*0.5))+(1-normcdf(x(9)))
        *((normpdf(rt(1),x(1),sqrt(startvarianz))*0.5
        /(normpdf(rt(1),x(1),sqrt(startvarianz))
        *0.5+normpdf(rt(1),x(1),sqrt(startvarianz))*0.5)));
p2t(1)=1-p1t(1);
deltat_1(1)=0;
deltat_2(1)=0;
asymdeltat_1(1)=sqrt(a^2+(deltat_1(1)-0)^2)-x(14)*(deltat_1(1)-0);
asymdeltat_2(1)=sqrt(a^2+(deltat_2(1)-0)^2)-x(15)*(deltat_2(1)-0);
if x(10)~=0
    h1t(1)=(x(2)+x(4)*sqrt(startvarianz)^x(10)*asymdeltat_1(1)^x(12)
            +x(6)*sqrt(startvarianz)^x(10))^(2/x(10));
```

```
else
    h1t(1)=(exp(x(2)+x(4)*asymdeltat_1(1)^x(12)+x(6)*log(sqrt(startvarianz))))^2;
end
if x(11)~=0
    h2t(1)=(x(3)+x(5)*sqrt(startvarianz)^x(11)*asymdeltat_2(1)^x(13)
            +x(7)*sqrt(startvarianz)^x(11))^(2/x(11));
else
    h2t(1)=(exp(x(3)+x(5)*asymdeltat_2(1)^x(13)+x(7)*log(sqrt(startvarianz))))^2;
end
if ~isreal(h1t(1)) || h1t(1)<=0
    h1t(1)=NaN;
end
if ~isreal(h2t(1)) || h2t(1)<=0
    h2t(1)=NaN;
end
p1t_1(1)=normcdf(x(8))*p1t(1)/(normcdf(x(8))*p1t(1)+(1-normcdf(x(9)))*p2t(1));
p1t_2(1)=(1-normcdf(x(8)))*p1t(1)/((1-normcdf(x(8)))*p1t(1)+normcdf(x(9))*p2t(1));
p2t_1(1)=1-p1t_1(1); p2t_2(1)=1-p1t_2(1);
ht_1(1)=p1t_1(1)*h1t(1)+(1-p1t_1(1))*h2t(1);
ht_2(1)=p1t_2(1)*h1t(1)+(1-p1t_2(1))*h2t(1);
for i=2:T-1
    p1t(i)=normcdf(x(8))*(normpdf(rt(i),x(1),sqrt(h1t(i-1)))*p1t(i-1)
        /(normpdf(rt(i),x(1),sqrt(h1t(i-1)))
        *p1t(i-1)+normpdf(rt(i),x(1),sqrt(h2t(i-1)))*p2t(i-1)))+(1-normcdf(x(9))
        *((normpdf(rt(i),x(1),sqrt(h2t(i-1)))*p2t(i-1)
        /(normpdf(rt(i),x(1),sqrt(h1t(i-1)))*p1t(i-1)
        +normpdf(rt(i),x(1),sqrt(h2t(i-1)))*p2t(i-1))));
    p2t(i)=1-p1t(i);
    deltat_1(i)=p1t_1(i-1)*(rt(i)-(x(1)))/(sqrt(h1t(i-1)))
    +p2t_1(i-1)*(rt(i)-(x(1)))/(sqrt(h2t(i-1)));
    deltat_2(i)=p1t_2(i-1)*(rt(i)-(x(1)))/(sqrt(h1t(i-1)))
    +p2t_2(i-1)*(rt(i)-(x(1)))/(sqrt(h2t(i-1)));
    asymdeltat_1(i)=sqrt(a^2+(deltat_1(i))^2)-x(14)*deltat_1(i);
    asymdeltat_2(i)=sqrt(a^2+(deltat_2(i))^2)-x(15)*deltat_2(i);
    if x(10)~=0
        h1t(i)=(x(2)+x(4)*sqrt(ht_1(i-1))^x(10)*asymdeltat_1(i)^x(12)
        +x(6)*sqrt(ht_1(i-1))^x(10))^(2/x(10));
    else
        h1t(i)=(exp(x(2)+x(4)*asymdeltat_1(i)^x(12)+x(6)*log(sqrt(ht_1(i-1)))))^2;
    end
    if x(11)~=0
        h2t(i)=(x(3)+x(5)*sqrt(ht_2(i-1))^x(11)*asymdeltat_2(i)^x(13)
        +x(7)*sqrt(ht_2(i-1))^x(11))^(2/x(11));
    else
        h2t(i)=(exp(x(3)+x(5)*asymdeltat_2(i)^x(13)+x(7)*log(sqrt(ht_2(i-1)))))^2;
    end
```

```
    if ~isreal(h1t(i)) || h1t(i)<=0
        h1t(i)=NaN;
    end
    if ~isreal(h2t(i)) || h2t(i)<=0
        h2t(i)=NaN;
    end
    p1t_1(i)=normcdf(x(8))*p1t(i)/(normcdf(x(8))*p1t(i)+(1-normcdf(x(9)))*p2t(i));
    p1t_2(i)=(1-normcdf(x(8)))*p1t(i)/((1-normcdf(x(8)))*p1t(i)+normcdf(x(9))*p2t(i));
    p2t_1(i)=1-p1t_1(i);
    p2t_2(i)=1-p1t_2(i);
    ht_1(i)=p1t_1(i)*h1t(i)+(1-p1t_1(i))*h2t(i);
    ht_2(i)=p1t_2(i)*h1t(i)+(1-p1t_2(i))*h2t(i);
end
P(1)=0;
for i=2:T
    P(i-1)=log(normpdf(rt(i),x(1),sqrt(h1t(i-1)))*p1t(i-1)
            +normpdf(rt(i),x(1),sqrt(h2t(i-1)))*p2t(i-1));
    if isnan(P) || ~isreal(P)
        P(i-1)=-inf;
    end
end

%The values after this point are not needed for the likelihood but may be
%used for further analysis of the process.

%Calculate the filter-probability for regime 1.
for i=1:T-1
p1tfilter(i)=normpdf(rt(i+1),x(1),sqrt(h1t(i)))*p1t(i)
            /(normpdf(rt(i+1),x(1),sqrt(h1t(i)))*p1t(i)
            +normpdf(rt(i+1),x(1),sqrt(h2t(i)))*p2t(i));
end

%Calculate the ex-post probability for regime 1.
p1tsmooth=NaN(1,T-1); p1tsmooth(T-1)=p1tfilter(T-1); for i=T-2:-1:1
   p1tsmooth(i)=p1tfilter(i)*(normcdf(x(8))*p1tsmooth(i+1)/p1t(i+1)
                +(1-normcdf(x(9)))*(1-p1tsmooth(i+1))/(1-p1t(i+1)));
end

%If you need the function to calculate only the likelihood of the parameter
%vector the next line has to be no comment. If you need the likelihood in
%every single point of time for the qml-estimator than comment the next
%line out.
P=sum(P);
end
```

maximierung_likelihood.m:

```
%ML estimation. At first define the algorithm (BFGS) and some optimization
%conditions. This version is a special case of the general model.
%Input are two time series, one sample group (indexbanken) and one control
%group (indexcontrol).
options=optimset('Algorithm','active-set','MaxIter',400,
                'MaxFunEvals',2200,'TolFun',1e-3);
rt=indexbanken;
T=length(rt);
load('ergebnismatrixtgarchtgarchbanken')
n=size(ergebnismatrixtgarchtgarchbanken,1);
untergrenze=[-0.01 0 0 0 0 0 0 1 1 -1 -1];
obergrenze=[0.01 0.1 0.1 0.2 0.2 1 1 6 6 1 1];

for i=1:n
    if ergebnismatrixtgarchtgarchbanken(i,19)==0
        && isfinite(ergebnismatrixtgarchtgarchbanken(i,18))
        [x,fval,exitflag] = fmincon(@(x) -likelihoodtgarchtgarch(rt,T,x),
                    [ergebnismatrixtgarchtgarchbanken(i,1:9)
                    ergebnismatrixtgarchtgarchbanken(i,16:17)],
                    [],[],[],[],untergrenze,obergrenze,[],options);
        ergebnismatrixtgarchtgarchbanken(i,1:19)=[x(1:9) 1 1 1 1 0 0
                                            x(10:11) -fval exitflag];
    end
end
save('ergebnismatrixtgarchtgarchbanken','ergebnismatrixtgarchtgarchbanken')

load('ergebnismatrixegarchtgarchbanken')
n=size(ergebnismatrixegarchtgarchbanken,1);
untergrenze=[-0.1 -inf 0 -inf 0 -inf 0 1 1 -inf -1];
obergrenze=[0.1 inf 0.2 inf .5 inf 1 6 6 inf 1];
for i=1:n
    if ergebnismatrixegarchtgarchbanken(i,19)==0
        && isfinite(ergebnismatrixegarchtgarchbanken(i,18))
        [x,fval,exitflag] = fmincon(@(x) -likelihoodegarchtgarch(rt,T,x),
                    [ergebnismatrixegarchtgarchbanken(i,1:9)
                    ergebnismatrixegarchtgarchbanken(i,16:17)],
                    [],[],[],[],untergrenze,obergrenze,[],options);
        ergebnismatrixegarchtgarchbanken(i,1:19)=[x(1:9) 0 1 1 1 0 0
                                            x(10:11) -fval exitflag];
    end
end
save('ergebnismatrixegarchtgarchbanken','ergebnismatrixegarchtgarchbanken')

rt=indexcontrol;
```

```
T=length(rt);
load('ergebnismatrixtgarchtgarchcontrol')
n=size(ergebnismatrixtgarchtgarchcontrol,1); untergrenze=[-0.01 0 0 0 0 0 0 1 1 -1 -1];
obergrenze=[0.01 0.1 0.1 0.2 0.2 1 1 6 6 1 1]; for i=1:n
    if ergebnismatrixtgarchtgarchcontrol(i,19)==0
        && isfinite(ergebnismatrixtgarchtgarchcontrol(i,18))
        [x,fval,exitflag] = fmincon(@(x) -likelihoodtgarchtgarch(rt,T,x),
                    [ergebnismatrixtgarchtgarchcontrol(i,1:9)
                    ergebnismatrixtgarchtgarchcontrol(i,16:17)],
                    [],[],[],[],untergrenze,obergrenze,[],options);
        ergebnismatrixtgarchtgarchcontrol(i,1:19)=[x(1:9) 1 1 1 1 0 0
                                            x(10:11) -fval exitflag];
    end
end
save('ergebnismatrixtgarchtgarchcontrol','ergebnismatrixtgarchtgarchcontrol')

load('ergebnismatrixegarchtgarchcontrol')
n=size(ergebnismatrixegarchtgarchcontrol,1);
untergrenze=[-0.1 -inf 0 -inf 0 -inf 0 1 1 -inf -1];
obergrenze=[0.1 inf 0.2 inf .5 inf 1 6 6 inf 1];
for i=1:n
    if ergebnismatrixegarchtgarchcontrol(i,19)==0
        && isfinite(ergebnismatrixegarchtgarchcontrol(i,18))
        [x,fval,exitflag] = fmincon(@(x) -likelihoodegarchtgarch(rt,T,x),
                    [ergebnismatrixegarchtgarchcontrol(i,1:9)
                    ergebnismatrixegarchtgarchcontrol(i,16:17)],
                    [],[],[],[],untergrenze,obergrenze,[],options);
        ergebnismatrixegarchtgarchcontrol(i,1:19)=[x(1:9) 0 1 1 1 0 0
                                            x(10:11) -fval exitflag];
    end
end
save('ergebnismatrixegarchtgarchcontrol','ergebnismatrixegarchtgarchcontrol')
```

startvalues.m:

```
%This program looks for starting values for the parameters of the likelihood
%maximization from our regime switching model. Therefore, we draw a random
%sample out of suitable intervals and compare the likelihood
%functions of these random start values. Output are a selection of this
%random start values. This version is a special case of the general model.
%Input are two time series, one sample group (indexbanken) and one control
%group (indexcontrol).

%Number of random draws per model.
N=20000;
```

```
%Number of how many of the best results should be saved.
n=40;

%Process for which the model should be adjusted.
rt=indexbanken;

%Length of the process.
T=length(rt);

%%%%%%%%%%%%%%%%%%%%%%%TGARCH-TGARCH model
tic
untergrenze=[-0.01 0 0 0 0 0.7 0.7 2 2 -1 -1];
obergrenze=[0.01 0.02 0.02 .1 .1 1 1 3 3 1 1];
zielmatrixfrei=NaN(n,12);
for i=1:n
    zielmatrixfrei(i,12)=-inf;
end
bester=-inf;
anzahl=0;
for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodtgarchtgarch(rt,T,y);
    if rating>bester
        vergleich=zielmatrixfrei(n-1,12);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),12);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,12);
    end
    anzahl=anzahl+isfinite(rating);
end
ergebnismatrixtgarchtgarchbanken(1:n,1:19)=[zielmatrixfrei(:,1:9) ones(n,1)
                        ones(n,1) ones(n,1) ones(n,1) zeros(n,1) zeros(n,1)
                        zielmatrixfrei(:,10:12) zeros(n,1)];
save('ergebnismatrixtgarchtgarchbanken','ergebnismatrixtgarchtgarchbanken')

%%%%%%%%%%%%%%%%%%%%%%%EGARCH-TGARCH model
untergrenze=[-0.01 -0.3 0 -.2 0 -1 0.5 2 2 -1 -1];
obergrenze=[0.01 0.1 0.1 .2 .2 1 1 3 3 1 1];
zielmatrixfrei=NaN(n,12);
for i=1:n
```

```
    zielmatrixfrei(i,12)=-inf;
end
bester=-inf;
anzahl=0;
for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodegarchtgarch(rt,T,y);
    if rating>bester
        vergleich=zielmatrixfrei(n-1,12);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),12);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,12);
    end
    anzahl=anzahl+isfinite(rating);
end
ergebnismatrixegarchtgarchbanken(1:n,1:19)=[zielmatrixfrei(:,1:9) zeros(n,1)
                        ones(n,1) ones(n,1) ones(n,1) zeros(n,1) zeros(n,1)
                        zielmatrixfrei(:,10:12) zeros(n,1)];
save('ergebnismatrixegarchtgarchbanken','ergebnismatrixegarchtgarchbanken')

%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%Now calculate start values for the control group.
%%%%%%%%%%%%%%%

%Process for which the model should be adjusted.
rt=indexcontrol;

%Length of the process.
T=length(rt);

%%%%%%%%%%%%%%%%%%%%%%%%%TGARCH-TGARCH model
untergrenze=[-0.01 0 0 0 0 0.7 0.7 2 2 -1 -1];
obergrenze=[0.01 0.02 0.02 .1 .1 1 1 3 3 1 1];
zielmatrixfrei=NaN(n,12);
for i=1:n
    zielmatrixfrei(i,12)=-inf;
end bester=-inf;
anzahl=0;
for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodtgarchtgarch(rt,T,y);
```

```
    if rating>bester
        vergleich=zielmatrixfrei(n-1,12);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),12);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,12);
    end
    anzahl=anzahl+isfinite(rating);
end
ergebnismatrixtgarchtgarchcontrol(1:n,1:19)=[zielmatrixfrei(:,1:9) ones(n,1)
                        ones(n,1) ones(n,1) ones(n,1) zeros(n,1) zeros(n,1)
                        zielmatrixfrei(:,10:12) zeros(n,1)];
save('ergebnismatrixtgarchtgarchcontrol','ergebnismatrixtgarchtgarchcontrol')

%%%%%%%%%%%%%%%%%%%%%%%EGARCH-TGARCH model
untergrenze=[-0.01 -0.3 0 -.2 0 -1 0.5 2 2 -1 -1];
obergrenze=[0.01 0.1 0.1 .2 .2 1 1 3 3 1 1];
zielmatrixfrei=NaN(n,12); for i=1:n
    zielmatrixfrei(i,12)=-inf;
end bester=-inf; anzahl=0; for i=1:N
    y=[random('unif',untergrenze,obergrenze)];
    rating=likelihoodegarchtgarch(rt,T,y);
    if rating>bester
        vergleich=zielmatrixfrei(n-1,12);
        j=0;
        while rating>vergleich && j<n-1
            zielmatrixfrei(n-j,:)=zielmatrixfrei(n-j-1,:);
            j=j+1;
            vergleich=zielmatrixfrei(max(n-j-1,1),12);
        end
        zielmatrixfrei(n-j,:)=[y rating];
        bester=zielmatrixfrei(n,12);
    end
    anzahl=anzahl+isfinite(rating);
end
ergebnismatrixegarchtgarchcontrol(1:n,1:19)=[zielmatrixfrei(:,1:9) zeros(n,1)
                            ones(n,1) ones(n,1) ones(n,1) zeros(n,1) zeros(n,1)
                            zielmatrixfrei(:,10:12) zeros(n,1)];
save('ergebnismatrixegarchtgarchcontrol','ergebnismatrixegarchtgarchcontrol')
```