Ludwig-Maximilians-Universität München

Mathematisches Institut

Master's Thesis

Default Forward Rate Models for the Valuation of Loans

Markus Parhofer

Under supervision of

Prof. Dr. Christian Fries and

Dr. Andrea Mazzon

March 1, 2024

Contents

1 Introduction

Loans play a significant role not only in the financial world, but also in the lives of private people. Statista shows that the total volume of loans given to private households in 2023 in Germany was roughly 1.5 trillion Euros, which is more than triple the volume of that from 1991 [?]. It can be assumed that the figure for loans granted to companies is significantly higher still.

This gives notion to the importance of loan valuation in financial markets. Understanding and measuring the risks associated with interest rates, but also and especially the ones associated to default probability, is essential for valuing loans correctly.

This master's thesis focuses on extending the existing and well researched discrete forward rate models (short: LIBOR models) to account for default risk, building upon the work of Professor Dr. Christian Fries [?].

These models are then used to value different kind of loan products.

1.1 Motivation

While they are very popular for the valuation of interest rate products, traditional LIBOR models do not account for the risk of default, which is a crucial aspect in financial mathematics.

Using exogenous models to adjust for the probability of default (PD) and a loss given default (LGD) is a work around, that is widely used in praxis. However, there is another solution that directly links the interest rate model with the risk of default.

Prof. Fries' Paper on "Discrete Forward Rate Model with Covariance Structure guaranteeing Positive Credit Spreads" [?] provides a foundation for models that address this gap by incorporating default risk dynamics directly into the LIBOR model.

This begs the question of how to apply these models to pricing and how this impacts prices which are otherwise only driven by interest rates.

1.2 Aim of the thesis

This thesis aims to contribute to the field by describing the discrete forward rate models, developing a valuation methodology with this framework and providing an implementation in the programming language Java for evaluating its usability.

We begin by stating some fundamentals in section 2, which are mainly results from financial mathematics in continuous time.
In section 3 we describe the traditional LIBOR market model, which is given by a set of stochastic differential equations.
Throughout section 4 we describe the model developed by Prof. Fries and derive some implications.
We then develop a general pricing methodology in section 4.4 and derive valuation formulas for some specific products.
Section 5 gives an overview of the actual numerical specification of the model and also displays results gained from the usage.
Finally, in section 7 we conclude the thesis by summarizing key findings, outlining limitations, and offering recommendations for future research.

1.3 Preliminaries

The thesis is aimed at an audience that has a deep analytical background and a basic knowledge in probability theory and financial mathematics.
A list of required fundamental topics may include, but is not limited to

- Brownian Motions and martingales,
- stochastic integration,
- Itô stochastic processes and
- definition of and theorems on arbitrage free markets.
- Monte Carlo methods

Furthermore we assume a fundamental understanding of what these mathematical concepts imply on the real world and the other way around: How are the mathematical concepts motivated by the economical world?

2 Fundamentals

In this section we provide some fundamentals for the thesis. Many of the results mentioned here can also be found in different versions in other scientific papers. We will generally stick to the lecture notes of the course "Stochastic Calculus and Arbitrage Theory in Continuous Time" by Professor Meyer-Brandis at the Ludwig-Maximilians-University in Munich [**?**].

2.1 Probability Theory

In our whole thesis we assume a filtered probability space

$$(\Omega, \mathcal{G}, \mathbb{G}, \mathbb{Q})$$

where

- Ω is the set of all states,
- $\mathcal{G}$ is a σ-algebra,
- $\mathbb{G} = (\mathcal{G}_t)_{t \in [0,\tilde{T}]}$ is a filtration with $\mathcal{G}_t \subset \mathcal{G} \quad \forall t \in [0, \tilde{T}]$, $\mathcal{G}_0 = \{\Omega, \emptyset\}$ and
- $\mathbb{Q}$ is a probability measure on $\mathcal{G}$.

$\mathbb{G}$ is the pricing filtration, which captures all states of all traded assets at each time t. Hence we implicitly assume all market values that we encounter to be $\mathbb{G}$-adapted.
Let us cover some further notations.

Notation 2.1. *Let $X = (X_t)_{t \in [0,\tilde{T}]}$ and $Y = (Y_t)_{t \in [0,\tilde{T}]}$ be Itô stochatic processes and W be a Brownian Motion with:*

$$X_t = X_0 + \int_0^t \mu_s^X ds + \int_0^t \phi_s dW_s,$$
$$Y_t = Y_0 + \int_0^t \mu_s^Y ds + \int_0^t \psi_s dW_s.$$

The sharp bracket *or* quadratic variation *of X is*

$$\langle X \rangle_t = \int_0^t \phi_s^2 ds.$$

The quadratic covariation *of X and Y is*

$$\langle X, Y \rangle_t = \int_0^t \phi_s \psi_s ds.$$

For a n-dimensional Brownian Motion $W = (W^i)_{i\in\{1,\dots,n\}}$ and therefore n-dimensional diffusion processes $\phi = (\phi^i)_{i\in\{1,\dots,n\}}$, $\psi = (\psi^i)_{i\in\{1,\dots,n\}}$ the quadratic covariation is

$$\langle X, Y \rangle_t = \sum_{i=1}^n \int_0^t \phi_s^i \psi_s^i ds$$

For convenience we state a formula for stochastic integration by parts and an extended version of Itô's theorem:

Lemma 2.2. *Let X and Y be two Itô stochastic processes. Then*

$$X_t Y_t = X_0 Y_0 + \int_0^t X_s dY_s + \int_0^t Y_s dX_s + \langle X, Y \rangle_t \quad \text{for } t \in \left[0, \tilde{T}\right]$$

Proof. See [**?**], page 51. □

Theorem 2.3. *Let X be an Itô stochastic process, $f : \mathbb{R}^n \times [0, \tilde{T}] \to \mathbb{R}$, $(x,t) \rightarrowtail f(x,t)$ be two times differentiable in x and differentiable in t. Then*

$$f(X_t, t) = f(X_0, 0) + \int_0^t (\partial_t f)(X_s, s) ds + \sum_{i=1}^n \int_0^t (\partial_{x^i} f)(X_s, s) dX_s^i$$
$$+ \frac{1}{2} \sum_{i,j=1}^n \int_0^t (\partial^2_{x^i x^j} f)(X_s, s) d\left\langle X^i, X^j \right\rangle_s$$

Proof. See [**?**], page 52. □

Through Itô's formula we can prove the following statement.

Lemma 2.4. *Let $W = (W^i)_{i\in\{1,\dots,d\}}$ be a d-dimensional Brownian Motion, μ be a 1-dimensional and $\sigma = (\sigma^i)_{i\in\{1,\dots,d\}}$ a d-dimensional stochastic process. Let following stochastic differential equation (SDE) be given:*

$$\begin{aligned} dY_t &= Y_t \mu_t dt + Y_t \sigma_t \cdot dW_t, \\ Y_0 &= y, \end{aligned} \tag{1}$$

where $y > 0$.

Then the solution of equation (1) *is*

$$\begin{aligned} Y_t &= \exp(X_t), \\ X_t &= X_0 + \int_0^t \mu_s - \frac{1}{2}\sum_{i=1}^d (\sigma_s^i)^2 ds + \int_0^t \sigma_s \cdot dW_s, \\ X_0 &= \log(y). \end{aligned} \tag{2}$$

Proof. We use Itô's formula on equation (2) with $f(x) = \exp(x)$, hence

$$(\partial_x f)(x) = \exp(x), \quad (\partial^2_{xx})(x) = \exp(x).$$

We get (in SDE format):

$$\begin{aligned} dY_t = d\exp(X_t) &= \exp(X_t)dX_t + \frac{1}{2}\exp(X_t)d\left\langle X\right\rangle_t \\ &= Y_t\left(\left(\mu_t - \frac{1}{2}\sum_{i=1}^d(\sigma_t^i)^2\right)dt + \sigma_t \cdot dW_t\right) + \frac{1}{2}Y_t\sum_{i=1}^d(\sigma_t^i)^2 dt \\ &= Y_t\mu_t dt + Y_t\sigma_t \cdot dW_t, \end{aligned}$$

which yields equation (1). □

Remark 2.5. *It is easy to see that because of the relation*

$$Y_t = \exp(X_t),$$

it holds that $Y_t > 0$ *for all* $t \in [0, \tilde{T}]$.

2.2 Financial Mathematics

While the reader should have a deep understanding of what arbitrage is and how to avoid it when pricing financial products, we formulate the following two fundamental results as a reminder.

Theorem 2.6. *Following statements are equivalent:*

- *The market is arbitrage-free and complete.*
- *There exists exactly one probability measure* $\mathbb{Q}^B$ *w.r.t. a numeraire* B*, such that the price process of every traded asset discounted by the numeraire* $\frac{X}{B}$ *is a martingale w.r.t. the filtration* $\mathbb{G}$.

Proof. See [?], page 92 and 93. □

This directly gives us a notation for the price of any product in an arbitrage-free and complete market.

Lemma 2.7. *Let X be the payoff of a T-claim. Then the arbitrage-free price of the claim at any time $t \in [0, T]$ is*

$$\Pi_t^X = B(t)\mathbb{E}^{\mathbb{Q}^B}\left[\left.\frac{X}{B(T)}\right|\mathcal{G}_t\right].$$

Proof. See [?], page 89, 90. □

Let us now cover a concept that we need for numerical purposes.

2.3 Factor Loadings

The problem we face in this section is that of correlated Brownian Motions. Assume we have two SDEs for stochastic processes X and Y:

$$dX_t = \mu_t^X dt + \sigma_t^X dW_t^{\mathbb{Q},1},$$
$$dY_t = \mu_t^Y dt + \sigma_t^Y dW_t^{\mathbb{Q},2},$$

where $W^{\mathbb{Q},1}$ and $W^{\mathbb{Q},2}$ are instantaneously correlated Brownian Motions under the same measure $\mathbb{Q}$. This correlation ρ is expressed by a quadratic covariation of the Brownian Motions [?]:

$$\left\langle W^{\mathbb{Q},1}, W^{\mathbb{Q},2}\right\rangle_t = \int_0^t \rho_s ds$$

or as SDE:

$$d\left\langle W^{\mathbb{Q},1}, W^{\mathbb{Q},2}\right\rangle_t = \rho_t dt.$$

However, implementation-wise we need to simulate SDEs using only independent Brownian Motions.

For our computations we stick to a constant instantaneous correlation ($\rho_t \equiv \rho \in \mathbb{R}$ for all $t \in \left[0, \tilde{T}\right]$), so lets formulate this assumption first:

Assumption 2.1. *For all* 1*-dim. Brownian Motions under the martingale measure* $\mathbb{Q}^B$ *the instantaneous correlation is assumed to be constant. That is for any* $\mathbb{Q}^B$*-Brownian Motions* W^1 *and* W^2 *the following holds for some* $\rho \in \mathbb{R}$*:*

$$d\left\langle W^1, W^2 \right\rangle_t = \rho dt$$

Let us now look at how to simulate correlated Brownian Motions using only independent ones by taking advantage of its properties.
Recall following lemmas:

Lemma 2.8. $W = (W^i)_{i\in\{1,...,d\}}$ *is a d-dimensional Brownian Motion if and only if* $W^1, ..., W^d$ *are independent 1-dimensional Brownian Motions*

Proof. See [?], page 6. □

Lemma 2.9. *For any d-dimensional standard Brownian Motion* $U = (U^i)_{i\in\{1,...,d\}}$ *and weights* $(a_i)_{i\in\{1,...d\}}$ *with* $a_1^2 + ... + a_d^2 = 1$ *it holds that the process* W *given by:*

$$W_t = \sum_{i=1}^{d} a_i U_t^i$$

is a 1*-dimensional standard Brownian Motion.*

Proof. We prove that W fulfills the properties of a Brownian Motion:
Property $W_0 = 0$ *a.s.*:

$$W_0 = \sum_{i=1}^{d} a_i U_0^i = 0 \text{ a.s.}$$

Property $(W_t - W_s) \sim \mathcal{N}(0, t-s)$ *for* $s < t$:

$$\begin{aligned} W_t - W_s = \sum_{i=1}^{d} a_i U_t^i - \sum_{i=1}^{d} a_i U_s^i &\sim \mathcal{N}\left(0, \sum_{i=1}^{d} a_i^2 (t-s)\right) \\ &\sim \mathcal{N}\left(0, (t-s)\right) \end{aligned}$$

Property W *has stationary independent increments*:
Stationarity is given by the last property. Let $0 \leq t_0 < t_1 < ... < t_k \leq \tilde{T}$. Then the stacked vector of the increments of U: $\mathcal{U} = (U_{t_1}^1 - U_{t_0}^1, ..., U_{t_1}^d - U_{t_0}^d, ..., U_{t_k}^1 -$

$U^1_{t_{k-1}}, ..., U^d_{t_k} - U^d_{t_{k-1}})^T$ is a dk-dim. Gaussian vector, with the identity matrix $\mathcal{I}^{dk}$ as correlation (because of the independence). We can define a $k \times dk$-dim. matrix A, such that $A\mathcal{U} = (W_{t_1} - W_{t_0}, ..., W_{t_k} - W_{t_k - 1})^T$ which is a linear transformation of $\mathcal{U}$ and therefore still a multivariate Gaussian vector with $\mathcal{I}^k$ as correlation matrix. A multivariate Gaussian vector, with identity matrix as correlation is independent, hence W has independent increments.

Property W has continuous sample paths a.s.:

Let $U^i(\omega)$ be continuous for $\omega \in A_i \subset \Omega$ where A_i^c is a null set. Then $W(\omega) = \sum_{i=1}^d a_i U^i(\omega)$ is continuous for $\omega \in \bigcap_{i=1}^d A_i$. Because $\left(\bigcap_{i=1}^d A_i\right)^c = \bigcup_{i=1}^d A_i^c$ is still a null set, we have that also W has continuous sample paths almost surely. $\square$

We can now create different linear combinations of the same d-dimensional Brownian Motion U and look at their correlation:

Lemma 2.10. *Let $U = (U^i)_{i\in\{1,...,d\}}$ be a d-dimensional Brownian Motion, let $(a_i)_{i\in\{1,...d\}}$ and $(b_i)_{i\in\{1,...d\}}$ be weights with $\sum_{i=1}^d a_i^2 = 1$, $\sum_{i=1}^d b_i^2 = 1$, respectively. Then W^1 and W^2 given by*

$$W^1_t = \sum_{i=1}^d a_i U^i$$
$$W^2_t = \sum_{i=1}^d b_i U^i$$

are 1-dimensional Brownian Motions with:

$$d\left\langle W^1, W^2 \right\rangle_t = \left(\sum_{i=1}^d a_i b_i\right) dt \tag{3}$$

Proof. By lemma 2.9 we have that W^1 and W^2 are Brownian Motions. Taking the quadratic covariation directly yields equation (3). $\square$

With this lemma we now have a way to construct correlated Brownian Motions $W^1, ..., W^d$. We still need to find the linear combinations, given a certain correlation matrix $R = (\rho_{i,j})_{i,j\in\{1,...,d\}}$, however. A way to do that is to use principal component analysis (PCA). It is beyond the scope of this thesis to give a detailed description of PCA, but the key idea is to take the eigenvectors of R and use them as linear combination. PCA also gives a way to reduce the number of factors, by only considering the Eigenvectors with the highest Eigenvalues [**?**]. Let us summarize this procedure in a lemma:

Lemma 2.11. *Let $U = (U^i)_{i\in\{1,...d\}}$ be a d-dim. Brownian Motion, let $R = (\rho_{i,j})_{i,j\in\{1,...,d\}}$ be a $\mathbb{R}^{d\times d}$ correlation matrix (positive-definite, symmetric, entries in $[-1, 1]$ and 1 on the diagonal). Let $(\lambda^{i,j})_{i,j\in\{1,...,d\}}$ be a matrix constructed by PCA from R.*
For $i = 1, ..., d$ let W^i be the 1-dim. Brownian Motion given by:

$$W_t^i = \sum_{j=1}^{d} \lambda^{i,j} U_t^j.$$

Then for all $i, k \in \{1, ..., d\}$

$$\sum_{j=1}^{d} \lambda^{i,j} \lambda^{k,j} = \rho_{i,k}$$
$$d\left\langle W^i, W^k \right\rangle_t = \rho_{i,k} dt.$$

Proof. This is a direct implication from the previous results together with the definition of PCA and the proof that it works in [Add source]. □

We now have a way of constructing correlated Brownian Motions. As mentioned we can perform a factor reduction with PCA, meaning we can construct d correlated Brownian Motions with a m-dim. Brownian Motion, where $m < d$. This comes at the cost of "losing" a bit of the independence of some factors. But because in most cases the advantages of a factor reduction (less computational cost) outweigh the disadvantages, we from here on assume that the correlation- and factor loading matrix are not of the same size.

3 LIBOR Market Model

The actual LIBOR, short for "London Inter-Bank Offered Rate" phased out in the last year, due to "scandals and questions around its validity as a benchmark rate"[**?**]. However, the LIBOR market model – or discrete forward rate model – is still a very popular mathematical model for simulation and valuation of financial products on fixed income markets.
The idea of the model is to discretize a given time horizon into periods, for each of which different rates hold. The main difference to other models however is, that each rate for a given period is driven by different stochastic parameters.
The basic assumption of the LIBOR Market Model is that we are in an arbitrage free and complete market.

3.1 Fixed Income Markets Terminology

We start with the definition of some fixed income market terms:

Definition 3.1. *A* zero coupon bond *with maturity $T \in [0, \tilde{T}]$ (short: T-bond) is a product that pays* 1 *at maturity. Its price process is denoted:*

$$P(t;T) := P(\omega, t; T).$$

Note: by construction $P(T;T) = 1$ and $P(\cdot, T)$ discounted with the numeraire must be a martingale under the corresponding martingale-measure $\mathbb{Q}^B$.
While the zero coupon bond does not yield any payoff (or coupons) between buying- and maturity time – hence the name – one can also find coupon paying bonds:

Definition 3.2. *Let $T_1 < ... < T_N$ be a tenor with $T_i \in [0, \tilde{T}]$ for all $i \in \{1, ..., N\}$.*
A (fixed) coupon bond *with nominal $\mathcal{N} \in \mathbb{R}$ and coupons $c_i \in \mathbb{R}$ for $i \in \{1, ..., N\}$ on the given tenor is a product that pays c_i at each time point T_i and additionally the nominal $\mathcal{N}$ at maturity T_N.*

Remark 3.3. *A variation of this definition is that c_i are defined as coupon rates and the actual coupon payment is then $c_i \mathcal{N}$ at each time step T_i. Another popular definition includes the terminal payment of the nominal $\mathcal{N}$ in the last coupon c_N. Additionally to this "normal" coupon bond one can also find* amortizing *coupon bonds in the market that distribute the nominal N in the coupons over all periods instead of paying it all at once at the maturity time.*

Lemma 3.4. *Let T_i and c_i be as in definition 3.2. The price of a fixed coupon paying bond is:*

$$\Pi(t) = \sum_{i=1}^{N} c_i P(t; T_i) + \mathcal{N} P(t; T_N) \tag{4}$$

for $t \in [0, T_1[$.

Proof. The coupon bond can be replicated by buying c_i T_i-zero-coupon-bonds for each $i \in \{1, ..., N\}$ and $\mathcal{N}$ T_N-bonds.
Such a portfolio of zero coupon bonds has a price as given in equation (4). As we are in a complete market, the two products must have the same value. □

Definition 3.5. *We define the* simple forward rate $L(t; S, T)$ *with fixing time S and payment time T at evaluation time t to be a relation of S- and T-bonds:*

$$1 + L(t; S, T)(T - S) = \frac{P(t; S)}{P(t; T)}. \tag{5}$$

We can define different products that are strictly positive as numeraires as alternatives to the money market account. We then use a change of measure which gives us a different martingale measure corresponding to the numeraire, i.e. under this new measure all price processes of traded assets discounted with the numeraire are martingales as well.
A simple example is the terminal measure:

Definition 3.6. *The* terminal measure *is the martingale measure $\mathbb{Q}^B$ gained by using the terminal bond as numeraire, i.e.*

$$B(t) = P(t; \tilde{T}). \tag{6}$$

A more complex example is the spot measure.

Definition 3.7. *Let* $0 = T_0 < T_1 < ... < T_N = \tilde{T}$ *be a tenor on the time set* $[0, \tilde{T}]$.

The spot measure *is the martingale measure* $\mathbb{Q}^B$ *gained by using the numeraire:*

$$B(t) = P(t; T_{m(t)+1}) \prod_{i=0}^{m(t)} \frac{1}{P(T_i; T_{i+1})}, \tag{7}$$

where $m(t) = \max\{i \in \{0, ..., N-1\} \mid T_i \leq t\}$.

Remark 3.8. *Note that equation* (7) *can be rewritten to:*

$$B(t) = P(t; T_{m(t)+1}) \prod_{i=0}^{m(t)} \left(1 + L\left(T_{i-1}; T_{i-1}, T_i\right) \cdot \left(T_{i+1} - T_i\right)\right). \tag{8}$$

The numeraire in the spot measure can be explained as follows:
At $T_0 = 0$ we invest 1 into T_1-bonds. Once these expire (at T_1) we reinvest the money gained from them into T_2-bonds and so on. This product is generally known as *rolling bond*.

We now have all fundamentals to specify the model itself.

3.2 Model specification

Through the LIBOR model we construct the stochastic differential equations of simple forward rates for a consecutive set of time periods.
We start with a fixed time tenor of $N + 1$ ($N \in \mathbb{N}$) points, that splits our time horizon $[0, \tilde{T}]$:

$$0 = T_0 < T_1 < ... < T_N = \tilde{T}.$$

The main objective is to simulate the one step simple forward rates for this tenor:

Assumption 3.1. *The one step simple forward rate (called LIBOR rate)*

$$L_i(t) := L(t; T_i, T_{i+1}) \quad \forall i \in \{0, ..., N\},\ t \in [0, T_i],$$

where $L(t;S,T)$ is defined as in definition 3.5, follows an Itô stochastic process satisfying

$$dL_i = \mu_t^i dt + \sigma_t^i dW_t^{\mathbb{Q}^B,L_i}, \tag{9}$$

$$L_i(0)(T_{i+1} - T_i) = \frac{P(0;T_i)}{P(0;T_{i+1})} - 1, \tag{10}$$

where $(W^{\mathbb{Q}^B,L_i})_{i\in\{0,\dots,N\}}$ are possibly instantaneously correlated Brownian Motions.

In the LIBOR model, all other variables are then derived from these interest rates, most importantly the T_i-bond prices. The attentive reader will have noticed, however, that there is a hole in this derivation: the so-called short-period bond $P(t;T_{m(t)+1})$ cannot be calculated by $(L_i(t))_{i\in\{0,\dots,N\}}$ alone, which is why we need the following assumption:

Assumption 3.2. *The short-period bond*

$$P(t;T_{m(t)+1})$$

is $\mathcal{G}_{m(t)}$-measurable. This means that all T_i-bond prices are predictable in $[T_{i-1},T_i]$. A specification that satisfies this assumption is:

$$P(t;T_{m(t)+1}) = \Big(1 + L_{m(t)}(t)\left(T_{m(t)+1} - t\right)\Big)^{-1}$$

Let us fix the spot measure as our valuation measure $\mathbb{Q}^B$. Given a variance-structure for the rates $(\sigma^i)_{i\in\{0,\dots,N\}}$ and a correlation-structure $(\rho^{i,j})_{i,j\in\{0,\dots,N\}}$ for the Brownian Motions we can specify a drift for the SDEs of the LIBORs under the spot measure:

Lemma 3.9. *Let $\mathbb{Q}^B$ be the spot measure. Let L_i satisfy assumptions 3.1 and 3.2. Let $(\sigma^i)_{i\in\{0,\dots,N\}}$ and $(\rho^{i,j})_{i,j\in\{0,\dots,N\}}$ be given, where*

$$d\left\langle W^{\mathbb{Q}^B,L_i}, W^{\mathbb{Q}^B,L_j}\right\rangle_t = \rho_t^{i,j} dt.$$

Then for each $i \in \{1,\dots,N\}$:

$$\mu_t^i = \sigma_t^i \sum_{j=m(t)+1}^{i} \frac{\rho_t^{ij}\sigma_t^j \Delta T_j}{1 + \Delta T_j L_j(t)},$$

Proof. The proof is beyond the scope of this thesis, but can be found in a variety of literature. We reference [?] (pages 301 - 303) and [?], (pages 81 - 86). One can also take a look at the next section at the proof of theorem 4.5, which is very similar and carries its idea and outline. □

Note that in its original form, the LIBOR model is a log-normal model (see [?]), hence the original model assumed $\sigma_t^i = L_i(t)\tilde{\sigma}_t^i$. However all proofs also work on a non log-normal model as well. The only restriction we need to apply is

$$\Delta T_i L_i(t) > -1 \quad \forall i \in \{0, ..., N\}$$

to prevent negative or no zero coupon bond prices.
Other commonly used covariance structures are the displaced log-normal model, where

$$\sigma_t^i = (L_i(t) + d)\,\tilde{\sigma}_t^i$$

with a constant $d \in \mathbb{R}$ and the blended covariance structure

$$\sigma_t^i = ((1-\alpha)L_i(t) + \alpha)\,\tilde{\sigma}_t^i$$

with $\alpha \in [0, 1]$.

As mentioned in the previous chapter, having a model in terms of correlated Brownian Motions is not ideal for numerical replication. We therefore move to another notation, which uses factor loadings instead:

Lemma 3.10. *There exist m-dimensional stochastic processes $\lambda^i = (\lambda^{ik})_{k\in\{1,...m\}}$ and a m-dimensional $\mathbb{Q}^B$-Brownian Motion $U = (U^k)_{k\in\{1,...,m\}}$ such that*

$$dL_i(t) = \mu_t^i dt + \sigma_t^i dW_t^{\mathbb{Q}^B, L_i} \iff dL_i(t) = \mu_t^i dt + \lambda_t^i \cdot dU_t$$

Furthermore

$$\sum_{k=1}^{m} \lambda_t^{ik}\lambda_t^{ik} = (\sigma_t^i)^2$$

as well as

$$\sum_{k=1}^{m} \lambda_t^{ik}\lambda_t^{jk} = \sigma_t^i \sigma_t^j \rho_t^{i,j}$$

Proof. Follows directly from section 2.3. □

With this notation we can also rewrite our drift term of the LIBOR rates:

Remark 3.11. *For the spot measure the drifts μ^i in equation* (9) *can be rewritten in terms of λ:*

$$\mu_t^i = \sum_{k=1}^{m} \lambda_t^{ik} \sum_{j=m(t)}^{i} \frac{\lambda_t^{jk} \Delta T_j}{1 + L_j(t) \Delta T_j}$$

Proof. By lemmas 3.9 and 3.10 we have:

$$\begin{aligned}
\mu_t^i = \sigma_t^i \sum_{j=m(t)+1}^{i} \frac{\rho_t^{ij} \sigma_t^j \Delta T_j}{1 + \Delta T_j L_j(t)} &= \sum_{j=m(t)+1}^{i} \frac{\rho_t^{ij,d} \sigma_t^{j,d} \sigma_t^i \Delta T_j}{1 + \Delta T_j L_j(t)} \\
= \sum_{j=m(t)+1}^{i} \frac{\left(\sum_{k=1}^{m} \lambda_t^{ik} \lambda_t^{jk}\right) \Delta T_j}{1 + \Delta T_j L_j(t)} &= \sum_{k=1}^{m^d} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{ik} \lambda_t^{jk} \Delta T_j}{1 + \Delta T_j L_j(t)} \\
= \sum_{k=1}^{m} \lambda_t^{ik} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk} \Delta T_j}{1 + \Delta T_j L_j(t)}
\end{aligned}$$

□

With this model one can go on and price interest rate products, which is done in praxis. However, the products priced – assuming no external model is used – would not take any possibility of default into account. This is what is discussed in the next chapter.

4 Defaultable LIBOR Market Models

In this section we introduce defaultable LIBOR market models that we can use to value credits and credit options.
Our main source for this section is the article "Defaultable Discrete Forward Rate Model with Covariance Structure guaranteeing Positive Credit Spreads" authored by Professor Christian Fries [**?**].

4.1 The Defaultable Forward Rate

We remain in the same setting as in the non-defaultable model, where we have a LIBOR tenor discretization $(T_i)_{i\in\{0,1,\dots,N\}}$ and a set of (non-defaultable) zero coupon bonds $(P(t;T_i))_{i\in\{0,1,\dots,N\}}$. Hence we can define the same products and apply the same valuation formulas. This also means we have to apply the same numeraire for pricing, as in the non-defaultable model.
We extend the model by defining an additional set of zero coupon bonds which are defaultable: $(P^d(t;T_i))_{i\in\{0,\dots,N\}}$.
Furthermore we will model only bonds and products that pay nothing if defaulted. However, products considering recovery rates can then be derived by applying a linear combination of non-defaultable and defaultable values.
We now introduce the concept of default and defaultable zero coupon bonds.

Definition 4.1. *The* default time *is a stopping time $\tau(\omega)$ on the filtration $(\mathcal{G}_t)_{t\in\mathbb{R}^+}$.*
The default indicator $J(t)$ *is the indicator process over the default time:*

$$J(t) := \mathbf{1}_{\{\tau(\omega)\leq t\}}$$

Definition 4.2. *The* Defaultable Zero Coupon Bond *with price process*

$$P^d(t;T_i)$$

at time $t\in[0,T]$ is a traded asset that pays $1-J(T_i)$ at maturity $T_i\in\{T_0,\dots,T_N\}$. Hence it pays 1 if the default has not happened until maturity.

It is easy to see that if default occurs, the price of a defaultable zero coupon bond jumps to zero. This means that the price process can be discontinuous at default events. It gives notion to the definition of a zero coupon bond conditional on pre-default.

Definition 4.3. *The* Defaultable Zero Coupon Bond conditional pre-default *is a continuous Itô-stochastic process $P^{d,*}(t;T_i)$ at time $t \in [0,T_i]$ with maturity T_i ($i \in \{0,...,N\}$) such that*

$$P^d(t;T_i) = P^{d,*}(t;T_i)(1 - J(t))$$

Definition 4.4. *The* simple Defaultable Forward Rate *is the rate gained from $P^{d,*}(t;T)$ by the same concept as in a non-defaultable model:*

$$L_i^d(t) := L^d(t;T_i,T_{i+1}) := \left(\frac{P^{d,*}(t;T_i)}{P^{d,*}(t;T_{i+1})} - 1\right)\Delta T_i, \tag{11}$$

where $T_i \in \{T_0, ...T_N\}$.

Note here that while L^d follows the same concept as L, we will use it only as an abstract model, not as an actual rate that can be received.

Assumption 4.1. *As in the non defaultable model we assume that the defaultable short period bond conditional pre-default*

$$P^{d,*}(t;T_{m(t)+1})$$

has no diffusion. This means that the only stochasticity on the defaultable short period bond is the default time.
I.e. we specify the defaultable short period bond as:

$$P^{d,*}(t;T_{m(t)+1}) = (1 + L_{m(t)}^d(t)(T_{m(t)+1} - t))^{-1}$$

Theorem 4.5. *Let L_i^d be defined as in equation (11). Let $B(t)$ be the numeraire under the spot measure (i.e. $B(t)$ is given by equation (7)) and $W^{\mathbb{Q}^B}$ a Brownian Motion w.r.t. the spot measure. Let $\sigma_t^{i,d} := \sigma^{i,d}(t,\omega)$ be a progressive stochastic process. Let*

$$dL_i^d(t) = \mu_t^{i,d}dt + \sigma_t^{i,d}dW_t^{\mathbb{Q}^B,L_i^d} \tag{12}$$

be the stochastic differential of L_i^d.

Then

$$\mu_t^{i,d} = \sigma_t^{i,d} \sum_{j=m(t)+1}^{i} \frac{\rho_t^{ij,d} \sigma_t^{j,d} \Delta T_j}{1 + \Delta T_j L_j^d(t)}, \tag{13}$$

where $\rho_t^{ij,d} = d\left\langle W_t^{\mathbb{Q}^B, L_i^d}, W_t^{\mathbb{Q}^B, L_j^d} \right\rangle$.

Proof. By construction the defaultable zero coupon bond $P^d(t;T_i)$ is a traded asset. We get

$$L_i^d(t) P^d(t;T_{i+1}) = (P^d(t;T_i) - P^d(t;T_{i+1}))\Delta T_i$$

is also a traded asset, because it is a portfolio of defaultable zero coupon bonds. Hence both processes discounted with the numeraire $B(t)$ are martingales.

By Itô we have:

$$\begin{aligned} d\left(L_i^d(t)\frac{P^d(t;T_{i+1})}{B(t)}\right) &= L_i^d(t) d\left(\frac{P^d(t;T_{i+1})}{B(t)}\right) + \frac{P^d(t;T_{i+1})}{B(t)} dL_i^d(t) \\ &\quad + d\left\langle L_i^d(t), \frac{P^d(t;T_{i+1})}{B(t)}\right\rangle. \end{aligned}$$

We analyze the drift terms on each diffusion:

$$d\left(L_i^d(t)\frac{P^d(t;T_{i+1})}{B(t)}\right) \quad \text{and} \quad d\left(\frac{P^d(t;T_{i+1})}{B(t)}\right)$$

are martingale diffusions and hence have no drift. Therefore the drift terms of the two remaining differentials must cancel each other out. I.e.:

$$\frac{P^d(t;T_{i+1})}{B(t)} \mu_t^{i,d} dt \stackrel{!}{=} -d\left\langle L_i^d(t), \frac{P^d(t;T_{i+1})}{B(t)}\right\rangle. \tag{14}$$

To calculate the quadratic variation we need the diffusion of the discounted defaultable zero coupon bond:

$$\begin{aligned} d\left(\frac{P^d(t;T_{i+1})}{B(t)}\right) &= d\left(\frac{P^d(t;T_{m(t)+1})}{B(t)} \prod_{j=m(t)+1}^{i} (1+\Delta T_j L_j^d(t))^{-1}\right) \\ &= (\ldots)dt - \frac{P^d(t;T_{m(t)+1})}{B(t)} \sum_{j=m(t)+1}^{i} \frac{\sigma_t^{j,d}\Delta T_j}{1+\Delta T_j L_j^d(t)} dW_t^{\mathbb{Q}^B, L_j^d} \\ &\quad + \prod_{j=m(t)+1}^{i} (1+\Delta T_j L_j^d(t))^{-1} d\left(\frac{P^d(t;T_{m(t)+1})}{B(t)}\right) \\ &\quad + d\left\langle \frac{P^d(t;T_{m(t)+1})}{B(t)}, \prod_{j=m(t)+1}^{i} (1+\Delta T_j L_j^d(t))^{-1}\right\rangle. \end{aligned}$$

With assumptions 3.2 and 4.1 we get

$$d\left(\frac{P^d(t;T_{m(t)+1})}{P(t;T_{m(t)+1})}\right) = (\ldots)dt - \frac{P^{d,*}(t;T_{m(t)+1})}{P(t;T_{m(t)+1})}dJ(t).$$

Furthermore we have

$$d\left(\frac{P^d(t;T_{m(t)+1})}{B(t)}\right) = \prod_{j=0}^{m(t)}(1+\Delta T_j L_j(T_j))^{-1} d\left(\frac{P^d(t;T_{m(t)+1})}{P(t;T_{m(t)+1})}\right)$$

and for any Itô-process X: $d\langle X, J\rangle = 0$ [Add source].

This yields

$$\begin{aligned} d\left(\frac{P^d(t;T_{i+1})}{B(t)}\right) = (\ldots)dt - \frac{P^d(t;T_{m(t)+1})}{B(t)} \sum_{j=m(t)+1}^{i} \frac{\sigma_t^{j,d}\Delta T_j}{1+\Delta T_j L_j^d(t)} dW_t^{\mathbb{Q}^B, L_j^d} \\ - \frac{P^{d,*}(t;T_{i+1})}{P(t;T_{m(t)+1})}dJ(t). \end{aligned}$$

We get

$$d\left\langle L_i^d(t), \frac{P^d(t;T_{i+1})}{B(t)}\right\rangle = \sigma_t^{i,d}\frac{P^d(t;T_{m(t)+1})}{B(t)} \sum_{j=m(t)+1}^{i} \frac{\sigma_t^{j,d}\Delta T_j}{1+\Delta T_j L_j^d(t)}\rho_t^{ij,d}dt.$$

Inserting into equation (14) yields our statement (13). □

Just as we did in the last section, we now move from the "covariance" process model to a "factor loading" model as described in section 2.3. Note that we also need to include the non defaultable covariance structure for our new model.

Lemma 4.6. *There exist m^d-dimensional stochastic processes $\lambda^{i,d} = (\lambda^{ik,d})_{k\in\{1,\ldots,m^d\}}$, $\lambda^i = ((\lambda^{ik})_{k\in\{1,\ldots m\}}, (0)_{k\in\{m+1,\ldots m^d\}})$ and a m^d-dimensional $\mathbb{Q}^B$-Brownian Motion $U = (U^k)_{k\in\{1,\ldots,m^d\}}$ such that*

$$\left.\begin{aligned} dL_i(t) &= \mu_t^i dt + \sigma_t^i dW_t^{\mathbb{Q}^B, L_i} \\ dL_i^d(t) &= \mu_t^{i,d} dt + \sigma_t^{i,d} dW_t^{\mathbb{Q}^B, L_i^d} \end{aligned}\right\} \iff \left\{\begin{aligned} dL_i(t) &= \mu_t^i dt + \lambda_t^i \cdot dU_t \\ dL_i^d(t) &= \mu_t^{i,d} dt + \lambda_t^{i,d} \cdot dU_t \end{aligned}\right.$$

Furthermore

$$\begin{aligned} \sum_{k=1}^{m^d} \lambda_t^{ik,d}\lambda_t^{ik,d} &= (\sigma_t^i)^2 \quad \textit{and} \\ \sum_{k=1}^{m^d} \lambda_t^{ik,d}\lambda_t^{jk,d} &= \sigma_t^i\sigma_t^j\rho_t^{i,j}. \end{aligned}$$

Proof. Follows directly from section 2.3. □

Hence from here on we use this computation friendly version:

$$dL_i^d(t) = \mu_t^{i,d}dt + \lambda_t^{i,d} \cdot dU_t. \tag{15}$$

Remark 4.7. *For the spot measure the drifts $\mu^{i,d}$ in equation* (15) *can be rewritten in terms of λ^d:*

$$\mu_t^{i,d} = \sum_{k=1}^{m^d} \lambda_t^{ik,d} \sum_{j=m(t)}^{i} \frac{\lambda_t^{jk,d}\Delta T_j}{1 + L_j^d(t)\Delta T_j}.$$

Proof. By equation (13) and lemma 4.6 we have:

$$\begin{aligned}
\mu_t^{i,d} &= \sigma_t^{i,d} \sum_{j=m(t)+1}^{i} \frac{\rho_t^{ij,d}\sigma_t^{j,d}\Delta T_j}{1+\Delta T_j L_j^d(t)} = \sum_{j=m(t)+1}^{i} \frac{\rho_t^{ij,d}\sigma_t^{j,d}\sigma_t^{i,d}\Delta T_j}{1+\Delta T_j L_j^d(t)} \\
&= \sum_{j=m(t)+1}^{i} \frac{\left(\sum_{k=1}^{m^d} \lambda_t^{ik,d}\lambda_t^{jk,d}\right)\Delta T_j}{1+\Delta T_j L_j^d(t)} = \sum_{k=1}^{m^d} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{ik,d}\lambda_t^{jk,d}\Delta T_j}{1+\Delta T_j L_j^d(t)} \\
&= \sum_{k=1}^{m^d} \lambda_t^{ik,d} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk,d}\Delta T_j}{1+\Delta T_j L_j^d(t)}
\end{aligned}$$

□

4.2 Covariance Structures guaranteeing positive Spreads

We now investigate how we can generate positive spreads from the defaultable LIBOR market model.

For this purpose let us define a spread:

Definition 4.8. *Let L^d and L be defined as before. The* spread *S_i for the LIBOR period $[T_i; T_{i+1}]$, where $i \in \{0, ..., N\}$ is defined as:*

$$S_i(t) := L_i^d(t) - L_i(t) \quad \textit{for } t \in \left[0, \tilde{T}\right]$$

Remark 4.9. *A negative spread would mean $L_i^d(t) < L_i(t)$. This constitutes an arbitrage possibility, which is why the model needs to be specified, such that this case is "impossible" (i.e. has probability 0).*

The spreads dynamics are given by

$$dS_i(t) = \mu_t^{i,S} dt + \lambda_t^{i,S} \cdot dU_t$$

where

$$\mu_t^{i,S} = \mu_t^{i,d} - \mu_t^i, \quad \text{and} \quad \lambda_t^{i,S} = \lambda_t^{i,d} - \lambda_t^i.$$

Note that we "extended" the vectors $\lambda_t^i = (\lambda_t^{i1}, ..., \lambda_t^{im}, 0, ..., 0)^T$.
Given a numeraire and the factor loadings of the non-defaultable LIBOR rates λ^i, the goal is, to find a specification for the defaultable factor loadings $\lambda^{i,d}$ such that S_i is always positive.
The most common dynamics that guarantee positivity are log-normal dynamics as discussed in lemma 2.4. So one idea is to find restrictions on $\lambda_t^{i,d}$ such that

$$\mu_t^{i,d} - \mu_t^i = S_i(t)\tilde{\mu}_t^{i,S}, \quad \text{and} \quad \lambda_t^{i,d} - \lambda_t^i = S_i(t)\tilde{\lambda}_t^{i,S}$$

for two processes $\tilde{\mu}^{i,S}$ and $\tilde{\lambda}^{i,S}$.

Lemma 4.10. *Let $\mathbb{Q}^B$ be the spot measure (i.e. $B(t)$ is given by equation (7)). Let L^d be defined as before with*

$$\begin{aligned} \lambda_t^{ik,d} &= \frac{1 + L_i^d \Delta T_i}{1 + L_i \Delta T_i} \lambda_t^{ik} & \text{for } k = 1, ..., m \\ \lambda_t^{ik,d} &= \left(L_i^d(t) - L_i(t)\right) f_t^{ik} & \text{for } k = m+1, ...m^d \end{aligned} \tag{16}$$

where f_t^{ik} are (possibly stochastic) processes for $i \in \{1, ..., N-1\}$ and $k \in \{m + 1, ..., m^d\}$.
Then $S = (S_i)_{i \in \{0,...,N-1\}}$ satisfies

$$dS_i = S_i(t)\tilde{\mu}_t^{i,S} dt + S_i(t)\tilde{\lambda}_t^{i,S} \cdot dU_t.$$

for some process $\tilde{\mu}^{i,S}$ and $\tilde{\lambda}^{i,S}$.

Proof. By remark 4.7 we have:

$$
\begin{aligned}
dS_i &= dL_i^d - dL_i \\
&= \sum_{k=1}^{m^d} \lambda_t^{ik,d} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk,d}\Delta T_j}{1+L_j^d(t)\Delta T_j} dt + \sum_{k=1}^{m^d} \lambda_t^{ik,d} dU_t^k \\
&\quad - \sum_{k=1}^{m} \lambda_t^{ik} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk}\Delta T_j}{1+L_j(t)\Delta T_j} dt - \sum_{k=1}^{m} \lambda_t^{ik} dU_t^k \\
&= \sum_{k=1}^{m} \left(\lambda_t^{ik,d} - \lambda_t^{ik}\right) \left(\sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk}\Delta T_j}{1+L_j(t)\Delta T_j} dt + dU_t^k \right) && \text{(A)} \\
&\quad + \left(L_i^d(t) - L_i(t)\right) \left(\sum_{k=m+1}^{m^d} f_t^{ik} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk,d}\Delta T_j}{1+L_j^d(t)\Delta T_j} dt + dU_t^k \right) \\
&= S_i \sum_{k=1}^{m} \frac{\lambda_t^{ik}\Delta T_i}{1+L_i(t)\Delta T_i} \left(\sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk}\Delta T_j}{1+L_j(t)\Delta T_j} dt + dU_t^k \right) && \text{(B)} \\
&\quad + S_i \left(\sum_{k=m+1}^{m^d} f_t^{ik} \sum_{j=m(t)+1}^{i} \frac{\lambda_t^{jk,d}\Delta T_j}{1+L_j^d(t)\Delta T_j} dt + dU_t^k \right)
\end{aligned}
$$

where (A) comes from the relation:

$$
\frac{\lambda_t^{jk,d}\Delta T_j}{1+L_j^d(t)\Delta T_j} = \frac{\lambda_t^{jk}\Delta T_j}{1+L_j(t)\Delta T_j} \quad \text{for } k \in \{1,...,m\}
$$

and (B) comes from

$$
\lambda_t^{ik,d} - \lambda_t^{ik} = \left(\frac{1+L_i^d(t)\Delta T_i}{1+L_i(t)\Delta T_i} - \frac{1+L_i(t)\Delta T_i}{1+L_i(t)\Delta T_i} \right) \lambda_t^{ik} = S_i \frac{\lambda_t^{ik}\Delta T_i}{1+L_i(t)\Delta T_i}
$$

for $k \in \{1, ..., m\}$. □

This gives us a whole set of covariance structures for the defaultable LIBOR model, which guarantee positivity of the spread. Note that there might be other structures that also guarantee positivity.

The next sections can be applied to any such model and do not depend on a covariance model that is structured like the one that was derived.

We now move on to deriving the survival probability, which we need for pricing.

4.3 The Survival Probability

Until now we assumed every stochastic process to be adapted to the filtration $\mathbb{G}$. This is what is commonly done in mathematical finance to find theoretical prices and it simulates the availability of information at each time t. This includes the indicator over the default time $J(t)$. However the products we are interested in, are rarely dependent on the (future) default state, but rather its probability, as we see in the next chapter. So it is more efficient (and less dependent on extra assumptions) to calculate the survival probability instead of simulating the default time.

It is important to note here that for pricing we are only interested in the behavior of the rates under the martingale measure $\mathbb{Q}^B$ and so we also only focus on the survival probability under this measure. This means the probability we derive, is not to be confused with the real world probability of survival.

As we work quite a lot with conditional probability and expectation we denote for a σ-algebra $\mathcal{F}$:

$$\mathbb{E}_{\mathcal{F}}^{\mathbb{Q}^B}[\,\cdot\,] = \mathbb{E}^{\mathbb{Q}^B}[\,\cdot \mid \mathcal{F}\,]$$
$$\mathbb{Q}_{\mathcal{F}}^{B}(\,\cdot\,) = \mathbb{Q}^{B}(\,\cdot \mid \mathcal{F}).$$

With the assumption that the short period bond is deterministic w.r.t. the previous LIBOR time, we can derive the price of a zero-coupon bond that only lives inside a LIBOR period:

Lemma 4.11. *Let assumptions 3.2 and 4.1 be given. For $T_i \leq s < t \leq T_{i+1}$ it holds that*

- $P(s;T_{i+1}) = P(s;t)P(t;T_{i+1})$ *or* $P(s;t) = \frac{P(s;T_{i+1})}{P(t;T_{i+1})}$ *and*
- $P^{d,*}(s;T_{i+1}) = P^{d,*}(s;t)P^{d,*}(t;T_{i+1})$ *or* $P^{d,*}(s;t) = \frac{P^{d,*}(s;T_{i+1})}{P^{d,*}(t;T_{i+1})}$ *for* $\tau > s$.

Proof. Let $T_i \leq s < t \leq T_{i+1}$. With assumption 3.2 we have:

$$\begin{aligned} P(s;T_{i+1}) &= B(s)\mathbb{E}_{\mathcal{G}_s}^{\mathbb{Q}^B}\left[\frac{P(t;T_{i+1})}{B(t)}\right] = P(t;T_{i+1})B(s)\mathbb{E}_{\mathcal{G}_s}^{\mathbb{Q}^B}\left[\frac{P(t;t)}{B(t)}\right] \\ &= P(t;T_{i+1})P(s;t) \end{aligned}$$

We can do a very similar derivation for $P^{d,*}(s;t)$ with assumption 4.1:

$$\begin{aligned}P^{d,*}(s;T_{i+1})\mathbf{1}_{\{\tau>s\}} &= B(s)\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_s}\left[\frac{P^d(t;T_{i+1})}{B(t)}\right]\\ &= P^{d,*}(t;T_{i+1})B(s)\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_s}\left[\frac{\mathbf{1}_{\{\tau>t\}}}{B(t)}\right]\\ &= P^{d,*}(t;T_{i+1})B(s)\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_s}\left[\frac{P^d(t;t)}{B(t)}\right]\\ &= P^{d,*}(t;T_{i+1})P^{d,*}(s;t)\mathbf{1}_{\{\tau>s\}}\end{aligned}$$

□

With this property we can express the probability for the survival inside a LIBOR period w.r.t. the spot measure:

Lemma 4.12. *Let $\mathbb{Q}^B$ be the spot measure. For $T_i \leq s < t \leq T_{i+1}$ it holds that:*

$$\mathbb{Q}^B_{\mathcal{G}_s}\left(\{\tau>t\}\middle|\{\tau>s\}\right) = \frac{P^{d,*}(s;t)}{P(s;t)}$$

Proof. Let $T_i \leq s < t \leq T_{i+1}$. From the martingale property of $\frac{P^d(s;t)}{B(s)}$ we have:

$$\begin{aligned}\frac{P^{d,*}(s;t)}{P(s;t)} &= \mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_s}\left[\frac{B(s)}{B(t)}\right]^{-1} B(s)\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_s}\left[\frac{\mathbf{1}_{\{\tau>t\}}}{B(t)}\middle|\{\tau>s\}\right]\\ &= \left(\frac{B(s)}{B(t)}\right)^{-1}\frac{B(s)}{B(t)}\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_s}\left[\mathbf{1}_{\{\tau>t\}}\middle|\{\tau>s\}\right]\\ &= \mathbb{Q}^B_{\mathcal{G}_s}\left(\{\tau>t\}\middle|\{\tau>s\}\right)\end{aligned}$$

□

Setting $s = T_i$ and $t = T_{i+1}$ gives us the one step survival probability:

Lemma 4.13. *Let B be the numeraire under the spot measure, $i \in \{1,...,N\}$. The one-step survival probability until T_{i+1} w.r.t. $\mathcal{G}_{T_i}$ given pre-default is*

$$q(T_{i+1},\omega) := \mathbb{Q}^B_{\mathcal{G}_{T_i}}\left(\{\tau(\omega)>T_{i+1}\}\mid\{\tau(\omega)>T_i\}\right) = \frac{P^{d,*}(T_i;T_{i+1})}{P(T_i;T_{i+1})}. \tag{17}$$

Proof. Can directly be derived from the lemma above with $s = T_i$ and $t = T_{i+1}$. □

Note that while we calculate a probability here, it is a *pathwise* probability: at each start of a LIBOR period T_i – if default has not happened yet – we can evaluate how probable it is that P^d survives the coming period.
Let us define a new value:

Definition 4.14. *Let $i \in \{1, ..., N\}$. We define the random variable Q as*

$$Q(T_i, \omega) = \prod_{j=1}^{i} q(T_j, \omega)$$

While Q in theory is not a probability, we can view it as the pathwise survival probability until T_i, if we have no information regarding the actual default state even at time T_i. This, however; is only an *interpretation* and the actual proof for this statement needs extra assumptions.
What we actually prove in the next chapter is that the total survival probability until T_i is given by the expectation of Q:

$$\mathbb{Q}^B\left(\{\tau > T_i\}\right) = \mathbb{E}^{\mathbb{Q}^B}\left[Q(T_i; \omega)\right]$$

Let us rewrite the before derived/defined values in terms of L and L^d:

Remark 4.15. *Note that*

$$q(T_{i+1}, \omega) = \frac{1 + \Delta T_i L_i(T_i)(\omega)}{1 + \Delta T_i L_i^d(T_i)(\omega)}.$$

Furthermore, with the definition of $q(T_i, \omega)$ one can rewrite Q in terms of the numeraire $B(T_i)$ (corresponding to the Spot measure):

$$Q\left(T_{i+1}, \omega\right) = \frac{B(T_i)}{\prod_{k=0}^{i}(1 + \Delta T_k L_k^d(T_k)(\omega))} =: \frac{B(T_i)}{B^d(T_i)}.$$

Note that $Q(T_i, \omega)$ is $\mathcal{G}_{T_{i-1}}$-measurable.

The denominator in this equation can be seen as a "defaultable numeraire", which we define next:

Definition 4.16. *We call B^d given by:*

$$B^d(t) = P^{d,*}(t; T_{m(t)+1}) \prod_{k=1}^{m(t)} \frac{1}{P^{d,*}(T_k; T_{k+1})} = P^{d,*}(t; T_{m(t)+1}) \prod_{k=1}^{m(t)} (1 + \Delta T_k L_k^d(T_k))$$

the defaultable numeraire.

Due to B^d not being a traded product, it is not a real numeraire. Hence there is not an equivalent martingale measure w.r.t. B^d that we can use to price other products. And yet, we will see that the valuation formula for defaultable products is similar to a change of numeraires with B^d. Therefore, let us look at valuation in more detail.

4.4 Valuation Methodology

A model is useless without a practical application for it. Hence we take a look at how to apply our results to pricing and valuation.
Note here that the model is for use in numerical option pricing, so deriving prices in terms of an expectation over model primitives (directly simulated values or values that can be derived from these) is sufficient. We can then use Monte Carlo methods to approximate the expectation as we will see in section 6.2.
For a start let us define our simulation filtration:

Definition 4.17. *The filtration* $\mathbb{F} := (\mathcal{F}_t)_{t\in[0,\tilde{T}]}$ *is given by*

$$\mathcal{F}_t := \sigma(U_t^k | k = 1, ..., m^d) \subset \mathcal{G}_t \quad \forall t \in [0, \tilde{T}].$$

Note that all model primitives are $\mathbb{F}$-adapted, as $\mathbb{F}$ is the filtration over the only simulated stochastic processes (in our case the Brownian Motion U). As mentioned before we are interested in pricing without simulating the default time τ, which might be driven by other stochastic values. This means that τ is not necessarily a stopping time w.r.t. $\mathbb{F}$. This is illustrated in figure 1.
While the model simulates only paths of $P^{d,*}(t; T_i)$ (in the figure $P^{d,*}(t; 4.0)$), there are indefinitely many paths of $P^d(t; T_i)$ (in the figure $P^d(t; 4.0)(\omega_j)$ for $j \in \{1, 2, 3\}$) which satisfy

$$P^d(t; T_i) = P^{d,*}(t; T_i)(1 - J(t)),$$

but are not equal.
We therefore use a trick and replace the default state $1 - J(t)$ in all our pricing formulas with the random variable Q that we defined in the previous chapter.

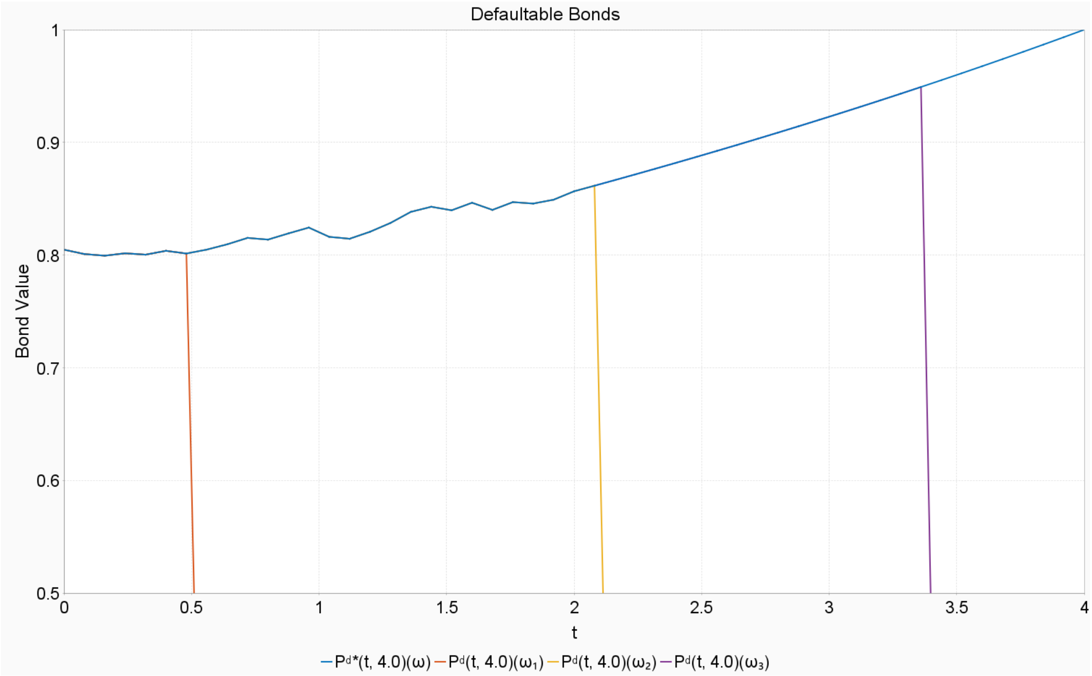

Figure 1: Paths of P^d

4.4.1 Products depending on a single defaultable party

We start with the simple case of a single defaultable party. To prove this works, assume a T_i-claim X that has a payoff that is dependent on survival of the defaultable bonds. We can then use the pricing methodology described in the following lemma.

Lemma 4.18. *Let X be a payoff at $T_i \in \{T_1, ..., T_N\}$, with*

$$X = X^S \mathbf{1}_{\{\tau(\omega) > T_i\}},$$

where X^S is the payoff conditional pre-default. Let X^S be $\mathcal{F}_{T_i}$-measurable. Then the price at time $t = 0$ is

$$\Pi^X = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)} Q\left(T_i, \omega\right)\right].$$

Note that by construction $Q(T_i, \omega)$ is $\mathcal{F}_{T_i}$-measurable, which means, we have a price in terms of model primitives. Let us now prove this statement.

Proof. For our proof we use following property of the conditional expectation extensively:

$$\mathbb{E}^{\mathbb{Q}^B}\left[Y\right] = \mathbb{E}^{\mathbb{Q}^B}\left[Y|A\right]\mathbb{Q}^B(A) + \mathbb{E}^{\mathbb{Q}^B}\left[Y|A^c\right]\mathbb{Q}^B(A^c) \qquad \text{(A)}$$

for any set $A \in \mathcal{G}$.

For ease of notation we write $q(T_i) := q(T_i, \omega)$ and $Q(T_i) := Q(T_i, \omega)$. Note $B(T_0) = 1$. We have:

$$
\begin{aligned}
\Pi^X &= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\right] \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\middle| \{\tau > T_1\}\right] q(T_1) && \text{(B)} \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_{T_1}}\left[q(T_1)\frac{X^S}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\middle| \{\tau > T_1\}\right]\right] && \text{(C)} \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_{T_1}}\left[q(T_1)\frac{X^S}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\middle| \{\tau > T_2\}\right] q(T_2)\right] \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_{T_2}}\left[Q(T_2)\frac{X^S}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\middle| \{\tau > T_2\}\right]\right] \\
&= \dots \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_{T_i}}\left[Q(T_i)\frac{X^S}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\middle| \{\tau > T_i\}\right]\right] && \text{(D)} \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[Q(T_i)\frac{X^S}{B(T_i)}\right], && \text{(E)}
\end{aligned}
$$

where (B) comes from (A) and (C) comes from the tower property and the fact that $q(T_1)$ is a constant. Then we repeat these steps until equation (D) always keeping in mind $q(T_j)$ is $\mathcal{G}_{T_i}$-measurable for all $j \leq i$. □

Remark 4.19. *Note that in (E) of the proof we omitted the condition on $\{\tau > T_i\}$. This means that X^S is independent of this set. We can make this assumption, w.l.o.g. because X^S was only ever defined for this set. So we can basically "extend" it, such that it has the same distribution under $\{\tau \leq T_i\}$, which makes it independent of these sets.*

This methodology can easily be extended to products where only parts of the payoff are dependent on survival and even where other parts are dependent on default:

Lemma 4.20. *Let X be a payoff at $T_i \in \{T_1, ..., T_N\}$, with*

$$X = X^O + X^S\mathbf{1}_{\{\tau(\omega)>T_i\}} + X^D\mathbf{1}_{\{\tau(\omega)\leq T_i\}},$$

where the payoff X^O is unconditional, X^S is conditional pre-default and X^D is conditional past-default and all are $\mathcal{F}_{T_i}$-measurable.
Then the price at time $t = 0$ is

$$\Pi^X = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^O}{B(T_i)}\right] + \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)}Q(T_i)\right] + \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^D}{B(T_i)}(1 - Q(T_i))\right].$$

Proof. For the parts X^O and X^S the statement clear. For X^D we have:

$$\begin{aligned}\mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^D}{B(T_i)}J(T_i)\right] &= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^D}{B(T_i)}\right] - \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^D}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\right]\\ &= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^D}{B(T_i)}(1 - Q(T_i))\right]\end{aligned}$$

with the same arguments as in the lemma before. □

Lets re-evaluate the above equation with our expression for Q. One might notice that pricing defaultable products is similar to a change of numeraire with the defaultable numeraire:

$$\mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)}Q(T_i)\right] = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B^d(T_i)}\right]. \tag{18}$$

While this cannot hold up as a proof, it can be used as an intuitive explanation: we exchange the theoretical time value of future money with one that also accounts for default possibilities.
With this expression it is also easy to derive an alternative numerical price for the defaultable zero-coupon bonds, by setting $X^S = 1$:

$$P^d(0; T_i) = P^{d,*}(0; T_i) = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{1}{B^d(T_i)}\right]$$

While we normally have the analytic value of $P^d(0, T_i)$ as an input value to our model (or at least calibrated from other input values), the numerical price can act as an error measurement for the model we create. Furthermore, we can later use it as a control variate in pricing, as it is done for the non defaultable model (see [**?**]), which is beyond the scope of this thesis.
We can also get a numerical expression for the total survival probability, because setting $X^S = B(T_i)$ and pricing the claim yields:

$$\begin{aligned}\Pi^X &= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)}\mathbf{1}_{\{\tau(\omega)>T_i\}}\right] = \mathbb{E}^{\mathbb{Q}^B}\left[\mathbf{1}_{\{\tau(\omega)>T_i\}}\right] = \mathbb{Q}^B(\{\tau > T_i\})\\ &= \mathbb{E}^{\mathbb{Q}^B}\left[Q(T_i)\right]\end{aligned}$$

Remark 4.21. *It is obvious that if a party defaults, it can not pay (all of) its liabilities. However, a party in default will normally still collect its claims, as their creditors will try to get compensated for their loss. This leads to cash flows of the following form:*

$$X = \Psi^+ - \Psi^- \mathbf{1}_{\{\tau(\omega) > T_i\}},$$

where Ψ is a (random) payoff, which is to be collected (resp. payed) by the defaultable party if it is positive (resp. negative) and $x^+ := \max(x,0)$ and $x^- := \max(-x,0)$ for any $x \in \mathbb{R}$.

4.4.2 Products depending on two defaultable parties

Let us now extend this model. Assume a product is dependent on the survival of two different parties. For use in later sections we will call them debtor and creditor.

Such a claim is of the form

$$X = X^S \mathbf{1}_{\{\tau^d > T_i\}} \mathbf{1}_{\{\tau^c > T_i\}}, \tag{19}$$

where τ^d is the default time of the debtor and τ^c that of the creditor. Before we go on to find the price of such a claim, let us first state an assumption for simplicity's sake:

Assumption 4.2. *The default time of the debtor τ^d is independent of that the creditor τ^c.*

While this assumption is not realistic for all counter parties (the default of big banks has a significant impact on the default probability of its clients, see [**?**]), we can assume that it is true for most.

The price of such a claim can be found in the same way as above:

Lemma 4.22. *Let assumption 4.2 hold. The price for a T_i claim with a payoff function as in equation* (19) *can be computed by*

$$\Pi^\Psi = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)} Q^d(T_i) Q^c(T_i)\right],$$

where Q^d and Q^c are computed as given in definition 4.14 from two separate defaultable LIBOR market models with the same underlying non-defaultable model.

Note that we now construct a totally new set of defaultable values: we have the defaultable LIBOR market model for the debtor with $P^{d^d,*}$, L^{d^d} and their default time τ^d and we have a second defaultable model for the creditor and their values: $P^{d^c,*}$, L^{d^c} and τ^c. At the same time we still have the underlying non defaultable LIBOR market model.

Proof. For ease of notation let $A_j := \{\tau^d > T_j\} \cap \{\tau^c > T_j\}$ for all $j \in \{1, ..., N\}$. We follow exactly the proof of lemma 4.18, only we exchange $\mathbf{1}_{\{\tau > T_i\}}$ with $\mathbf{1}_{A_i}$:

$$\begin{aligned}
\Pi^X &= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)}\mathbf{1}_{A_i}\right] \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^S}{B(T_i)}\mathbf{1}_{A_i}\middle| A_1\right] q^d(T_1)q^c(T_1) && \text{(A)} \\
&= \dots \\
&= \mathbb{E}^{\mathbb{Q}^B}\left[\mathbb{E}^{\mathbb{Q}^B}_{\mathcal{G}_{T_i}}\left[Q^d(T_i)Q^c(T_i)\frac{X^S}{B(T_i)}\mathbf{1}_{A_i}\middle| A_i\right]\right] && \text{(B)}
\end{aligned}$$

where we get (A) from the independence of τ^d and τ^c:

$$\mathbb{Q}^B\left(\{\tau^d > T_1\} \cap \{\tau^c > T_1\}\right) = q^d(T_1)q^c(T_1)$$

and (B) in the same way as in the proof of lemma 4.18. $\square$

4.4.3 Funding considerations

In this subsection we will only discuss products, that can not be traded after the initial time. This means we have to separate the terms *pricing* and *valuation*. While the first term corresponds to determining the cost to again set up a product like this, the latter corresponds to determining the worth at the time, for a specific party. We will see, that the difference is the numeraire.

We have previously established that a party facing default risk can offer self-distributed zero-coupon bonds at a price of

$$P^{d,*}(0;T) = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{1}{B^d(T)}\right],$$

compensating the buyer for the associated default risk. Consequently, such a party can only borrow funds at a rate derived from the defaultable numeraire. This further implies that the valuation of any forward contract from their perspective is based on this rate, as any self-financing replication strategy for any financial product would involve borrowing at this rate for the initial investment. The valuation of a T-claim with payoff X is hence of the form:

$$\mathbb{E}^{\mathbb{Q}^B}\left[\frac{X}{B^d(T)}\right] \tag{20}$$

Remark 4.23. *It is important to note, that this formula holds only for valuation, not pricing. While it is priced higher (namely $\mathbb{E}^{\mathbb{Q}^B}\left[\frac{X}{B(T)}\right]$), the value for the defaultable party is as specified above.*

We extend our valuation to defaultable products. Assuming the counter-party d is defaultable and the T_i-claim X is structured as in lemma 4.20. According to equation (20), the value of X is determined from the viewpoint of party c, as expressed by

$$\mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^O + X^S Q^d(T_i) + X^D\left(1 - Q^d(T_i)\right)}{B^{d^c}(T_i)}\right].$$

Note that this valuation is analogue to pricing a product, where all cash flows are dependent on the survival of c. This analogy offers an intuitive perspective: in the event of c defaulting, they cease to benefit or incur liabilities from cash flows as they effectively cease to exist. Consequently, their valuation considers all cash flows to be dependent on their own survival.

Let us look at the case where the evaluating party (again c) is also the one, whose survival some cash flows depends on. We again consider the same structure of the T_i claim X as in lemma 4.20. Applying equation (20) we get:

$$\mathbb{E}^{\mathbb{Q}^B}\left[\frac{X^O + X^S}{B^{d^c}(T_i)}\right]$$

We see that an entire term is omitted: X^D, which is dependent on the default of c. We can also justify this with the analogy above: X^D is only relevant in the case in which c defaults, so c can neither benefit from X^D nor incur liabilities.

Next we will see how we can also account for some behavioural aspects in pricing.

4.4.4 Introducing Behavioral Aspects

Let us look at a specific product category: American loan options. This is the option to buy or sell a loan at a fixed nominal, even if the market conditions have changed. We assume an over the counter (OTC) product, meaning trading the product to other counter parties is not possible. The payoff of such a product if exercised at t would be (in this case we can buy the loan):

$$\Psi(t) = (\mathcal{N} - \Pi_t)^+ ,$$

where $\mathcal{N}$ is the nominal and Π_t denotes the current value at time t of the repayment obligation, typically structured as a coupon bond.

In option pricing it is always assumed that all market participants have a perfect overview of the market (captured by the pricing filtration $\mathbb{G}$). Furthermore it is assumed that debtors always act in an optimized manner (exercise the option if it has a positive value). While big companies may have implemented ways such that these are feasible assumptions, this is far from realistic for small market participants such as private people.

A study from 2021 from Germany asked people how to best spend €1,000 given two options: making a special repayment of an existing loan of theirs, which charges 5 % interest or invest in a fixed coupon bond with 3 % interest and maturity of 3 years [**?**]. The results showed that only 60 % of the participants would make the special repayment. 15 % stated that they did not know the answer, 13 % thought both options were equally good and 12 % would rather buy the coupon bond [**?**].

This shows that the assumption of optimal exercising is an idealization and hence for fair pricing models we need to adjust for this deficit.

Remark 4.24. *Note that in general option pricing an adjustment would generate an arbitrage possibility, because small market participants could buy cheap options and resell them to big companies at a higher price, without any risk.*

However, as established we are talking about products that are not transferable (the defaultable LIBOR models are calibrated to the individual client). This closes the arbitrage possibility.

How do we inject such a behavioral aspect in our valuation formula? We could construct a stochastic model which simulates the exercising of a market participant. However, we already omitted modeling the default time for the simple reason of computational cost and modeling behavior would require at least the same amount of computing power.

Hence we try a simpler approach: we apply a so called deterministic shadow barrier $\tilde{S}_t$ (for flexibility we leave it dependent on time), which leads to a different cash flow:

$$\Psi(t) = \left(\mathcal{N} - \Pi_t - \tilde{S}_t\right)^+ ,$$

Generally one can see $\tilde{S}_t$ as the profit it takes, until the debtor would be motivated enough to exercise the option. As an example consider a private person entering a cancellable loan (see section 6.2.2). They would not cancel the loan, the moment that they would gain a profit, because that profit would be too small to be a real incentive. Instead, they would wait until the profit is larger (or it has again dropped in which case they would not exercise).

5 Numerical Specification

The implementation of the described concepts are a crucial part of this thesis. We use Java, which is a purely object oriented programming language, for having a good code readability while still performing very well. We assume a basic knowledge in Java. As build system we use Maven.
Before we can jump into any code, though, we need to describe ways to discretize a stochastic process.

5.1 Numerical Schemes for Stochastic Processes

In this chapter we introduce ways to approach the simulation of stochastic processes described by SDEs. These numerical schemes can also be found in [**?**], which is also the main source of this section.
As basis for the schemes we have a known SDE for a n-dimensional stochastic process $X = (X_t)_{t\in[0,\tilde{T}]}$ that we want to simulate:

$$\begin{aligned} dX_t &= \mu(t, X_t)dt + \sigma(t, X_t) \cdot dU_t, \\ X_0 &= x, \end{aligned} \tag{21}$$

where U is a d-dimensional standard Brownian Motion, $\mu : [0, \tilde{T}] \times \mathbb{R}^n \to \mathbb{R}^n$ and $\sigma : [0, \tilde{T}] \times \mathbb{R}^n \to \mathbb{R}^{n\times d}$ are functions and $x \in \mathbb{R}^n$.
Let us start with the well-known Euler or Euler-Maruyama scheme:

Definition 5.1. *Let $X = (X_t)_{t\in[0,\tilde{T}]}$ be as above in (21).*
Furthermore let $m \in \mathbb{N}$, $(t_i)_{i\in\{0,...,m\}}$, where $0 = t_0 < t_1 < ... < t_m = \tilde{T}$.
Then we call the discrete process $\hat{X} = (\hat{X}_{t_i})_{i\in\{0,...,m\}}$, given by

$$\begin{aligned} \hat{X}_{t_0} &= x, \\ \hat{X}_{t_{i+1}} &= \hat{X}_{t_i} + \mu(t_i, \hat{X}_{t_i})\Delta t_i + \sigma(t_i, \hat{X}_{t_i}) \cdot \Delta U_{t_i}, \end{aligned}$$

where $i \in \{0, ..., m-1\}$, an Euler-Maruyama scheme *of X.*

The Euler-Maruyama scheme takes the concept of approaching deterministic integrals and applies it on stochastic integrals. While this works quite well for constant and deterministic factor loadings $\sigma(t_i, x) = \sigma(t_i)$ [**?**], it has a weakness for stochastic factor loadings.

An improvement would be the Milstein scheme as described in [**?**], but due to its correction term, which gets fairly complicated for higher dimensional Brownian motions, it is not well suited for our purpose. Instead we consider the functional Euler Scheme:

Definition 5.2. *Let $X = (X_t)_{t\in[0,\tilde{T}]}$ be as above in (21).*
Let $Y = (Y_t)_{t\in[0,\tilde{T}]}$ be defined as $Y_t = f(t, X_t) \quad \forall t \in [0,\tilde{T}]$ for a 2-times differentiable function $f : [0,\tilde{T}] \times \mathbb{R}^n \to \mathbb{R}^k$ and $k \in \mathbb{N}$.
Let $m \in \mathbb{N}$, $(t_i)_{i\in\{0,...,m\}}$, where $0 = t_0 < t_1 < ... < t_m = \tilde{T}$. Let $\hat{X} = (\hat{X}_{t_i})_{i\in\{0,...,m\}}$ be an Euler-Maruyama scheme of X.
Then we call the discrete process $\hat{Y} = (\hat{Y}_{t_i})_{i\in\{0,...,m\}}$, given by

$$\hat{Y}_{t_i} = f(t, \hat{X}_{t_i}),$$

where $i \in \{0, ..., m\}$, a functional Euler scheme *of Y.*

This scheme performs well in situations, where the factor loadings of X are less dependent on X, than the factor loadings of Y are on Y.

It is also very useful, if f preserves a property of Y which otherwise might be lost through the numerical error of the other schemes (such as staying in a certain domain).

Considering this, the functional Euler scheme is great for the approximation of log-normal processes, as it cancels a part of the numerical error due to the dependency of the factor loadings on the approximated process and at the same time it preserves its positivity.

In figure 2 we can see a direct comparison of the schemes for two different paths of a geometric Brownian motion, i.e. $X_t = X_0 \exp((\mu - 0.5\sigma^2)t + \sigma W_t)$, where $X_0 = 0.1$, $\mu = -0.1$ and $\sigma = 0.7$. As time discretization we choose $\Delta t = 0.2$.

In this case the functional Euler scheme generates exact paths, which do not have any numeric error from discretizing the SDE. We see that the Euler scheme portrays a numerical error that is quite extensive at certain times. One path

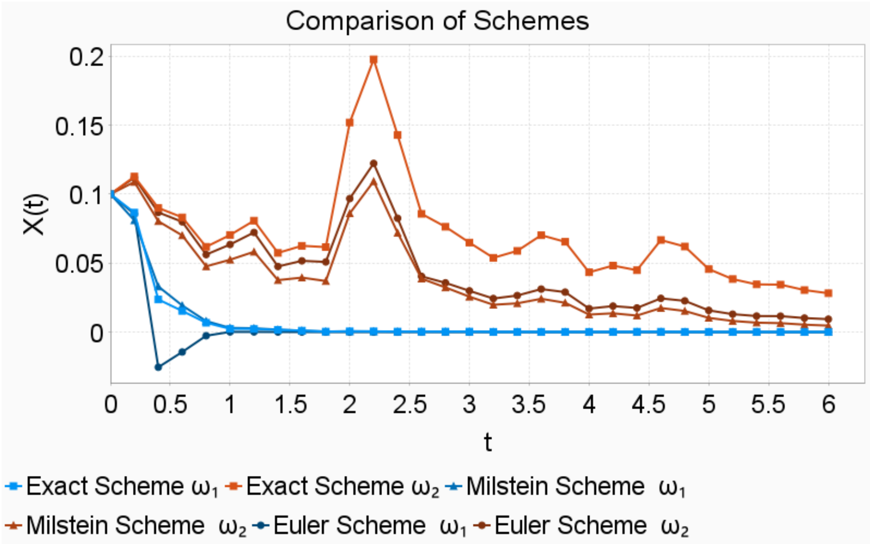

Figure 2: Comparison of Numerical Schemes for two paths of a Geometric Brownian Motion

of the Euler scheme even reaches a value below zero, while the dynamics of the geometric Brownian motion are such that this should be impossible. It is worth to note, however, that the time discretization is chosen quite rough for demonstration purposes.

To also prove that the schemes work analytically, we define some convergence rates as in [?]:

Definition 5.3. *Let $\hat{X}^\delta$ be a time-discrete scheme for the approximation of a process X, where δ is the maximum time difference between approximations, i.e. $\delta = \max\{t_i - t_{i-1} | i \in \{1, ..., m\}\}$.*

Then we say that $\hat{X}^\delta$ converges strongly to X if:

$$\lim_{\delta \downarrow 0} \mathbb{E}\left[\left|X_{\tilde{T}} - \hat{X}^\delta_{\tilde{T}}\right|\right] = 0.$$

We say that $\hat{X}^\delta$ converges strongly with order $\gamma > 0$ to X if $\exists C > 0$ and a δ_{max} such that

$$\mathbb{E}\left[\left|X_{\tilde{T}} - \hat{X}^\delta_{\tilde{T}}\right|\right] \leq C\delta^\gamma$$

for all $\delta \in]0, \delta_{max}[$.

It is well known that the Euler scheme fulfills the following properties:

Lemma 5.4. *Let X be the solution of equation* (21)*. Let* $\|\cdot\|$ *denote the euclidean norm. Let furthermore*

$$|\mu(t,x)-\mu(t,y)|+\|\sigma(t,x)-\sigma(t,y)\| \leq K_1\|x-y\|, \tag{22}$$

$$|\mu(t,x)|+\|\sigma(t,x)\| \leq K_2(1+\|x\|), \tag{23}$$

$$|\mu(t,x)-\mu(s,x)|+\|\sigma(t,x)-\sigma(s,x)\| \leq K_3(1+\|x\|)|t-s|^{\frac{1}{2}} \tag{24}$$

for all $s,t \in \left[0,\tilde{T}\right]$ *and* $x,y \in \mathbb{R}^n$ *for some constants* $K_1, K_2, K_3 \in \mathbb{R}$*. Then the Euler scheme given in definition 5.1 converges strongly to X with order* $\frac{1}{2}$*.*

Proof. See [**?**], pp. 342. □

Note that we have to apply the conditions component-wise. We see that they are the same as for the unique solution theorem [**?**]. So if these were not satisfied, we could not guarantee the SDEs of the LIBOR model to have a solution.
Let us state the following lemmata:

Lemma 5.5. *The following statements hold for any $\mathbb{R}$-valued Lipschitz functions f and g with the same closed domain A:*

- *the function $(f \cdot g)$ (i.e. the product) is also Lipschitz on A.*
- *the function $(f+g)$ (i.e. the sum) is also Lipschitz on A*

Proof. Let us denote the absolute maximum of g and f on A (which is attained because of continuity and the closed domain) $\bar{g}$ and $\bar{f}$ respectively. We get:

$$\begin{aligned}
|f(x)g(x)-f(y)g(y)| &= |f(x)g(x)-f(y)g(x)+f(y)g(x)-f(y)g(y)| \\
&= |f(x)g(x)-f(y)g(x)+f(y)g(x)-f(y)g(y)| \\
&\leq |f(x)-f(y)||\bar{g}|+|g(x)-g(y)||\bar{f}| \\
&\leq |x-y|\left(K^f|\bar{g}|+K^g|\bar{f}|\right),
\end{aligned}$$

where K^f and K^g are the Lipschitz constants of f and g respectively. The second statement is trivial. □

Lemma 5.6. *Let* $L_t := (L_i(t))_{i\in\{0,...,N-1\}}$ *and* $L_t^d := \left(L_i^d(t)\right)_{i\in\{0,...,N-1\}}$. *Let* $i \in \{0,...,N-1\}$, $C_1^i \in \left]-\frac{1}{\Delta T_i}, 0\right]$ *and* $C_2 \in [0,\infty[$. *Let* $A \subset \mathbb{R}^{1+2N}$ *be defined as the box* $A := \left[0,\tilde{T}\right] \times \prod_{j=0}^{N-1}\left[C_1^j, C_2\right] \times \prod_{j=0}^{N-1}\left[C_1^j, C_2\right]$. *Let* $\lambda_t^{ik,d}$ *be such that there exist Lipschitz-continuous* $\mathbb{R}$*-valued functions* $\lambda^{ik,d}(t,x_1,x_2)$ *on* A *such that*

$$\lambda^{ik,d}\left(t, L_t, L_t^d\right) = \lambda_t^{ik,d}$$

if $L_i(t), L_i^d(t) \in [C_1^i, C_2]$ *for all* $k \in \{1,...,m^d\}$.
Then equations (22) *to* (24) *are satisfied for the SDEs of* L^d *for* $s,t \in \left[0,\tilde{T}\right]$ *and* $x,y \in \prod_{j=0}^{N-1}\left[C_1^j, C_2\right] \times \prod_{j=0}^{N-1}\left[C_1^j, C_2\right]$.

Proof. Let

$$\mu^{i,d}(t,x,y) := \sum_{k=1}^{m} \lambda^{ik,d}(t,x,y) \sum_{j=m(t)+1}^{i} \frac{\lambda^{jk,d}(t,x,y)\,\Delta T_j}{1+y_j\Delta T_j} \tag{25}$$

on A. It is easy to see that, if $L_i(t), L_i^d(t) \in [C_1^i, C_2]$ for all $i \in \{0,...,N-1\}$, then $\mu^{i,d}\left(t, L_t, L_t^d\right) = \mu_t^{i,d}$.
Because y_j is bounded by its domain, we have that $\frac{\Delta T_j}{1+y_j\Delta T_j}$ is Lipschitz-continuous on A for all j. With lemma 5.5, it is trivial to see that $\mu^{i,d}(t,x,y)$ is a sum of Lipschitz-continuous functions and, hence, Lipschitz, which means that equation (22) is satisfied for all i.
Because both $\mu^{i,d}$ and $\lambda^{ik,d}$ are Lipschitz and defined on the closed set A, they have an absolute maximum, which we denote $\bar{\mu}^{i,d}$, $\bar{\lambda}^{ik,d}$. Setting $K_2 = \bar{\mu}^{i,d} + \sum_{k=1}^{m^d} \bar{\lambda}^{ik,d}$ yields equation (23).
As $\mu^{i,d}$ and $\lambda^{ik,d}$ are Lipschitz:

$$\begin{aligned}
|\mu^{i,d}(t,x,y) - \mu^{i,d}(s,x,y)| + \|\lambda^{ik,d}(t,x,y) - \lambda^{ik,d}(s,x,y)\| \\
&\leq C|t-s| \\
&= C\sqrt{|t-s|}\sqrt{|t-s|} \\
&\leq C\sqrt{\tilde{T}}\,(1+\|(x,y)\|)\sqrt{|t-s|}
\end{aligned}$$

which yields equation (24). $\square$

Note that as a starting point we need to choose Lipschitz-continuous λ and λ^d. We have not proved equations (22) to (24) for the whole domain of λ^d and μ^d.

If C_1^i and C_2 are chosen such that the probability of $L_i(t), L_i^d(t) \in [C_1^i, C_2]$ is effectively zero for all $t \in \left[0, \tilde{T}\right]$, this is sufficient for our numerical analysis, however. We use a trick: we argue that we can use the Euler scheme for SDEs with adjusted drift $\hat{\mu}^{i,d}$ and factor loadings $\hat{\lambda}^{ik,d}$:

$$\hat{\mu}_t^{i,d} := \mu^{i,d}\left(t, L_t\Big|_{[C_1,C_2]}, L_t^d\Big|_{[C_1,C_2]}\right)$$
$$\hat{\lambda}_t^{ik,d} := \lambda^{ik,d}\left(t, L_t\Big|_{[C_1,C_2]}, L_t^d\Big|_{[C_1,C_2]}\right),$$

where we used the functional form of $\mu^{i,d}$, $\lambda^{i,d}$ respectively, and we used $x\Big|_{[C_1,C_2]} = \left(\min\left(\max\left(x_i, C_1^i\right), C_2\right)\right)_{i\in\{0,\ldots,N-1\}}$. This yields $\mu^{i,d}$ and $\lambda^{ik,d}$ for all $i \in \{0, ..., N-1\}$ and $k \in \left\{1, ...m^d\right\}$ until L_t or L_t^d exit the box A, in which case we just fix $\hat{\mu}^{i,d}$ and $\hat{\lambda}^{ik,d}$ to be the edge case. It is easy to see, that the adjusted functions satisfy equations (22) to (24) for all $s, t \in \left[0, \tilde{T}\right]$ and $x, y \in \mathbb{R}^{2N}$.

The trick is to simulate the paths for the SDEs with the normal $\mu^{i,d}$ and $\lambda^{ik,d}$ anyway. We can only simulate a finite number of paths $Z \in \mathbb{N}$. Hence, we check if there is any path, for which the Euler scheme of $L_i(t)$ and $L_i^d(t)$ hit either $\frac{-1}{\Delta T_i}$ or ∞ for any $i \in \{0, ...N-1\}$. If there is none, we argument, that we had set C_1^i and C_2 to the lowest point of all paths (highest resp.) of $L_i^d(t)$ and $L_i(t)$. Otherwise we should rethink our choice of parameters.

Lemma 5.7. *Let $C_1^i \in \left]-\frac{1}{\Delta T_i}\right]$ and $C_2 \in [0, \infty[$ be arbitrary for all $i \in \{0, ..., N-1\}$. Let the factor loadings of the non-defaultable model $\lambda^{ik}(t, L(t))$ be Lipschitz-continuous on $\left[0, \tilde{T}\right] \times \prod_{j=0}^{N-1}\left[C_1^j, C_2\right]$ for all $i \in \{0, ..., N-1\}$ and $k \in \{0, ..., m\}$. Then the factor loadings $\lambda^{ik,d}$ described in lemma 4.10 are Lipschitz on the box A (defined in lemma 5.6), if the free parameters $f_t^{ik} = f^{ik}(t, x, y)$ are Lipschitz on the box A.*

Proof. This is a trivial consequence from lemmas 5.5 and 5.6. □

This concludes the analysis on how to simulate the LIBORs for our model.

5.2 The finMath Library

We can now start to look at the actual implementation.

As a starting point for the code base we use the finMath Library of Prof. Christian Fries.

The finMath library is written in Java and provides interfaces and classes for the use in stochastics and financial mathematics.

In this section we take a look at the ones that are most frequently used by our code. For a more thorough description one can use the website of the finMath library [?], which is also our main source for this section.

RandomVariable

As one can tell from the name, `RandomVariable` is an interface for working with stochastic values, hence it is the basis for Monte Carlo methods. Operator overloading is not possible in Java so the workaround is to have methods that represent these operations, which is what `RandomVariable` declares. Taking the expectation (the mean) and the variance is also supported. A nice feature is a `RandomVariableFactory`, which handles all creations of a `RandomVariable`.

ProcessModel

The `ProcessModel` interface is the finMath equivalent of an SDE. It specifies that any implementation has methods for getting the initial state, the drift and the factor loadings.

```
public interface ProcessModel {
   int getNumberOfComponents();
   RandomVariable[] getInitialState(MonteCarloProcess process);
   RandomVariable[] getDrift(MonteCarloProcess process, int timeIndex,
       ↪ RandomVariable[] realizationAtTimeIndex, RandomVariable[]
       ↪ realizationPredictor);
   RandomVariable[] getFactorLoading(MonteCarloProcess process, int timeIndex,
       ↪ int componentIndex, RandomVariable[] realizationAtTimeIndex);
}
```

Listing 1: The `ProcessModel` interface (some methods are ommitted)

We can use a `ProcessModel` as a plug in to numerical schemes simulating an SDE such as `EulerSchemeFromProcessModel`.

MonteCarloProcess

The `MonteCarloProcess` is an interface for the numerical scheme simulating an SDE that was before specified as `ProcessModel`. The most important implementation of this interface is `EulerSchemeFromProcessModel`, which handles as well the computation for normal Euler schemes as for functional Euler schemes.

LIBORMarketModel

The `LIBORMarketModel` is a representation for a LIBOR model. It is an interface that extends the `ProcessModel` and allows for querying LIBOR and other forward rates for different periods at different evaluation times, given a `MonteCarloProcess`.

LIBORCovarianceModel

We use the `LIBORCovarianceModel` interface for the flexible implementation of covariance structures of LIBOR models. Its only responsibility is to provide covariances and their factor loadings. One can use it as plug in to a `LIBORMarketModel`, as well as one to other `LIBORCovarianceModels`, which from there derive other factor loadings.

Abstraction

We see that all the objects described above are interfaces, not directly classes. This means that none of them have a specific implementation and yet we can describe what they are and what they do. Neither the user nor any class, that uses such objects need to know the exact implementation to use them. The main advantage is that it allows to reuse code in many situations. Consider as example the simulation of SDEs: if we could not specify a `ProcessModel` as an input, there would need to be a `MonteCarloProcess` class for every SDE that we create, even though the calculation of its paths is always the same. This concept is commonly known as abstraction in computer science.
We follow along with that concept also in the implementation for our model.

5.3 Implementation of the Defaultable LIBOR Model

In this section we describe the classes that were implemented for this thesis. The Java files and project setup can be found in the Github repository [?]. The repository is setup as Maven project, with all dependencies – such as the finMath library – linked in the "pom.xml" file. Further description of the packages and classes can be found in the README and the respective Java-Doc.

DefaultableLIBORMarketModel

To use the full flexibility of the finMath Library, the implementation of our models is structured in the same way as in the finMath library: we have an Interface for the defaultable LIBOR market model which extends the `ProcessModel`:

```
public interface DefaultableLIBORMarketModel extends LIBORMarketModel,
    ↪ ProcessModel {
```

Listing 2: Declaration of `DefaultableLIBORMarketModel`

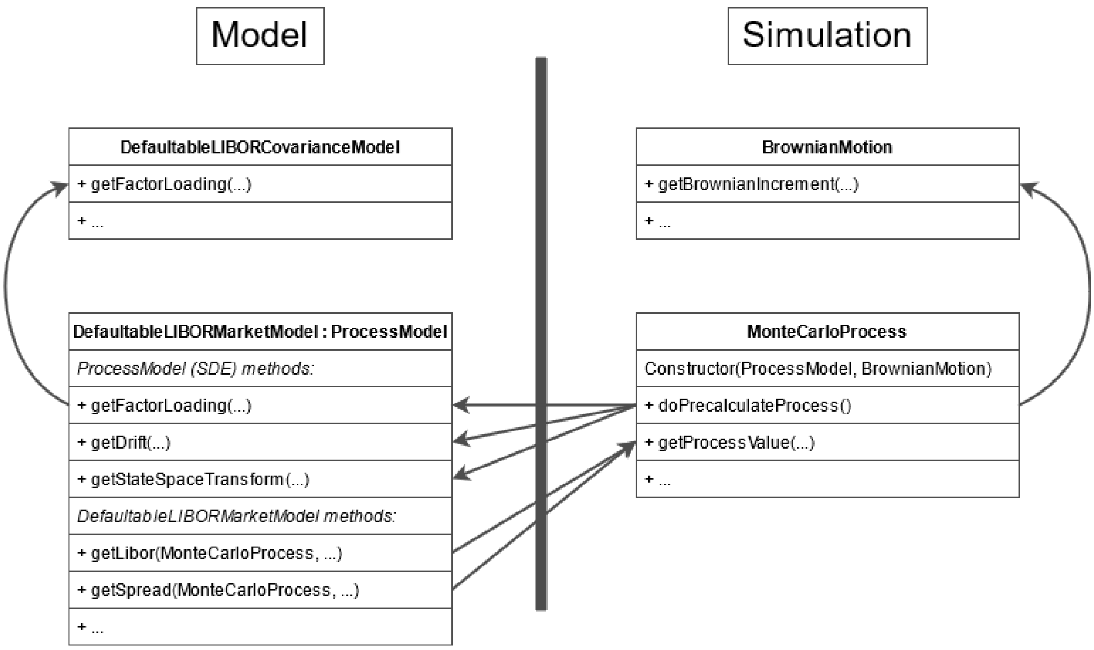

Figure 3: Using `DefaultableLIBORMarketModel` as Plug-In for `MonteCarloProcess`

Let us take a look at the cooperation between our model and the numerical scheme simulating the Process in figure 3.

The `DefaultableLIBORMarketModel` can be used as a plug in for a class implementing the `MonteCarloProcess` interface, which calls SDE-methods of the `ProcessModel` to pre-calculate the process. Calls to `getFactorLoading(...)` should be delegated to the `DefaultableLIBORCovarianceModel`, as we can then reuse the code of `DefaultableLIBORMarketModel`, even though creating different kinds of covariance (factor-loading) models.

Note here that we also extend the `LIBORMarketModel`. This however does not necessarily mean that the SDE specified by the `ProcessModel` simulates the LIBOR rates. In fact the process values approximating the SDE are accessed by our model through input parameters and can be processed further. We therefore have a clean separation of model specifications and simulation.

DefaultableLIBORFromSpreadDynamic

We take advantage of this fact in the class `DefaultableLIBORFromSpreadDynamic`, which implements the interface. We here allow the user to specify which values should be simulated by the SDE: either the spreads or the defaultable LIBOR rates directly. When getting the defaultable LIBOR rates we query the given `MonteCarloProcess` for its values and process them as needed:

```java
@Override
public RandomVariable getDefaultableLIBOR(MonteCarloProcess process, int timeIndex, int liborIndex) throws CalculationException {
    if(timeIndex == 0 || getLiborPeriod(liborIndex) == 0d)
        return getRandomVariableForConstant(getForwardRateCurve().getForward(getAnalyticModel(), getLiborPeriod(liborIndex)));

    if (simulationModel == SimulationModel.SPREADS)
        return getNonDefaultableLIBOR(process, timeIndex, liborIndex).add(process.getProcessValue( timeIndex, getSpreadComponentIndex(liborIndex)));
    else
        return process.getProcessValue(timeIndex, getDefaultableComponentIndex(liborIndex));
```

Listing 3: Getting the defaultable LIBOR rates in `DefaultableLIBORFromSpreadDynamic`

The ProcessModel in this class describes as well the SDEs of the non-defaultable model as well as the ones for the defaultable one (either spreads or directly the LIBORs). This is why in line 6 of listing 3 we add the spread to the process value at $liborIndex$, which is the non defaultable LIBOR, to return the defaultable rate.

All methods corresponding to ProcessModel check if the given component index is smaller than the number of LIBORs, that are simulated. If so, they delegate the call to the non-defaultable model, which must be given in the constructor. As one might notice, we need to delegate the call with a reduced MonteCarloProcess, which is handled through getNonDefaultableProcess(...).

```java
@Override
public RandomVariable[] getDriftOfNonDefaultableModel(MonteCarloProcess process, int timeIndex, RandomVariable[] realizationAtTimeIndex, RandomVariable[] realizationPredictor) {
    RandomVariable[] sRealization = null;
    if(realizationAtTimeIndex != null) {
        sRealization = Arrays.copyOf(realizationAtTimeIndex, getNonDefaultableLIBORModel().getNumberOfComponents());
    }
    RandomVariable[] sRealizationPredictor = null;
    if(realizationPredictor != null) {
        sRealizationPredictor = Arrays.copyOf(realizationPredictor, getNonDefaultableLIBORModel().getNumberOfComponents());
    }
    return getNonDefaultableLIBORModel().getDrift(getNonDefaultableProcess(process), timeIndex, sRealization, sRealizationPredictor);
```

Listing 4: Delegating of the non-defaultable drift calculation in DefaultableLIBORFromSpreadDynamic

MultiLIBORVectorModel

In a similar way, the MultiLIBORVectorModel acts as a wrapper class to simulate multiple defaultable LIBOR models at the same time, which we can then use to value products depending on two or more defaultable parties. A wrapper class is necessary, as the joint simulation of the multiple models – non-defaultable and defaultable – is essential for a meaningful valuation.

The initial inclination to separately simulate the instances of `DefaultableLIBORMarketModel` – which each contains its own non-defaultable LIBOR simulation – within distinct `MonteCarloProcess`es, is problematic. This approach would lead to either generating numerous disparate simulations of the non-defaultable model with different stochastic drivers, or employing the same stochastic driver for every defaultable model, resulting in a strong interdependence among the models.

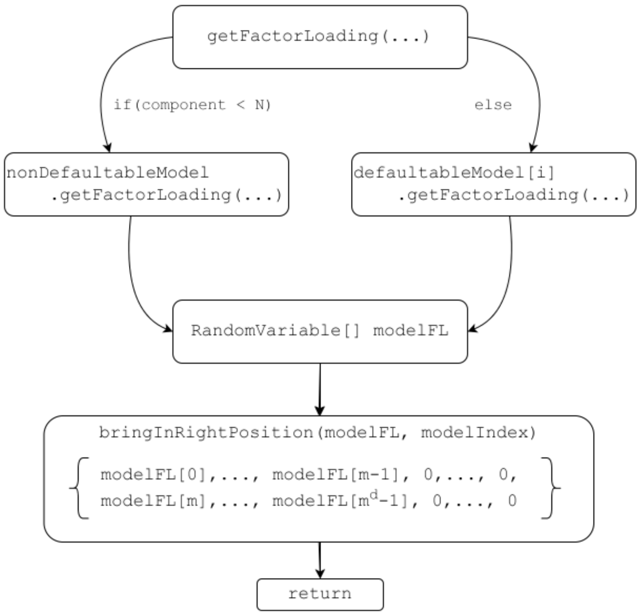

Figure 4: Workflow of `MultiLIBORVectorModel.getFactorLoading(...)`

In figure 4 we see how the workflow of `getFactorLoading(...)` in `MultiLIBORVectorModel` achieves returning a joint factor loading vector for multiple models. First the component index is checked, which model it corresponds to, and the respective factor loadings are fetched. Then they are reordered to represent a model with $m+\sum_{i=0}^{l-1}\left(m^{d^i}-m\right)$ factors. Here l is the number of defaultable models and m^{d^i} is the number of factors corresponding to the i-th defaultable model. Note that the first Brownian Motion factors of each defaultable model match the Brownian Motion factors of the non-defaultable model. All factors that correspond exclusively to other defaultable models are set to constant zero.

6 Numerical Analysis

In this section we will analyze the numerical results that our model yields. It is divided into three parts:
In the first subsection we will inspect the performance, which we measure by time, numerical error and the realization of the positivity condition in the numerical implementation.
The second subsection is dedicated to a qualitative analysis of some specific product valuations, which were found to be interesting. Each product will be first described theoretically and then the numerical implementation is analyzed.
In the third part we show how the input parameters, namely the initial spread curve and the covariance parameters impact the variance and correlation of the model.
Note that we have not given a simpler parametrization for the covariance structure than the free parameter matrix. For all our valuations we used equally distributed randomly generated free parameter matrices, centered around zero. To still have a very simple grip on the input, we define Δf as the size of the interval where the free parameters lie in. Hence it always holds that:

$$\max\left(f^{ik} - f^{jl} | i, j \in \{0, ..., N-1\}, l, k \in \{m+1, ..., m^d\}\right) \leq \Delta f$$

6.1 Performance of the model

We start with analyzing the performance of the model. While for theoretical analysis irrelevant, timing is an important measure for programming, as it can reveal inefficiencies and bottle necks of the model.We start with this topic.

6.1.1 Timing

Timing is performed by extending the class "Time", which only has two relevant (static) methods: *tic()* which starts the timer, and *toc()* which yields the result (in Nanoseconds). Note that the calculation of the paths is performed using lazy initialization, which means, the calculation is only performed, if we actually *use* the results. Figure 5 shows that the calculation time of the defaultable model

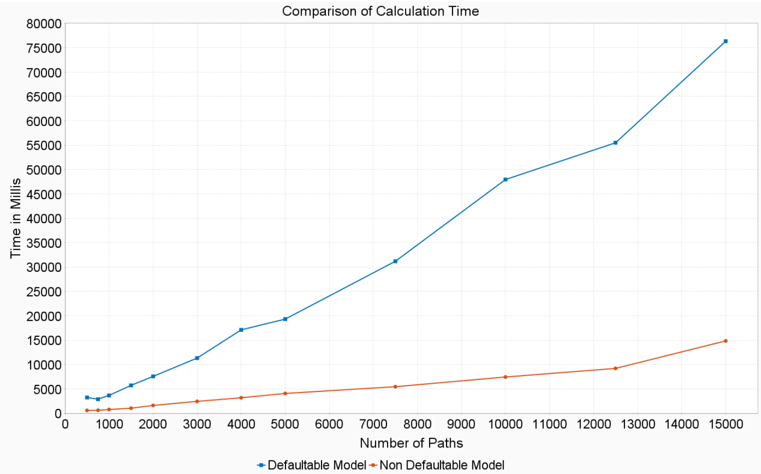

Figure 5: Comparison of the calculation time for a 20-tenor LIBOR model

is not only double of the calculation time of the non-defaultable model, as one could expect. It rather lies between 5-7 times as much computational time. Note that while the computation time itself depends heavily on the system used (we used a Windows System with 11 cores and 16 GB RAM), the relative position of the two curves shown, should be roughly equal. The reason for this result might lie in the programming design: because the simulating process expects every SDE value to be in an array, a heavily amount of time is spent copying data from one array to another. Another might lie in the duplication of computing the non-defaultable values, such as drift and factor loadings (which are needed for the calculation of the drift of the spread/defaultable LIBORs).

6.1.2 Numerical error

We can estimate the numerical error, by looking at the defaultable bond prices at time 0. These are given in analytical form, by deriving them from the initial value of the defaultable LIBOR rate:

$$P^d(0;T) = \frac{1}{1 + L^d_{m(T)}(0)\left(T - T_{m(T)}\right)} \prod_{j=0}^{m(T)} \frac{1}{1 + L^d_j(0)\Delta T_j}$$

but we can also price them numerically with Monte Carlo methods:

$$\hat{P}^d(0;T) = \hat{\mathbb{E}}^{\mathbb{Q}^B}\left[\frac{1}{B(T)}Q(T)\right]$$

Figure 6 shows how this error evolves over time. One can see that at initial time

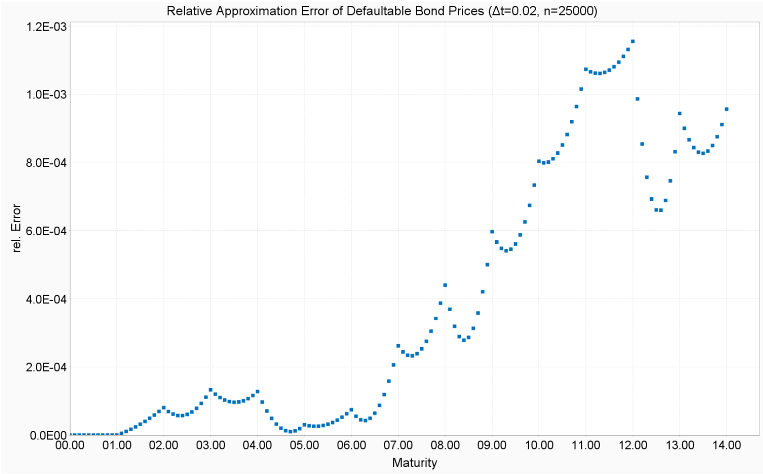

Figure 6: Relative error of defaultable bond prices calculated under the spot measure by a defaultable LIBOR model through a functional Euler-Maruyama scheme with time step $\Delta t = 0.02$ and 25000 sample paths and exponential state space.

the error is quite low, but exponentially rising after some LIBOR periods. The reason for this structure are two main factors:

- The exponential structure of the spread model also implies that all numerical errors are exponential over time.
- The use of the spot measure implies that the numeraire gathers more stochasticity over time, making Monte Carlo methods more prone to numerical errors.

Note that the level of the error is quite low with 0.12 % being the largest relative error.

The zero-coupon bond error as shown in figure 6 can be extinguished by applying the analytic initial prices as a control variate. Note however that this will not totally extinguish the numerical error for the valuation of other products.

Another possibility to measure the numerical error, is to directly use the LIBOR rates. From the proof of theorem 4.5 we know that $L_i^d(t)P^d(t;T_{i+1})$ discounted is a martingale. Hence we know, that its expectation is also its starting value:

$$L_i^d(0)P^d(0;T_{i+1}) = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{L_i^d(T_i)}{B(T_{i+1})}\mathbf{1}_{\{\tau>T_{i+1}\}}\right]$$

Using Monte Carlo on the same product yields the approximation:

$$L_i^d(0)P^d(0;T_{i+1}) \approx \mathbb{E}^{\mathbb{Q}^B}\left[\frac{L_i^d(T_i)}{B(T_{i+1})}Q(T_{i+1})\right]$$

The relative error is shown in figure 7. There we see that the errors do not follow

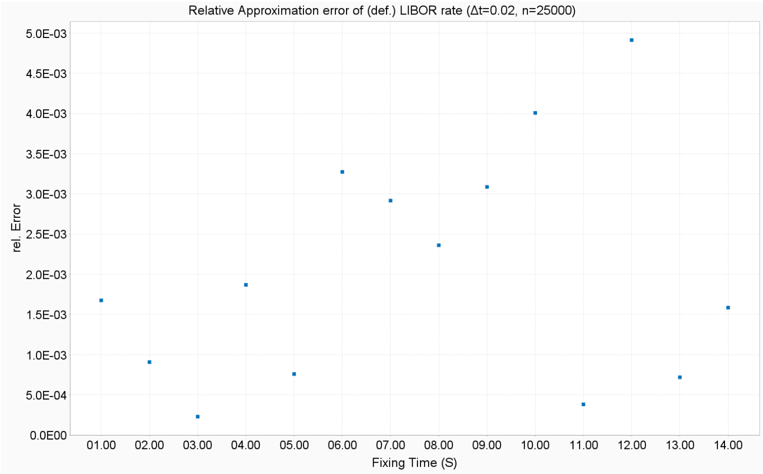

Figure 7: Relative error for the approximation of LIBOR rates calculated under the spot measure by a defaultable LIBOR model through a functional Euler-Maruyama scheme with time step $\Delta t = 0.02$ and 25000 sample paths and exponential state space.

as strong of a pattern as the bond price errors. It is true, however, that also here a larger maturity time yields a larger error, which is normal for Monte Carlo methods. Also here we can observe that the relative error stays a low level even at a large fixing time.

6.1.3 Positivity of the model

While we have proven that the model in theory does not allow for negative spreads, we test this numerically. This can also be seen as a test for the convergence of the Euler-Maruyama scheme. Note that the functional Euler-Maruyama scheme, uses an exponential and hence eliminates all risks of returning negative paths. Therefore one can look at the minimal path for each spread simulated as odone in table 1. There we indeed see that the normal Euler-Maruyama scheme

Table 1: Minimum value of all simulated paths for the spread with Δf being the range of the (constant, randomly generated) free parameters centered around 0

Time Step Δt	Euler Scheme $\Delta f = 1.0$	Euler Functional $\Delta f = 1.0$	Euler Scheme $\Delta f = 0.5$	Euler Functional $\Delta f = 0.5$
0.5	-1.695E-01	5.666E-06	-4.227E-03	2.882E-04
0.25	-7.206E-02	3.476E-06	8.061E-05	2.359E-04
0.1	-7.583E-03	3.125E-06	1.741E-04	2.161E-04
0.05	-1.168E-04	3.461E-06	1.735E-04	2.259E-04
0.025	2.180E-06	3.225E-06	2.189E-04	2.478E-04
0.01	2.814E-06	2.965E-06	2.084E-04	2.152E-04

generates negative values, as well for the free parameters with a large range as for those with a smaller range. While for a smaller Δf (i.e. the range of the free parameters) the scheme seems to be stable from $\Delta t = 0.1$ downwards, the larger values of Δf seem to pose a problem for the Euler scheme even when choosing rather small time steps. This comes as no surprise as we see in section 6.3.2 that Δf has a direct link to the variance of the model.

6.2 Numerical Results on Loan Valuation

In this section we look at some concrete valuation examples. Since our model has its main strength in capturing default probability directly within the discount curve it has a native application in loan pricing.

Therefore let us first take a look at what a loan is in general. "A loan is a sum of money that one or more individuals or companies borrow from banks or other financial institutions [...]. In doing so, the borrower incurs a debt, which he has to pay back with interest and within a given period of time."[**?**].

Hence a loan is nothing else than a fixed coupon bond, where the nominal $\mathcal{N}$

represents the debt and the coupons c_i represent the interest. We see that in the defaultable case the price is analogous to the non defaultable case.

Lemma 6.1. *The price of a coupon paying bond which has defaultable cashflows is:*

$$\mathcal{N} = \sum_{i=1}^{N} c_i P^{d,*}(0;T_i) + \mathcal{N} P^{d,*}(0;T_N)$$

Proof. Let $c \in \mathbb{R}$ be constant. Then for $i \in \{1, ..., N\}$:

$$\begin{aligned}\mathbb{E}^{\mathbb{Q}^B}\left[\frac{c\,\mathbf{1}_{\{\tau(\omega)>T_i\}}}{B(T_i)}\right] &= c\,\mathbb{E}^{\mathbb{Q}^B}\left[\frac{P^d(T_i;T_i)}{B(T_i)}\right] \\ &= c\,P^d(0;T_i) = c\,P^{d,*}(0;T_i).\end{aligned}$$

Replacing c with c_i, $\mathcal{N}$ repectively, and taking the sum yields the statement. □

Note that for a simple loan it does not matter if the creditor is defaultable, because they have no debt w.r.t. the loan after $t = 0$ and in case of their default the pending claim would still be collected.

6.2.1 Loan Futures

We look at a future on defaultable loans, or rather a future on a defaultable coupon paying bond. Remember that in contrast to an option a future brings the obligation to enter into an agreement. Hence from the debtors point of view we have a payoff function

$$\Psi(T_s) = \left(\mathcal{N} - \sum_{i=s+1}^{N} c_i P^{d,*}(T_s;T_i)\right)\mathbf{1}_{\{\tau>T_s\}} \tag{26}$$

where T_s is the start time of the bond and for ease of notation the terminal coupon c_N includes the terminal payment of the nominal $\mathcal{N}$.

This payoff function gives us a direct approach to price the product:

Lemma 6.2. *A T_s claim with a payoff function as in equation* (26) *has the price*

$$\begin{aligned}\Pi^{\Psi} &= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{\mathcal{N}}{B^d(T_s)} - \sum_{i=s+1}^{N}\frac{c_i}{B^d(T_i)}\right] \\ &= \mathcal{N} P^{d,*}(0;T_s) - \sum_{i=s+1}^{N} c_i P^{d,*}(0;T_i)\end{aligned}$$

at $t = 0$.

Proof. Follows directly form section 4.4 □

For a loan future we can also account for the default probability of the creditor. For this we adjust the payoff

$$\Psi(T_s) = \left(\mathcal{N} - \sum_{i=s+1}^{N} c_i P^{d^d,*}(T_s;T_i)\right) \mathbf{1}_{\{\tau^d > T_s\}} \mathbf{1}_{\{\tau^c > T_s\}} \tag{27}$$

and hence also the price:

$$\Pi^\Psi = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{\mathcal{N} - \sum_{i=s+1}^{N} c_i P^{d^d,*}(T_s;T_i)}{B(T_s)} Q^d(T_s) Q^c(T_s)\right] \tag{28}$$

$$= \mathbb{E}^{\mathbb{Q}^B}\left[\frac{\mathcal{N} Q^c(T_s)}{B^{d^d}(T_s)} - \sum_{i=s+1}^{N} c_i \frac{Q^c(T_s)}{B^{d^d}(T_i)}\right] \tag{29}$$

Note that we applied remark 4.21, which suggests, a creditor in default still collects their claims, while not pay their liabilities (the default risk of the creditor only matters until T_s).

Equation (28) also gives us the valuation from the perspective of the debtor as discussed in section 4.4.3. For the valuation from the creditors perspective we need a slightly different approach, as we also need to discount by the creditors funding curve during the loan itself:

$$\mathbb{E}^{\mathbb{Q}^B}\left[\frac{\mathcal{N} Q^c(T_s)}{B^{d^d}(T_s)}\right] - \sum_{i=s+1}^{N} c_i \mathbb{E}^{\mathbb{Q}^B}\left[\frac{Q^c(T_i)}{B^{d^d}(T_i)}\right]$$

Let us use our model to value the derived products from the different perspectives. In figure 8 we see the prices of the product if both parties involved are defaultable in perspective to only the debtor being defaultable.

The figure visualizes that the forward coupon bond has two main stochastic cost drivers:

- The default probability *past* maturity time, which has a positive relation to the price: if the default probability rises, $P^{d,*}(T_s;T_i)$ decreases, which drives the price up, visualized by the case of a defaultable debtor.

- The default probability *before* maturity time, which has a negative relation to the price intensity: if the default probability rises, $Q(T_s)$ decreases,

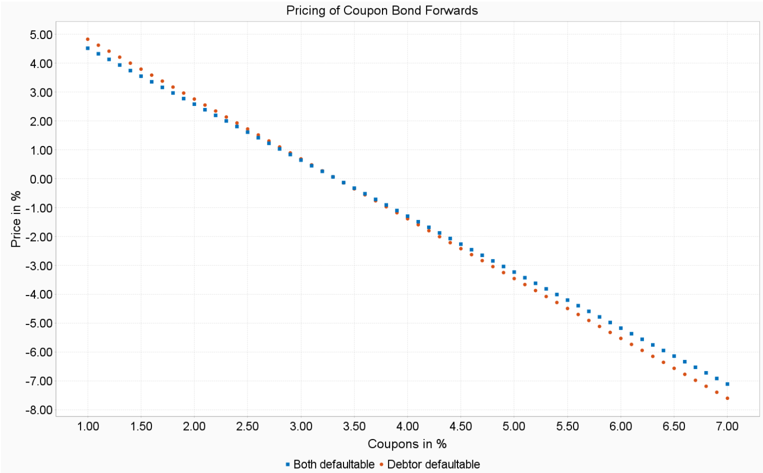

Figure 8: Price of a coupon bond forward if both involved parties are defaultable and only the debtor being defaultable plotted against the coupon rates.

> which lowers the absolute value of the price (as $0 < Q(T_s) < 1$). This is visualized by the case of the creditor being defaultable, as their default probability is only accounted until T_s.

While for presentation purposes the creditor used for the valuation has a higher default probability than the debtor (larger Δf and larger initial spread), the effect – though strongly reduced – has the same tendency: figure 9 shows the same valuations only with the roles swapped. We see, that also here accounting for the default probability of the creditor flattens the price curve, although way less. A bigger effect can be seen in figure 10 where we show the effect of the

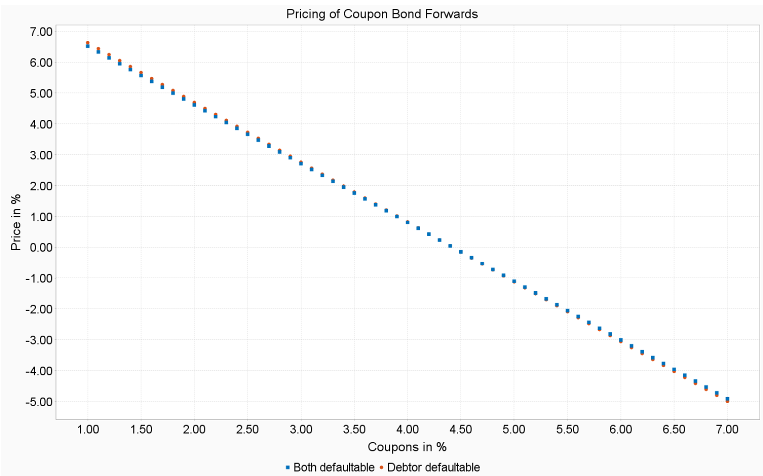

Figure 9

funding considerations as discussed in section 4.4.3. Note that the two curves are almost parallel.

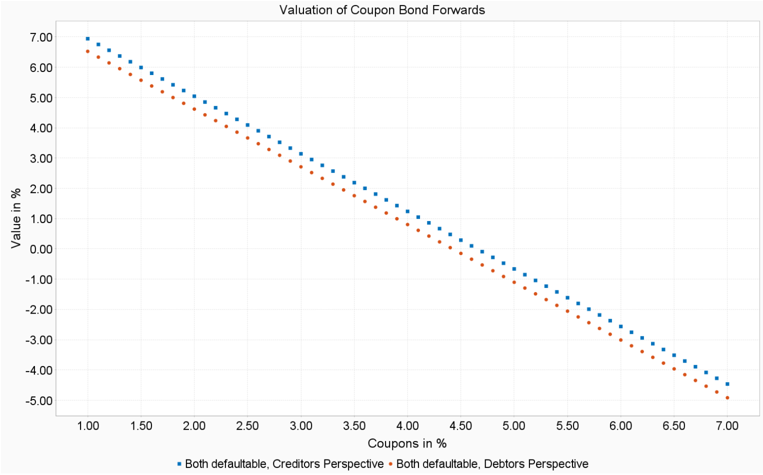

Figure 10: Valuation of a coupon bond forward valued from the creditor's perspective and the debtor's plotted against the coupon rates

One can explain this phenomena through the default probability of the creditor. While in the example before only the probability of default until maturity time affected the value, now one has to consider also the one past maturity time for valuation.

6.2.2 Cancellable Loans

In the last subsection we considered pricing loans and loan forwards. Essentially cash flow that is set in stone (with the exception of default). Now let us consider more intriguing products which bring an optionality with them: cancellable loans. We start with the case, where one can cancel the loan at a single time point T_k for $k \in \{1, ...N-1\}$.

Consider the cash flow of such a product from the debtors point of view: We get the nominal $\mathcal{N}$ at the start of the loan. Then we pay coupons each tenor until T_k. If we cancel the loan, we have to pay the redemption $\tilde{R}$, otherwise we keep paying coupons until the end of the loan.

To derive a price for this product, we split the cash flow in two parts: a loan and a put option on a loan, as visualized in figure 11. The option is then treated as

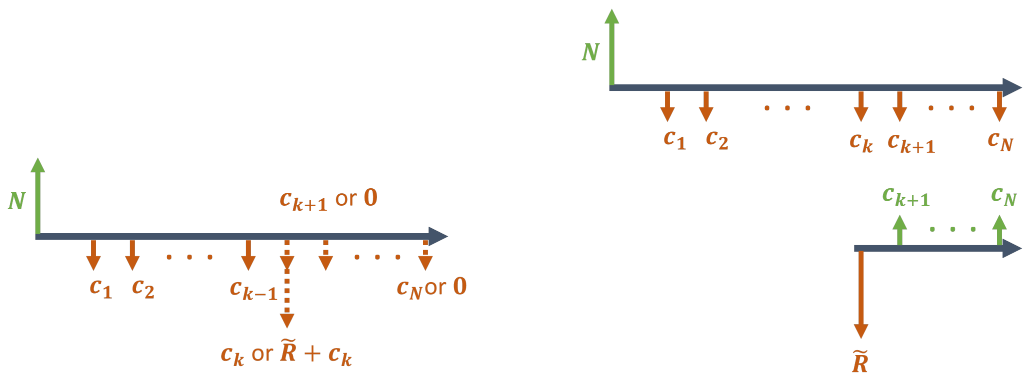

Figure 11: Cashflow of a cancellable loan and its splitting into two products

single cash flow in terms of the price, which leads us to a payoff of:

$$\begin{array}{cl} \mathcal{N} & \text{at } T_0, \\ -c_i \mathbf{1}_{\{\tau > T_i\}} & \text{at } T_i \text{ for } i \in \{1, ..., N\}, \\ \left(\sum_{j=k+1}^{N} c_j P^{d,*}(T_k; T_j) - \tilde{R} \right)^{+} \mathbf{1}_{\{\tau > T_k\}} & \text{at } T_k. \end{array}$$

Note that if $\left(\sum_{j=k+1}^{N} c_j P^{d,*}(T_k; T_j) - \tilde{R} \right) > 0$ at T_k, the debtor could get a new loan on the nominal $\tilde{R}$ for better conditions than the initial one.

From there it is easy to compute the price:

$$\mathcal{N} - \sum_{i=1}^{N} c_i P^{d,*}(0; T_i) + \mathbb{E}^{\mathbb{Q}^B} \left[\left(\sum_{j=k+1}^{N} \frac{c_i}{B^d(T_i)} - \frac{\tilde{R}}{B^d(T_k)} \right)^{+} \right] \tag{30}$$

Note that neither the option nor the loan contain a cash flow that the creditor has to pay besides the initial payment, which makes the price independent of the creditors default probability. This is due to the fact, that even if the option is exercised it is still the debtor paying the creditor. For valuation from the creditors perspective, however, we still have to account for their funding costs:

$$\mathcal{N} - \mathbb{E}^{\mathbb{Q}^B} \left[\sum_{i=1}^{N} \frac{c_i}{B^{d^c}(T_i)} Q^d(T_i) + \mathbb{E}^{\mathbb{Q}^B} \left[\left(\sum_{j=k+1}^{N} \frac{c_i Q^d(T_i)}{B^{d^c}(T_i)} - \frac{\tilde{R} Q^d(T_k)}{B^{d^c}(T_k)} \right)^{+} \middle| \mathcal{F}_{T_k} \right] \right] \tag{31}$$

Let us yet make this a little more intriguing: we generalize this product for the possibility to cancel at any time past T_k for a fixed $k \in \{1, ..., N-1\}$. Note that in this case the redemption $\tilde{R}_t$ would depend on the cancel time, but still be deterministic. Then the adjusted cash flow is

$$\begin{array}{ll} \mathcal{N} & \text{at } T_0, \\ -c_i \mathbf{1}_{\{\tau > T_i\}} & \text{at } T_i \text{ for } i \in \{1, ..., N\}, \\ \left(\sum_{j=m(\nu)+1}^{N} c_j P^{d,*}(\nu; T_j) - \tilde{R}_\nu\right) \mathbf{1}_{\{\tau > \nu\}} & \text{at } \nu, \end{array}$$

where $\nu = \nu(\omega)$ is the stopping time:

$$\nu(\omega) = \min\left(t \in [T_k, T_N[\;\middle|\; \left(\sum_{j=m(t)+1}^{N} c_j P^{d,*}(t; T_j) - \tilde{R}_t\right) > 0\right) \wedge T_N$$

Hence this takes the form of an American option, where one can choose to exercise the option in the time interval $[T_k, T_N]$. As always the price can be obtained by taking the discounted expectation:

$$\begin{aligned} &\mathcal{N} - \sum_{i=1}^{N} c_i P^{d,*}(0; T_i) \\ &\quad + \mathbb{E}^{\mathbb{Q}^B}\left[\mathbb{E}^{\mathbb{Q}^B}\left[\left(\sum_{j=m(\nu)+1}^{N} \frac{c_i}{B^d(T_i)} - \frac{\tilde{R}_\nu}{B^d(\nu)}\right)^+ \middle| \mathcal{F}_\nu\right]\right] \end{aligned}$$

As ν is only dependent on $\mathbb{F}$-adapted values, it is also a value in terms of model primitives, which makes it usable to us. A valuation as seen by the creditor can be performed by

$$\begin{aligned} &\mathcal{N} - \sum_{i=1}^{N} \mathbb{E}^{\mathbb{Q}^B}\left[\frac{c_i Q^d(T_i)}{B^{d^c}(T_i)}\right] \\ &\quad + \mathbb{E}^{\mathbb{Q}^B}\left[\mathbb{E}^{\mathbb{Q}^B}\left[\left(\sum_{j=m(\nu)+1}^{N} \frac{c_i Q^d(T_i)}{B^{d^c}(T_i)} - \frac{\tilde{R}_\nu Q^d(\nu)}{B^{d^c}(\nu)}\right)^+ \middle| \mathcal{F}_\nu\right]\right] \end{aligned}$$

Note that the creditor would have a different stopping criterion ν if it were their option to exercise. Hence, for valuation we can see two different paths: either the creditor values with their stopping criterion or they value with the debtor's criterion. As visualized in figure 12 we can see, that considering funding costs does significantly impact the valuation. In fact the value as seen from the creditor's perspective is substantially lower as that seen from the debtor. The reason for this is explained in section 4.4.3: the creditor has to hedge against the debtors default risk, but also consider that he can not borrow money at the risk free rate. While the debtor can also not borrow money at the risk free rate, he does not have to hedge against his own default.

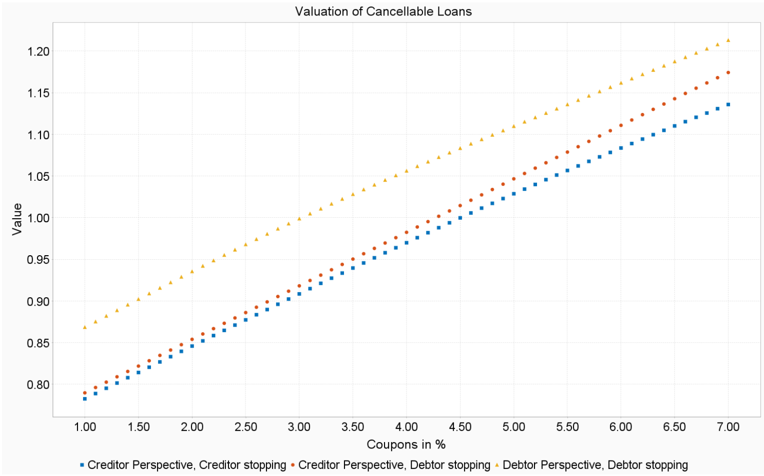

Figure 12: Cancellable loan by coupon rate valued from the creditor's perspective with the creditor's stopping criterion, as well as with the debtor's stopping criterion and from the debtor's perspective

6.3 Impact of the model inputs

Let us analyze how the input parameters – i.e. initial spread and the matrix of free parameters – affect the model. We start with the initial spread.

6.3.1 Impact of the initial spread

Figure 13 illustrates how the variance of the spread of two models (S^d and S^c) is impacted by their initial spread curve. For illustration purposes the initial spread is fixed for each LIBOR period in the example. We see that the effects are exponential. This is to be expected as the model is lognormal. We also see that the effect is dependent on Δf: the larger the range of the free parameter the greater the impact of the initial spread. As the example simulates the different models jointly, one could argue that part of the processes' variance is due to a strong dependence between the two, which can be discarded, as the covariance shows no impact.

Let us take a look at the correlation of two jointly simulated defaultable models. Also here we are interested in reviewing how the model input affects the values. Remember that in section 4.4 we assumed two models for the joint valuation of products to be independent. Figure 14 shows that at least numerically this

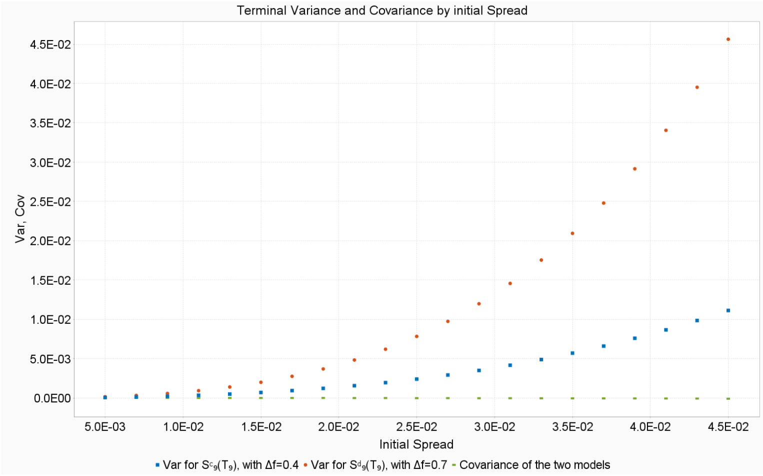

Figure 13: Impact of the initial spread on the variance and covariance of the terminal spread.

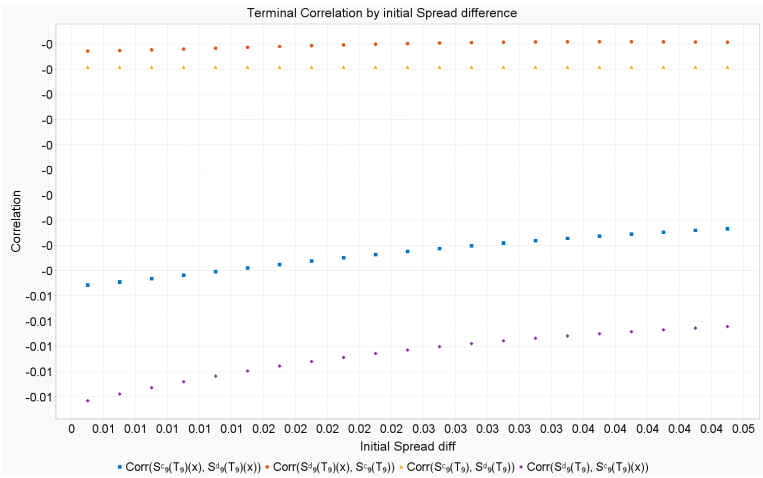

Figure 14: Terminal correlation of two spread of defaultable LIBOR models to the same non defaultable model by the initial spread curve.

seems to be possible to achieve. Even when using only two additional Brownian Motion factors, there is almost no correlation to be found between the terminal spreads of the two models. We see that the initial spread indeed has an impact on the correlation structure, although at a very low level.

6.3.2 Impact of the free parameters

The model as described in section 4 has another input, which is the free parameter matrix for the covariance structure of the defaultable LIBORs. Since our research does not address the parametrization of this structure (note the potential for time-dependent f_t^{ik}), we employ a matrix of time-constant free parameters, generated randomly and evenly distributed around 0. We then use Δf as our handle to experiment the impact of the input. For each model we fix the seed of the generator to visualize the actual impact.

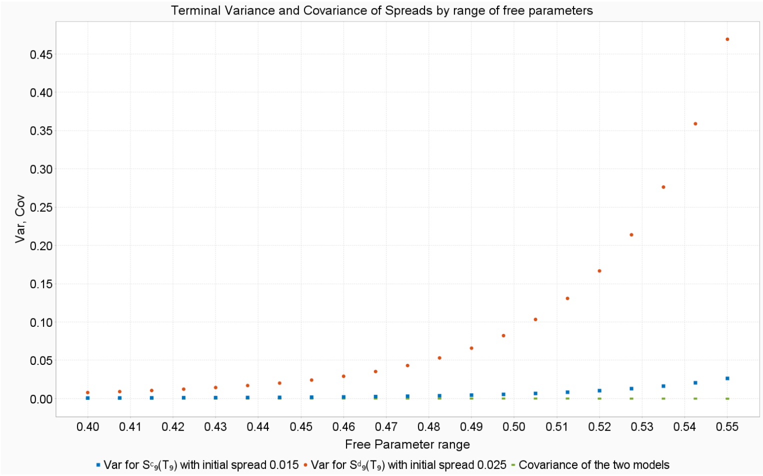

Figure 15: Variance of the terminal spread of two models and their Covariance plotted against Δf

Let us again analyze the variance. As visualized in figure 15 we see as expected that the range of free parameters indeed has a major impact. Even though only changing Δf by 0.15 the variance has a jump from almost zero to 0.45. It is clear, that the relation is again exponential. As the model of the spread is log-normal and the factor loadings of the last $m^d - m$ factors are the free parameters this relation comes as no surprise. We look at the correlation, which again plays at an insignificant level as seen in figure 16. However here one can see that, though on a low level, the correlation indeed is impacted by the range of free parameters. A lower Δf implies a higher dependence (keep in mind that a negative correlation implies the same level of dependence as the same value in the positive direction). This makes sense, as the lower the free parameters, the more randomness is explained by the first factors, which are common for both models. A free parameter matrix that has constant zero values would constitute the same SDEs for the spread with the same Brownian Motion, with only the initial values being different.

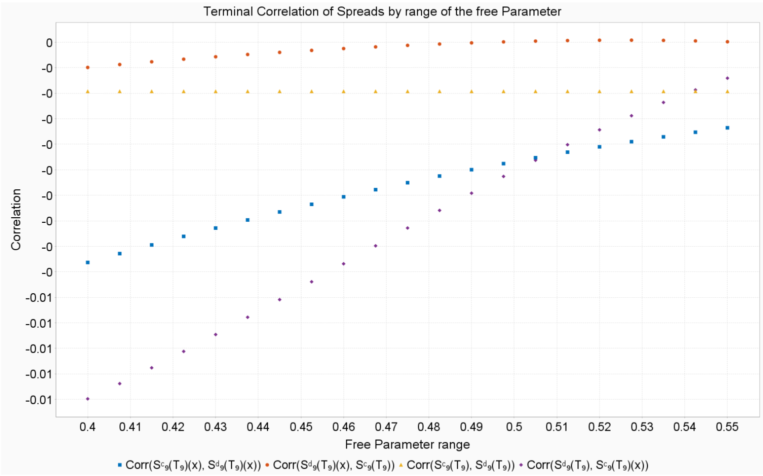

Figure 16: Correlation of two models spread plotted against Δf

7 Conclusion

In conclusion, it can be said that default forward rate models make a significant contribution to the modeling and analysis of defaultable bond and loan behavior. The implementation of the model in Java provides a solid basis for the simulation and valuation of financial products such as loans. However, despite the performance achieved, there is still room for improvement, especially in terms of time performance.

Investigating a potential implementation of the model in C++ could provide promising results, as this language often offers higher speed for computationally intensive tasks. Furthermore, evaluating the use of an implementation of `RandomvariableFactory` that runs calculations on the GPU is a promising way to improve computational speed. Such implementations already exist, however were not used in our analysis.

The defaultable LIBOR model has several notable strengths that underscore its relevance and applicability in the financial world. One of its outstanding features is its ability to combine default risk and interest rates into a single model, allowing for a consistent analysis of products that are naturally bound to both values, such as bonds. This integration allows financial players to develop a more comprehensive understanding of market conditions and risk factors.

In addition, there is a need for further research, particularly in the area of generating an appropriate free parameter matrix for the given covariance model. A solid structure based on fewer parameters is crucial for an effective calibration of the model and enables a precise adjustment to market conditions. This aspect underlines the importance of continuous research efforts to continuously improve the model and ensure its adaptability to changing market conditions.

Another promising approach for future research efforts is to link two defaultable LIBOR models through a correlation to capture and simulate potential structural dependencies between different financial instruments or markets. By integrating correlation structures throughout multiple default forward rate models, one can gain a more comprehensive understanding of the impact of the default probability of large market participants on the valuation of products even for small market participants.

In summary, the further development and optimization of defaultable LIBOR models is an important area of research that can have a significant impact on the financial industry, both theoretically and practically.

8 List of Symbols

Annotation	Meaning
SDE	stochastic differential equation
w.r.t.	with respect to
$\mathbb{Q}^B$	martingale measure w.r.t. the numeraire $B(t)$
$m(t)$	For a tenor $T_0 < ... < T_N$, $m(t) := \max\{i \in \{0, ..., N-1\} \mid T_i \leq t\}$
$(\partial_x f)(x)$	Same as $\frac{\partial}{\partial x} f(x)$.
$(\partial_{xy} f)(x, y)$	Same as $\frac{\partial^2}{\partial x \partial y} f(x, y)$.
$\mathbb{E}_{\mathcal{F}}^{\mathbb{Q}^B}[\,\cdot\,]$	Same as $\mathbb{E}^{\mathbb{Q}^B}[\,\cdot \mid \mathcal{F}\,]$. for a σ-algebra $\mathcal{F}$

Ehrenwörtliche Erklärung

Ich erkläre hiermit ehrenwörtlich, dass ich die vorliegende Arbeit selbständig angefertigt habe; die aus fremden Quellen direkt oder indirekt übernommenen Gedanken sind als solche kenntlich gemacht.

Die Arbeit wurde bisher keiner anderen Prüfungsbehörde vorgelegt und auch noch nicht veröffentlicht.

Ort, Datum	Unterschrift