Imperial College London

Imperial College London

Department of Mathematics

Methods of Pricing Cross-Currency Bermudan Swaptions

Author: David Malone (CID: 02285306)

A thesis submitted for the degree of

MSc in Mathematics and Finance, 2022-2023

Declaration

The work contained in this thesis is my own work unless otherwise stated.

Acknowledgements

The work achieved in this thesis would not have been possible without the invaluable tutelage and guidance from both Dr. Thomas Cass and Miro Vukoje. Both of whom selflessly went above and beyond in offering me their time, wisdom, and insight. For this I am incredibly grateful. Their advice and our discussions were an endless source of inspiration for me, and I hope they can see that reflected in this work. I am also very grateful to the wider RPA team at EBRD for welcoming me with open arms and offering their support and interest in my work.

To my mother, father, brother and friends, thank you for the unending and unconditional love and support you have given me over the past year, and many years prior. Without question, I am where I am today because of all of you. So with more passion and love than I could ever express, wholly and truly, thank you.

Abstract

This thesis aims to study a multitude of methods that are employed by financial institutions to price Cross-Currency Bermudan Swaptions. Cross-Currency Swaps are contracts in which two parties exchange notional amounts in different currencies and make payments based on fixed or floating rates relative to these notionals. A Bermudan Swaption on such a contract gives the option holder the opportunity to either enter or exit these contracts at a predetermined set of dates. Two pricing methodologies will be considered in this paper. First, we seek to extend an existing backward induction method, described by Hagan in [1] for pricing the single-currency equivalent of these contracts, to the cross-currency case. This has not been researched previously to the best of our knowledge. Secondly, we seek to implement a more common method using Monte Carlo with a Longstaff-Schwartz [2] approach. We then provide an in-depth exploration, analysis and comparison of these methods in both the single-currency and cross-currency contexts. In the single-currency case we find that Hagan's method is slightly slower but more accurate than Monte Carlo, both offer a high degree of accuracy though. We ultimately failed to implement Hagan's method in the cross-currency case due to computational complexity but discussed a potential alternative solution for future research. We implemented the Monte Carlo method in the cross-currency case successfully. We note that the Monte Carlo method would have the same level of convergence as in the single-currency case, but the contract we have chosen to price is deep-out-of-the-money which leads to slower convergence than we would otherwise expect. This also serves as a real-world example of the shortcomings of a Longstaff-Schwartz approach. Despite this, the method provides a high level of accuracy. With the advent of new overnight rates and the transition away from Inter-Bank-Offer-Rates, many investors and institutions are seeking to understand their potential future exposures to interest and FX rates. Products such as these can be used for significant speculation and funding strategies in circumstances of significant change. These types of products are complex by nature. The cross-currency component introduces a multi-dimensional aspect while the option component requires the calculation of optimal stopping times. For this reason, a common thread through past literature is the struggle to find a balance between accuracy and tractability, the method produced by Hagan has the potential to circumvent certain inaccuracies that are unavoidable in other methods. This method has never been studied in the context of cross-currency products before to the best of our knowledge. Similarly, little work has been done to contrast this method to Monte Carlo methods in the single currency case. Monte Carlo methods have certainly been utilised for these types of products in the cross-currency case, but have rarely been assessed on the model we chose to use.

Contents

List of Figures

List of Tables

Introduction

A wide array of interest-rate products and derivatives are traded frequently, so it is of utmost importance to a financial institution to be able to price these products accurately and efficiently. Failure to do so could lead to adverse arbitrage opportunities and significant potential losses. Interest rate products are often used by banks and investors for either speculative purposes or hedging against the risk of adverse rate movements. In the context of risk management, the ability to price these products is necessary for determining potential future exposures. This has been of great importance in recent years. Due to increased amounts of refinancing and monetary policies, in order to try and quell inflation in the wake of the Covid-19 pandemic, we have seen significant movements in interest rates, making prices more unpredictable. Alongside this many financial institutions have begun to take great interest in these products as we approach the end of the once dogmatic use of Inter-Bank-Offer-Rates(IBOR). This has undoubtedly led to a great deal of uncertainty around the future of interest rates, meaning investors are looking for additional ways to capitalize on and protect themselves from the potential risks posed in such a transition.

Cross-Currency Bermudan Swaptions are highly customizable interest rate options. They can typically be thought of as two parties exchanging debt obligations in different currencies. Each party will exchange the notional payments and make interest-rate payments in the other party's currency. Options on these types of swaps are typically to enter or exit the swap at a given exercise date, the Bermudan aspect means there is a finite set of time points at which the owner can do this. These products are typically not used for any kind of hedging, any necessary positions could likely be hedged by more simple or fundamental products than this.

In order to facilitate understanding we will provide a basic example, from [3], of when investors may be interested in entering a cross-currency Swap. Consider two parties A and E, where A is an American investor who can obtain funding in USD at cheap rates, and E is a European investor who equally can avail of cheap funding rates in EUR. Both investors may only have access to less favourable overseas borrowing rates but are interested in borrowing for overseas projects. A cross-currency Swap would likely be beneficial to both parties in such a case. Suppose A requires 1 million EUR and E requires 1.08 million USD (equal to 1 million EUR) for their respective projects. Then the cross-currency Swap would involve A borrowing 1.08 million USD and E borrowing 1 million EUR. They then exchange the initial notional amounts and future payment obligations. Such that E receives the 1.08 million USD and pays USD interest, and A receives the 1 million EUR and pays EUR interest. Finally, at the maturity of the contract, A will return the 1 million EUR borrowed and E will return the 1.08 million USD. This example was adapted from [3]. An option on such a contract simply gives an investor the opportunity to enter or exit the contract at the most favourable times, if exchange rates or funding rates move in their favour.

In Chapter 1 we provide an introduction to the most common basic interest rate products. In particular, we will discuss how Bonds, Forward Rate Agreements (FRA), and various forms of Interest Rate Swaps are priced. These products are the building blocks required to work with the Cross-Currency Bermudan Swaption. There are many aspects and features of these products that we will describe in more detail at this stage as well.

In Chapter 2 we discuss the Linear-Gauss-Markov Model (LGM), a variation to the standard Hull-White model. We will first consider the single currency case and the necessary theory behind it. This will be the foundation of our approach to pricing our products of interest. Here we will also discuss the relevant change of numeraire and martingale pricing theory that will simplify our calculations. Following this we will extend the model to the cross-currency case. The problem of pricing these complex products becomes increasingly more difficult as the dimensionality of these products increases and upon introducing interaction between interest and foreign exchange (FX) rates.

Chapter 3 discusses the methodologies involved in pricing these Swaptions. Namely, we will observe how Bermudan Swaptions can be priced using the "Rollback" method in [1]. We start by explaining how it functions in the case of a single-currency Swaption. Following this we consider how we might formulate the problem in the cross-currency case and the shortcomings with this. We offer a potential solution at this point as well, namely the use of Cubatures on Wiener space, but we did not have time to explore this unfortunately.

In Chapter 4 we present the American Monte Carlo method that follows the Longstaff-Schwartz methodology [2] of pricing. We briefly consider the theory behind this, and contextualize it to the case of both the single-currency and cross-currency Swaptions.

Our results are presented and assessed in Chapter 5. We explore the choice of parameters for our Rollback method, and compare to the Monte Carlo method in the single-currency case in order to obtain a benchmark price. We then use this to contrast the accuracy, convergence, and computation times of both methods. We find that Rollback can be slower but more accurate than Monte Carlo in the single currency case. Rollback produces a single output for given inputs, but Monte Carlo will always have some amount of sampling error and requires a sufficient number of simulations to converge. We also explore certain regression choices for the Monte Carlo method, here we find that standard families of orthogonal polynomials work well with three degrees of freedom, particularly Hermite, power series, and Laguerre polynomials. In general, both methods produce accurate price estimates. We then repeat this for the cross-currency case where we notice an interesting edge case that highlights an issue with the Longstaff-Schwartz methodology as well. In particular, deep-out-of-the-money options pose issues with regressing on data that has excess zeros. We demonstrate how the error introduced by this dominates our Monte Carlo estimate and leads to a slower rate of convergence. We find that the same orthogonal polynomial regressions all perform equally poorly in such cases. We also briefly assess a Double Hurdle regression method [4], designed for such instances but find that even this cannot overcome the issue. Despite this, our estimates for the cross-currency Bermudan price are quite accurate, though they require significantly more simulations and longer computation times than the single-currency case.

Literature Review

The Foreign Exchange Market is easily one of the largest and most frequently traded markets in the world, averaging a turnover of 7.5 trillion USD a day in 2022. The market has recently been experiencing a period of high volatility due to an unparalleled amount of inter-dealer trading and a significant increase in the number of short-term maturity derivatives [5]. Alongside this the upcoming transition away from LIBOR rates means banks have to take much more caution in their management of assets and liabilities as current funding costs may no longer be optimal under new benchmark interest rates [6, Page 45]. Accordingly, many institutions are considering their potential future exposures to interest rates and FX rates, such is the motivation behind this thesis as well.

The problem of pricing cross currency products has been tackled in many different ways. Three-factor models are quite commonly used, as the domestic and foreign interest rates along with the FX rate must be modeled. Some researchers have opted to model short rates, the Hull-White model is often used to model individual short rates, these then incorporate a correlation structure through correlated Brownian Motions. This is chosen by [7] for its mean reversion characteristics, however, the model is also considered in [8] but is criticized for its inability to conform to the standard shape of a yield curve. Vasicek and Black-Karasinski short rate models are each criticised in [8] as well. Vasicek can lead to negative short rates which is not always desirable, and Black-Karasinski has no tractable Zero-Coupon Bond formula. A number of researchers have instead chosen to consider forward rate models, often using the Heath-Jarrow-Morton(HJM) framework or the Brace-Gatarek-Musiela (BGM) framework, also commonly referred to as the Libor Market Model. In particular [9] uses a HJM framework, and [8] discusses the Cheyette model which is an extension to the HJM framework. Meanwhile [10], [8], and [11] all use various forms of the BGM model for forward rates. The FX rates are of course also modeled, [7], [9], and [10] each use the same local volatility model given by,

$$\frac{ds(t)}{s(t)} = (r_d(t) - r_f(t))dt + \gamma(t, s(t))dW_s(t)$$

where $s(t)$ denotes the FX rate, r_d and r_f are the associated processes for the domestic and foreign short rates respectively, W_s is the Brownian motion correlated to the Brownian motions in the short rate and $\gamma(t, s(t))$ is the local volatility function. The form and notation are taken from [7, Page 2372]. Many of the other papers cited here lack detail about the modelling of FX rates. There are only a small number of papers that have explicitly considered Bermudan Swaptions here, namely [8], [9], [11], the others consider similar FX, interest rate related products with some form of callable feature, but under a different model than what we use here.

Once the model has been decided an array of methods are available to actually price the option. Monte Carlo simulation is one of the most approachable methods [9] suggests using an Euler discretization for simulation. Backwards induction methods such as Longstaff-Schwartz [2] are often used to determine the optimal exercise times in these types of problems. In [9], [11] [10] similar methods are used by comparing the expected value of the Swap at a future time to the current intrinsic value of the Swap. A PDE-based approach employing a Crank-Nicholson discretization is used in [7], Monte Carlo methods are criticized in this paper for having slow convergence and computationally difficult estimation of hedging parameters. However it is observed in [8] that finite difference schemes are limited by a computational complexity that grows exponentially with respect to the number of factors, making them inefficient for pricing Bermudan Cross-Currency Swaptions.

The book written by Lichter, Stamm and Gallagher [12] is one of the more relevant pieces of literature to this thesis. They specifically deal with cross-currency Swaptions and multi-factor

models. In particular, they propose the use of the Linear-Gauss-Markov (LGM) model for its analytical tractability as an interest rate model. It uses a unique numeraire in order to achieve this and we will see later how it is connected to the Hull-White Model. Although this is only a single-factor model, they quickly demonstrate how a common cross-currency model developed by Babbs in [13] can be adjusted to work with LGM. Babbs models the short rates using Hull-White, and correlated Brownian Motions, the FX rate is modelled as a Geometric Brownian Motion. By assigning the short rates to separate currencies it is then possible to obtain a Zero-Coupon Bond price under both a domestic and foreign currency. Resultingly, in the LGM model two numeraires for both domestic and foreign currencies are required, and eventually the foreign currency model is converted into the domestic currency. By modelling them separately and then converting to a single currency it allows for different rate dynamics and a correlation structure between the currencies. These are both of course also correlated to the FX rate. They find in [12] that the foreign numeraire and bond are martingales when converted to the domestic currency and divided by the domestic numeraire. These are desirable properties when pricing assets or derivatives.

They propose using overnight index swaps, and vanilla interest rate swaps for calibrating the LGM model to the market discount and forward curves respectively. This can be done for both the domestic and foreign short rates. Similarly, the FX rate can be calibrated to market data using FX forwards and Vanilla European FX options. Correlations are calibrated based on computed covariations between the short-rate and FX models that are then equated to historical covariations observed from market data.

There are very few papers that use Neural Networks or deep learning methods to price these options, however, one was conducted early last year by Kandhai, Jain, and Hoencamp in [14] that attempts to use Neural Networks to create a semi-static replication of certain interest rate derivatives. However, the extension to cross-currency products, and their methodology is only demonstrated on fixed-floating interest rate swaps. They employ a method that is somewhat similar to the Monte-Carlo methods we have seen so far. A regression of the product of interest against standard interest rate products or risk factors is performed. For example, a Bermudan Swaption could be regressed against Zero-Coupon Bond prices or forward rates. They suggest approximating the regression step in American Monte Carlo using Neural Networks. Typically basis functions are chosen and their coefficients are estimated using regression in order to determine the necessary relationship, but a Neural Network can also be used to determine this by using the same input and test data for an appropriate loss function. Standard least squares methods seek to minimize the square error between the estimated and observed values in regression. It is suggested in [14] to use an equivalent, weighted version, of the typical squared error loss function. They ultimately conclude that the use of Neural Networks in such a problem is effective for avoiding the computational demand required for performing multiple sequential regressions in the American Monte Carlo approach.

A "Rollback" method specifically for pricing single-currency Bermudan Swaptions is described by Hagan in [1]. This is a backward induction method, which is quite a common approach to these problems involving optimal stopping features. Such methods often require a large number of simulations or training of a specific model to make predictions. Complications can easily arise with these, for example training or simulating on the risk-neutral measure can often lead to erroneous behaviour when approximations are made based on real-world data [15]. Typically the expected value of the option conditional on information at a previous time point must be calculated using these methods. But Hagan describes a method involving weighted sums and interpolation to achieve this. Subsequently, one can retrieve a pricing estimate without the need for model training and large numbers of simulations.

Chapter 1

Interest Rate Products

1.1 The Basics

1.1.1 Zero Coupon Bonds and Rates

Before we can consider basic products such as FRAs and Swaps we should familiarise ourselves with a few definitions essential for understanding and pricing these. We will follow many of the definitions and derivations outlined in [16].

We begin with the stochastic discount factor

Definition 1.1.1 (Stochastic Discount Process)**.** The stochastic discount process D(t,T) is defined:

$$D(t,T) := \exp(-\int_t^T r_s ds)$$

and can be thought of as the price it would cost at time t to receive an amount 1 at time T, r_t is the short rate. [16, Definition 1.1.2., page 3]

The Zero-Coupon Bond is the crux of pricing many of the interest rate products we will deal with, so it is necessary to define it here.

Definition 1.1.2 (Zero Coupon Bond)**.** We denote $P(t,T)$ as the Zero-Coupon Bond price, this is a contract entered at time t, that is guaranteed to pay an amount 1 to the buyer at future time T, this relates to the stochastic discount factor by the following relation:

$$P(t,T) := \mathbb{E}_t^{\mathbb{Q}}[D(t,T)] = \mathbb{E}_t^{\mathbb{Q}}[\exp(-\int_t^T r_s ds)]$$

Where $\mathbb{Q}$ is the risk-neutral measure [16, Equation 3.2, page 51], we will discuss this more closely in Chapter 2. An explicit formula is dependent on our choice of model for the short rate r_t.

From here we can define some of the more commonly used rates.

Definition 1.1.3 (LIBOR Rate)**.** The LIBOR rate is a simple compounding spot interest rate, often considered between a starting time t and maturity time T, denoted $L(t,T)$. It is the rate at which an initial amount $P(t,T)$ must accrue interest in order to be of value 1 at time T. Correspondingly it is defined as follows:

$$L(t,T) := \frac{1 - P(t,T)}{(T-t)P(t,T)}$$

[16, Definition 1.2.4, page 6]

We will see later how the pricing of Swap contracts depends on FRAs, so we define these first.

Definition 1.1.4 (Forward Rate)**.** An FRA is a contract in which two parties agree to exchange a future fixed payment at a rate K for a future floating payment based on the LIBOR rate $L(t,T)$. The contract spans a time interval $[S,T]$, with the spot rate resetting at time S, and the exchange

occurring at time T. So the value of the contract to the party paying the fixed rate K at the time of exchange T is.

$$(T-S)(L(S,T)-K)$$

For the party paying the floating rate, the value is simply the negative of this. Note that the value above can be negative if the LIBOR rate over this period is lower than the fixed rate K, in which case the party paying the fixed rate makes a loss. An equivalent consideration can be made from the perspective of the party paying the floating rate. The forward rate itself arises naturally from this as the value of K that solves

$$(T-S)(L(S,T)-K)=0$$

By substituting $L(S,T)$ with the expression in Definition 1.1.3

$$(T-S)K-\frac{1}{P(S,T)}+1=0 \tag{1.1.1}$$

The payoff $\frac{1}{P(S,T)}$ at time T can be replicated as follows. By holding one bond at time t, with maturity S, we receive an amount 1 at time S, we can then use this to purchase Bonds at time S with maturity T, their price is simply $P(S,T)$, so we can purchase $\frac{1}{P(S,T)}$ of these Bonds, giving a payoff of $\frac{1}{P(S,T)}$ at T, the cost of setting up this portfolio is $P(t,S)$ for the original S dated Bond [17]. So the time t value of the left-hand side of (1.1.1) becomes

$$(T-S)KP(t,T)-P(t,S)+P(t,T)=0$$

Isolating K we find the Forward Rate to be defined as:

$$F(t,S,T):=\frac{1}{T-S}\left(\frac{P(t,S)}{P(t,T)}-1\right)$$

[16, Pages 11 and 12]

1.1.2 Swap Products

Finally, we are in a position to define and price an Interest Rate Swap.

Interest Rate Swap (IRS)

This product can be considered as a series of consecutive FRAs, in which two parties exchange a series of floating and fixed-rate payments. We will also use $\tau(t,T):=T-t$ to denote the time interval from now on. One factor that determines the price of the swap is the schedule of floating and fixed-rate payments which we will denote as

$$\begin{aligned}&T:=\{T_1,...,T_n\}\,\text{Floating rate payment schedule,}\\&S:=\{S_1,...,S_m\}\,\text{Fixed rate payment schedule,}\end{aligned}$$

Where T_0, S_0 are the first reset dates for calculating rates. Then at times T_i and S_j each leg of the contract has value:

$$\begin{aligned}&L(T_{i-1},T_i)\tau(T_{i-1},T_i), i=1,...,n,\ \textbf{Floating Leg,}\\&K\tau(S_{j-1},S_j), j=1,...,m, \textbf{Fixed Leg,}\\&\text{discounting these back to time } t<T_0,\\&\mathbf{IRS}_{\text{float},i}:=P(t,T_i)F(t,T_{i-1},T_i)\tau(T_{i-1},T_i), i=1,...,n,\\&\mathbf{IRS}_{\text{fixed},j}:=KP(t,S_j)\tau(S_{j-1},S_j), j=1,...,m,\end{aligned}$$

In the case of receiving the floating leg and paying the fixed leg, the price of the swap is

$$\mathbf{IRS}(t,T,S)=\sum_{i=1}^{n}\mathbf{IRS}_{\text{float},i}-\sum_{j=1}^{m}\mathbf{IRS}_{\text{fixed},j} \tag{1.1.2}$$

In the case of the payer swap the price is simply the negative of this. We can simplify (1.1.2) as follows

$$\mathbf{IRS}(t,T,S) = (R^{\text{IRS}}(t,T,S) - K)\sum_{j=1}^{m} P(t,S_j)\tau(S_{j-1},S_j)$$

where R^{IRS} is the Swap Rate, given by the value of the fixed rate K that makes the Swap price 0, and can be shown to be:

$$R^{\text{IRS}} := \frac{\sum_{i=1}^{n} P(t,T_i)F(t,T_{i-1},T_i)\tau(T_{i-1},T_i)}{\sum_{j=1}^{m} P(t,S_j)\tau(S_{j-1},S_j)}$$

[18, Definition 4.3.4, Page 26]

Swaption

This is an option on a Swap. Usually, these products give the holder the option to enter into an interest rate Swap at a specified time. European Swaptions allow for only a single exercise date, typically before the first reset of the Swap. Options can be made for either the receiver or payer end of the Swap. If we consider a Swaption in which we receive the fixed rate payments, then an investor would only enter the contract if the value of the fixed leg is greater than the floating leg. So the present value of the payoff is

$$\left(\sum_{j=1}^{m} P(t,S_j)\tau(S_{j-1},S_j)K - \sum_{i=1}^{n} P(t,T_i)F(t,T_{i-1},T_i)\tau(T_{i-1},T_i)\right)^{+}$$

For a Swaption receiving the floating leg the present value is instead

$$\left(\sum_{i=1}^{n} P(t,T_i)F(t,T_{i-1},T_i)\tau(T_{i-1},T_i) - \sum_{j=1}^{m} P(t,S_j)\tau(S_{j-1},S_j)K\right)^{+}$$

Pricing of European Swaptions typically requires a model in order to be priced analytically, see [16] or [12]. However we are interested in Bermudan Swaptions, this means that unlike European Swaptions there is a finite set of exercise dates. The payoff, when receiving the fixed leg, at a given exercise date t^* could then be written as

$$\left(\sum_{j=j^*}^{m} P(t^*,S_j)\tau(S_{j-1},S_j)K - \sum_{i=i^*}^{n} P(t^*,T_i)F(t^*,T_{i-1},T_i)\tau(T_{i-1},T_i)\right)^{+}$$

where S^*, T^*, j^*, i^* are the earliest payments dates and corresponding indexes after the exercise date t^*. Due to the multiple opportunities to exercise early the option holder must decide whether to exercise or wait for a future exercise date. Subsequently, we can no longer produce an analytical formula to price these products. We will see later the methodologies that are available to price these products. These formulations have been adapted from [16].

Overnight Indexed Swaps (OIS)

The Overnight Indexed Swaps are quite similar to regular interest rate Swaps, the main difference being that the floating rate is compounded daily using overnight rates rather than annually or semi-annually like what might be seen for standard Swaps. The term structure of the coupons is similar to a normal swap as well despite the daily compounding. For this reason the fixed leg has the same exact price as a normal swap, the floating leg replaces the Libor rate we would normally observe with the compounded overnight rate which we call $\tilde{R}_{on}$, we define it in the following way.

$$\tilde{R}_{on}(T_{i-1},T_i) := \frac{1}{\tau(T_{i-1},T_i)}\left(\prod_{k=1}^{n_i}(1 + R_{on}(T_{i,k-1},T_{i,k})\tau(T_{i,k-1},T_{i,k})) - 1\right)$$

$$(T_{i-1},T_i] = \bigcup_{k=1}^{n_i}(T_{i,k-1},T_{i,k}]$$

Here $R_{on}(T_{i,k-1}, T_{i,k})$ is the quoted overnight rate between days $k-1$ and k over the coupon period $(T_{i-1}, T_i]$. This is otherwise priced in the same way as a standard Swap, so its price and Swap rate can be shown to be the following respectively.

$$\mathbf{OIS}(t, T, S, R_{on}) = (R_{on}^{OIS}(t, T, S) - K) \sum_{j=1}^{m} P(t, S_j)\tau(S_{j-1}, S_j)$$

$$R_{on}^{\text{OIS}}(t, T, S) = \frac{\sum_{i=1}^{n} P(t, T_i)\tilde{R}_{on}(t, T_i)\tau(T_{i-1}, T_i)}{\sum_{j=1}^{m} P(t, S_j)\tau(S_{j-1}, S_j)}$$

The notation and formulations used here follow what is used in [18].

1.2 Cross-Currency Products

We are ready to introduce the most simplified version of the products of interest to this thesis, the Cross-Currency Swap. These are Swap products that allow two parties to exchange debt obligations and cashflows in other currencies. One example of this is exchanging a loan taken out in EUR with another party who has taken a loan out in USD. Each party receives the initial loan amount in the other currency and makes repayments in that currency as well. This is done in order to receive cheaper loans or funding, to manage liquidity in multiple currencies as well as for purely speculative purposes. The notation, definitions and understanding for this section are drawn from [3].

1.2.1 Features

These are OTC products and come in a variety of forms. For example, any combination of fixed-float, float-float, or even fixed-fixed swap legs can be obtained. A single currency for valuation purposes must also be chosen, and any necessary exchange rate calculations are performed accordingly. Naturally, there is a risk of default with these products in being unable to meet the necessary debt obligations, particularly when one has to account for both FX and interest rate movements. Accordingly, these products can be Marked to Market, which means that resets of the notional amounts are required in order to reduce these potential risks. Notionals are adjusted depending on the movement of the FX rates, and exposures to these rates are re-adjusted by setting the coupon notionals to be equivalent in each currency. Essentially whenever the FX rate leads to differing levels of coupon payments this difference is settled by a cash exchange so that these coupons remain equivalent on either leg of the contract for its duration. This reduces the amount of risk created by long-term FX exposure. In such cases, the contract holders will have to decide which leg of the contract will be adjusted to account for these resets and which leg will remain fixed.

Alongside this OTC markets require collateral posting in case of counter-party default. Since this offers less risk in the case of default than unsecured loans or debt obligations, investors are in turn rewarded with lower interest rates for repayments. For the sake of simplicity though we consider trades where these factors are not required.

We denote the currencies as *ccy* for the discount or cashflow currency. We add some additional notation to the Zero-Coupon Bond to help highlight this

$$P(t, T)^{ccy}$$

For example $P(t, T)^{EUR}$ is a Zero-Coupon Bond in Euro.

In order to be able to effectively work with these products it is necessary to convert each aspect into a single currency. This is done using FX rates, which we will denote as $x^{FOR/DOM}(t)$, this is the cost of 1 unit of foreign currency in the domestic currency. Where FOR is the foreign currency and DOM the domestic [3, Section 2, Pages 5-10].

1.2.2 Pricing

There is a great deal involved in the pricing of the Cross-Currency Swaps, so similar to [3], it will be broken down into multiple steps that involve new notation. In order to identify the necessary aspects and features of the option we will first introduce the notation suggested in [3, 3.1, Page 11]. Let Ω_{Xccy} be the set of parameters or features for a given contract, and Ω_{ccy} denote the currency

Table 1.1: Table of Cross-Currency Swap Features

Notation	Meaning
$\Omega_{Xccy} = MTM$	Marked to Market Cross-Currency Swap
Ω_{ccy}	Swap leg in currency ccy
$\Omega_{ccy} = Fixed$	Fixed trade leg in currency ccy
$\Omega_{ccy} = Floating$	Floating trade leg in currency ccy

leg of Swap, be that foreign or domestic. Then we can easily express these features in Table 1.1. $PV(\Omega_{Xccy})$ also denotes the present value of the Cross-Currency Swap, this applies similarly to the other notations above. We will not be considering Marked to Market Swaps so we exclude the pricing involved with this.

Notional Exchanges

The notional exchanges at the start and end of the contract account for the initial endowment and eventual repayment of any funding or loan amount. So the present value of these payments is given by:

$$\begin{aligned} PV(\text{Exchange}, \Omega_{FOR}) &= N^{FOR} x(t_n)^{FOR/DOM} P(0,t_n)^{FOR} - N^{FOR} x(t_0)^{FOR} P(0,t_0)^{FOR} \\ PV(\text{Exchange}, \Omega_{DOM}) &= N^{DOM} P(0,t_n)^{DOM} - N^{DOM} P(0,t_0)^{DOM} \end{aligned} \tag{1.2.1}$$

The first term accounts for the repayment of the funding, and the second is the initial receipt of funds. The equations in (1.2.1) account for all of the notional exchanges in the contract [3, Pages 12-16].

Coupons

Since these swaps can appear as with any pair of floating and or fixed legs we will consider how to price the coupons of floating and fixed legs separately.

The fixed-rate coupons with rate r_{ccy} have the following present value

$$PV(\text{Cpn}, \Omega_{ccy} = Fixed, r_{ccy}) = \sum_{j=1}^{m} N^{ccy} \Psi(S_j, ccy) r_{ccy} \tau(S_{j-1}, S_j) P(0, S_j)^{ccy} \tag{1.2.2}$$

We can similarly define the floating rate coupons with a spread s_{ccy} and based on the LIBOR rate as follows:

$$PV(\text{Cpn}, \Omega_{ccy} = Float, s_{ccy}) = \sum_{i=1}^{n} N^{ccy} \Psi(T_i, ccy)(L(T_{i-1}, T_i) + s_{ccy}) \tau(T_{i-1}, T_i) P(0, T_i)^{ccy} \tag{1.2.3}$$

Where

$$\Psi(t, ccy) := \begin{cases} 1 & \text{, if } ccy = DOM \\ x(t)^{FOR/DOM} & \text{, if } ccy = FOR \end{cases}$$

We now have all of the necessary pieces to price one of these swaps. By simply summing together the present values and taking the difference between foreign and domestic legs we can calculate the price of the option as:

$$\begin{aligned} PV(\Omega_{Xccy}) = \phi \, [&PV(\text{Cpn}, \Omega_{FOR}) + PV(\text{Exch}, \Omega_{FOR}) \\ &- PV(\text{Exch}, \Omega_{DOM}) - PV(\text{Cpn}, \Omega_{DOM})] \end{aligned}$$

Where we allow ϕ to be 1 for the foreign leg holder's perspective and -1 for the domestic leg holder's perspective. By plugging in the formulas developed in equations (1.2.1), (1.2.2), and (1.2.3), we can find the price for the Cross-Currency Swap [3, 3.7, Page 17].

1.2.3 Optionality

We are of course interested in a Bermudan option on this product. This tends to present itself in one of two ways. An "Exercise into" Swaption is one that allows the holder to enter into a Swap at a time of their choosing. A "Callable" Swaption is a Swaption that allows the holder to exit a Swap at a time of their choosing. When the option holder exercises the option the necessary notional exchanges are moved to that date. A Callable Swap can be priced as the underlying Swap and an "Exercise Into" Swaption for the opposite leg of the Swap [1]. Heuristically, the optimal time to enter a Swap receiving one leg is the optimal time to exit a swap receiving the other leg and vice versa. So by the Fundamental Theorem of Asset Pricing since the payoff of a swap can be replicated in this way we require that the initial price of this combination of Swaptions be equal to that of the underlying Swap [1, Section 2]. This concept will be necessary later for pricing our Swaptions.

For either of these types of Swaptions, it is necessary to determine the optimal exercise time. As we have seen from the literature review this is typically approached by modelling the underlying forward or short rates and using Monte Carlo methods to determine the optimal exercise time. In the single-currency case, this can simply be performed using a backward induction method and numerical integration to approximate necessary expected values [1, Section 4], similar in spirit to the backward induction of binomial tree models for American options. In the following chapter, we will discuss our choice of model.

Chapter 2

Modelling

2.1 Preliminary Theory

Before we can introduce and derive the models we will be working with it is necessary to consider some preliminary concepts. We should first introduce the notion of martingales and equivalent martingale measures.

Definition 2.1.1 (Martingale)**.** A Martingale is a stochastic process or sequence of random variables denoted as $X = \{X_n : n \geq 0\}$ such that

1. $\mathbb{E}(|X_n|) < \infty$
2. $\mathbb{E}(X_{n+1}|\mathcal{F}_n) = X_n$

where $\mathcal{F}_n$ is the filtration or relevant information set up to period n [19].

Then the Equivalent Martingale Measure is defined as follows.

Definition 2.1.2 (Equivalent Martingale Measures)**.** For a measurable probability space Ω, two probability measures $\mathbb{Q}$ and $\mathbb{P}$ are said to be equivalent if for any set $A \subseteq \Omega$, $\mathbb{Q}(A) = \mathbb{P}(A) = 0$ [20].

Then $\mathbb{Q}$ is an equivalent martingale measure if

1. $\mathbb{Q}$ is equivalent to $\mathbb{P}$ on Ω
2. Any asset or derivative divided by the numeraire associated with $\mathbb{Q}$ is a martingale under the measure $\mathbb{Q}$.

[17]

2.1.1 Fundamental Theorems of Asset Pricing

Arbitrage can be identified as an opportunity for risk-free profit. These can occasionally be observed in actual markets, however, they tend to become infeasible once a bid-ask spread is introduced or are extremely short-lived. Assuming that investors are perfectly reasonable, such opportunities will be exploited until changes in supply and demand cause the price to shift subsequently eliminating any chance for risk-free profit. For these reasons, Arbitrage Pricing Theory theorizes that arbitrage opportunities are essentially nonexistent and that we should price assets and products accordingly [21, Section 1]. This is a necessary assumption in many financial models such as Black-Scholes [22].

This is synonymous with risk-neutral pricing, and so depends greatly on the idea of the Risk-Neutral Measure. When considering probabilities the physical measure can be considered as what we can observe and measure empirically from data. The Risk-Neutral Measure can be thought of as a reweighting of this physical measure in order to eliminate arbitrage opportunities. [23].

Theorem 2.1.3 (First Fundamental Theorem of Asset Pricing)**.** *The existence of an equivalent martingale measure is synonymous with an arbitrage-free market model [17][Chapter 10].*

Combining this with the second point in definition 2.1.2, we understand now that the risk-neutral measure is the probability measure used that allows the discounted expected value of a product's payoff to equal the product's price [24]. This is also written in Bjork's book [17] as the General Pricing Formula.

Theorem 2.1.4 (General Pricing Formula)**.** *The arbitrage-free price $\Pi(t, X)$ of an asset or derivative with payoff X is*

$$\Pi(t, X) = N(t)\mathbb{E}^{\mathbb{Q}}\left[\frac{X}{N(T)}\middle|\mathcal{F}_t\right]$$

where $\mathbb{Q}$ is the martingale measure and $N(t)$ the associated numeraire.

Much of the work done in pricing is centred around this General Pricing Formula.

2.1.2 Change of Numeraire

By changing Numeraire we can often simplify the necessary calculations required within our models, the Radon-Nikodym derivative plays a key role in this by allowing us to transition between measures.

Theorem 2.1.5 (Radon-Nikodym Theorem)**.** *Given a measure space (X, Σ), and two measures μ and ν. If ν is absolutely continuous with respect to μ, then there exists a Σ-measurable function f that maps X to the positive real line, such that for any set $A \subseteq X$,*

$$\nu(A) = \int_A f d\mu$$

Then f is called the Radon-Nikodym derivative and would be denoted here as $\frac{d\nu}{d\mu}$. More commonly this can be seen as $\frac{d\mathbb{Q}}{d\mathbb{P}}$, for changing between equivalent martingale measures $\mathbb{Q}$ and $\mathbb{P}$ [25].

We can now explain how to perform the change of numeraire using the toolkit described in [16]. Any positively valued asset that does not pay dividends can be used as a numeraire. They can be considered as a reference upon which the product is standardized, this most commonly appears as the bank account process. This can be changed to other reference assets such as the underlying stock [16, Section 2.2].

Given two equivalent martingale measures $\mathbb{Q}^{\mathbb{B}}$ and $\mathbb{Q}^{\mathbb{N}}$, and two associated numeraires B and N respectively, due to the martingale measure any payoff or product to be priced, X, relative to these numeraires is a martingale such that

$$\frac{X_t}{B_t} = \mathbb{E}^{\mathbb{Q}^B}\left[\frac{X_T}{B_T}|\mathcal{F}_t\right]$$
$$\frac{X_t}{N_t} = \mathbb{E}^{\mathbb{Q}^N}\left[\frac{X_T}{N_T}|\mathcal{F}_t\right]$$

The Radon-Nikodym derivative can also be written

$$\frac{d\mathbb{Q}^N}{d\mathbb{Q}^B} = \frac{N_T B_0}{B_T N_0} \tag{2.1.1}$$

which allows us to change measure and numeraire as follows

$$\mathbb{E}^N\left[\frac{X_T}{N_T}\right] = \mathbb{E}^B\left[\frac{X_T}{N_T}\frac{d\mathbb{Q}^N}{d\mathbb{Q}^B}\right]$$

[16].

Another important theorem involved in the change of measure is Girsanov's Theorem

Theorem 2.1.6 (Girsanov's Theorem)**.** *Let M be a continuous local martingale under the measure $\mathbb{P}$, then define the martingale Z by*

$$Z_t = \exp\left(M_t - \frac{1}{2}[M]_t\right)$$

where $[M]_t$ is the quadratic variation of M. If Z is uniformly integrable then we may define a new equivalent measure $\mathbb{Q}$ by

$$\frac{d\mathbb{Q}}{d\mathbb{P}} = Z_\infty$$

Then for a continuous local martingale X under $\mathbb{P}$, X-$[X,M]$ is a $\mathbb{Q}$ local martingale. [26]

2.1.3 Stochastic Calculus

Definition 2.1.7 (Wiener Process)**.** A Wiener Process or Brownian Motion is a random variable $W(t)$, often referred to as a process, satisfying the following conditions

1. $W(0) = 0$.
2. $W(T) - W(S) \sim N(0, T - S), T > S$.
3. W has independent increments.
4. $W(t)$ is almost surely continuous.

[27]

A corollary to Theorem 2.1.6 is particularly useful to us for defining new Brownian motions.

Corollary 2.1.8. *For a Brownian Motion $W(t)$ under $\mathbb{P}$, the process*

$$\hat{W}(t) := W(t) - [W, M]_t$$

is a $\mathbb{Q}$ Brownian Motion. [26]

Definition 2.1.9 (Itô Integral of elementary functions)**.** We first define the set of functions ν such that

$$f(t, \omega) : [0, \infty) \times \Omega \mapsto \mathbb{R}$$

and satisfies

1. $(t, \omega) \mapsto f(t, \omega)$ is $\mathcal{B} \times \mathcal{F}$ measurable, $\mathcal{B}$ is the Borel σ-algbera
2. $f(t, \omega)$ is $\mathcal{F}_t$ adapted
3. $\mathbb{E}[\int_S^T f(t, \omega)^2 dt] < \infty$

Consider an elementary function $\phi(t, \omega) \in \nu$ of the form

$$\phi(t, \omega) = \sum_j \alpha_j(\omega) \chi_{\{t_{j-1}, t_j\}}(t)$$

where α_j is $\mathcal{F}_{t_j}$ measurable. We can then define an Itô integral by

$$\int_S^T \phi(t, \omega) dW_t(\omega) = \sum_{j \geq 0} \alpha_j(\omega)[W_{t_{j+1}} - W_{t_j}](\omega)$$

[26]

This definition can then be extended to more general square-integrable functions [26]. Here we are more interested in highlighting how the Itô integral can be represented as the sum of independent increments of Brownian Motion, meaning that the integrals themselves can have Gaussian distributions. A useful result for dealing with expectations of Itô Integrals is the Itô Isometry.

Lemma 2.1.10 (Itô Isometry)**.** *For any integrable function f*

$$\mathbb{E}\left[\left(\int_S^T f(t, \omega) dW_t\right)^2\right] = \mathbb{E}\left[\int_S^T f^2(t, \omega) dt\right]$$

[27].

2.2 Linear Gauss Markov (LGM) Model

For the derivation and understanding of this model we will closely follow the work done in [12, Chapter 11]. The model is closely related to the Hull-White model for short rates given by:

$$dr_t = (\theta_t - \lambda_t r_t)dt + \sigma_t dW_t$$

Where r_t is the short-rate, θ_t is the time dependent mean, λ_t the rate of mean-reversion, σ_t is the volatility of the short rate, and $(W_t)_{t\geq 0}$ is a standard Brownian motion. This can be decomposed into two processes for its deterministic part and its stochastic part as follows.

$$\begin{aligned} d\mu_t &= (\theta_t - \lambda_t \mu_t)dt \\ dx_t &= -\lambda_t x_t dt + \sigma_t dW_t \\ r_t &= \mu_t + x_t \end{aligned}$$

Then since these are Ornstein Uhlenbeck processes they have solutions

$$\begin{aligned} \mu_t &= r_0 e^{-\int_0^t \lambda_s ds} + e^{-\int_0^t \lambda_s ds} \int_0^t e^{\int_0^s \lambda_u du} \theta_s ds \\ x_t &= e^{-\int_0^t \lambda_s ds} \int_0^t e^{\int_0^s \lambda_u du} \sigma_s dW_s. \end{aligned}$$

Some notation should be defined at this point as well.

$$\begin{aligned} \beta_t &:= \int_0^t \lambda_s ds \\ \alpha_t &:= e^{\beta_t} \sigma_t \\ H_t &:= \int_0^t e^{-\beta(s)} ds \end{aligned}$$

Then the drift and variance of the process r_t are respectively

$$\begin{aligned} \int_0^t \mu(s)ds &= -\ln(P(0,t)) + \frac{1}{2}\int_0^t (H_t - H_s)^2 \alpha_s^2 ds \\ \mathbb{V}[r_t] = \mathbb{V}[x_t] &= e^{-2\beta_t} \int_0^t \alpha_s^2 ds \end{aligned}$$

These are derived in the appendix A.1.2.

The Zero-Coupon Bond price has the following expression under this model

$$P(t,T) = \frac{P(0,T)}{P(0,t)} \exp\left(-(H_T - H_t)e^{\beta_t} x_t - \frac{1}{2}\int_0^t \left[(H_T - H_s)^2 - (H_t - H_s)^2 \right] \alpha_s^2 ds \right) \tag{2.2.1}$$

We derive this in appendix A.1.1. $P(0,T)$ can be obtained from market data. The bank account numeraire $B(t)$ can be written in terms of the model parameters by expressing r_t in terms of μ_t and x_t [12].

$$B(t) = e^{\int_0^t r(s)ds} = \frac{1}{P(0,t)} \exp\left(-\int_0^t (H_T - H_s)\alpha_s dW_s + \frac{1}{2}\int_0^t (H_t - H_s)^2 \alpha_s^2 ds \right) \tag{2.2.2}$$

Accordingly, the *reduced* (divided by the numeraire) price of the Zero-Coupon Bond can be found by dividing equation (2.2.1) by (2.2.2).

$$\frac{P(t,T)}{B(t)} = P(0,T) \exp\left(-\int_0^t (H_T - H_s)\alpha_s dW_s - \frac{1}{2}\int_0^t (H_T - H_s)^2 \alpha_s^2 ds \right) \tag{2.2.3}$$

At this point, we see that (2.2.2) and (2.2.3), depend on both random variables $\int_0^t (H_T - H_s)\alpha_s dW_s$ and $\int_0^t (H_t - H_s)\alpha_s dW_s$, which makes finding an analytical solution quite difficult. It is suggested

in [12] to reformulate this by introducing

$$y(t) := e^{\beta_t} x_t = \int_0^t \alpha_s dW(s)$$
$$\Psi(t) := \int_0^t H_s \alpha_s^2 ds$$
$$z(t) := y(t) - \Psi(t)$$

This leads to the following dynamics

$$dz(t) = dy(t) - d\Psi(t) = \alpha(t)dW(t) - H(t)\alpha^2(t)dt.$$

It is then observed in [12] that $z(t)$ can be converted into a martingale, by ensuring the SDE has 0 drift. Then after performing a change of numeraire we would require

$$dz(t) = \alpha(t)d\tilde{W}(t).$$

The necessary change of measure can be found using the change of numeraire toolkit in [16], resulting in the new measure

$$\tilde{W}(t) = W(t) + \int_0^t \frac{-H(s)\alpha^2(s)}{\alpha(s)} ds = W(t) - \int_0^t H(s)\alpha(s)ds.$$

By Girsanov's Theorem and Corollary 2.1.8, this is a valid Brownian Motion [16]. Then $z(t)$ is Gaussian with zero mean and variance $\xi(t) = \int_0^t \alpha_s^2 ds$ [12]. From here we can also calculate the Radon-Nikodym derivative as

$$Z(t) = \exp\left(\int_0^t H_s \alpha_s dW_s - \frac{1}{2}\int_0^t H_s^2 \alpha_s^2 ds\right), \tag{2.2.4}$$

this is derived more explicitly in appendix A.1.3. Then by the Radon-Nikodym equation in (2.1.1), we can rearrange to find the new numeraire

$$N(t) = B(t)Z(t) = \frac{1}{P(0,t)} \exp\left(H_t z_t + \frac{1}{2}H_t^2 \xi_t\right)$$

Finally, we can find the new *reduced* price of the Zero Coupon Bond under the LGM model by dividing the Zero Coupon Bond price in (2.2.1) by the new numeraire $N(t)$.

$$\tilde{P}(t,T,z_t) = \frac{P(t,T,z_t)}{N(t)} = P(0,T)\exp\left(-H_T z_t - \frac{1}{2}H_T^2 \xi_t\right) \tag{2.2.5}$$

We observe that the model has been simplified by now only depending on a single Gaussian random variable z_t, whereas previously it depended on two random variables [12].

The martingale property of course still holds for some value function $V(T,X)$ as

$$\mathbb{E}\left[\frac{V(T,X)}{N(T,X)}\middle| X(t) = x\right] = \frac{V(t,x)}{N(t,x)}$$

It is mentioned in [12] that this framework supports multiple curves. Curves for discounting and calculating forward rates over different tenors are required in the wake of the 2008 crash when the basis spread between the replicated and market-quoted forward rate was no longer 0. The replicated forward rate is the rate implied by the replicating argument for a Forward Rate Agreement [18]. For a discount curve d and any other rate curve c we might desire for calculating forward rates, we can find the Zero Coupon Bond price for the curve c by

$$P_c(t,T,z_t) = \frac{P_c(t,T,0)}{P_d(t,T,0)} P_d(t,T,z_t) \tag{2.2.6}$$

this implicitly assumes that all curves are driven by the same process z, according to [12], allowing the single-factor model to be used in the multi-curve context. It should be noted that some additional steps need to be taken before the model can be used in a multi-factor context, currently, this is only a single-factor model.

2.3 Cross-Currency LGM Model

In this context, we now introduce a second interest rate, the foreign interest rate, and the foreign exchange (FX). These are each modelled somewhat similarly to the interest rate in the single currency case, which we now call the domestic interest rate. We will also introduce a correlation structure in the model between each of these random processes. Many of the aspects in the single currency LGM model are still relevant in the multi-currency case. In [12, Chapter 12] this is developed from Babbs' Cross-Currency model [13]. The domestic and foreign rates are modelled as single-factor Hull-White models, and the FX rate is treated as a Geometric Brownian Motion. Each of these 3 factors are correlated with one another. We present these in the domestic bank account numeraire first, so letting r_d, r_f, x be the domestic, foreign interest rates and FX rate respectively the Babbs' model is given by

$$dr_d = \lambda_d[\theta_d(t) - r_d(t)]dt + \sigma_d dW_d^{\mathbb{Q}} \tag{2.3.1}$$

$$dr_f = \lambda_f[\theta_f(t) - r_f(t)]dt - \rho_{xf}\sigma_x\sigma_f dt + \sigma_f dW_f^{\mathbb{Q}} \tag{2.3.2}$$

$$dx/x = [r_d(t) - r_f(t)]dt + \sigma_x dW_x^{\mathbb{Q}} \tag{2.3.3}$$

$$dW_a dW_b = \rho_{ab}dt \quad a, b \in \{d, f, x\}$$

as presented in [12], where we again follow the derivation of this model. Since we are using the domestic bank account numeraire we acquire an extra drift term for the foreign interest rate due to a change of numeraire. This numeraire is associated with the measure $\mathbb{Q}$.

We can follow the process for the change of numeraire that was used in Section 2.2 for changing to the domestic and foreign numeraire for z_d and z_f respectively. The domestic case is identical, and the foreign case is similar but instead uses a foreign measure and numeraire. No additional steps are required at this stage for the FX process. Following this another change of numeraire is performed, so that each process is in the domestic measure. As a result we have new drift terms $\gamma_f(t)$ and $\mu_x(t)$ for the foreign process and the FX rate. Ultimately in [12] equations (2.3.1),(2.3.2), and (2.3.3) are reduced to

$$\begin{aligned} dz_d &= \alpha_d(t)dW_d^N \\ dz_f &= \gamma_f(t)dt + \alpha_f(t)dW_f^N \\ dx/x &= \mu_x(t)dt + \sigma_x(t)dW_x^N. \end{aligned} \tag{2.3.4}$$

At this stage however, we do not yet know the drift terms $\mu_x(t)$ and $\gamma_f(t)$, the derivation for z_d is identical to the single currency case as no currency change is required.

Before proceeding, the LGM numeraire for the domestic and foreign measures are presented in [12] as

$$N_i(t) = \frac{1}{P_i(0,t)} \exp\left(H_i(t)z_i(t) + \frac{1}{2}H_i^2(t)\xi_i(t)\right).$$

Where $i = \{d, f\}$ for domestic and foreign processes. The functions $H_i(t)$ and $\xi_i(t)$ are defined the same as in the single currency case. Alongside this $P_i(t,T)$ is the Zero Coupon Bond in currency i, it has the same expression as the single currency case, where the relevant values are replaced for the corresponding currency. Similarly, equation (2.2.6) for alternating between discount and forward rate curves for the Zero Coupon Bond applies to both currencies identically to the single currency case.

We can determine the drifts of each SDE in equations (2.3.4) by checking that certain no-arbitrage conditions hold. We now present a summary of the derivation performed in [12, Section 12.1], and present the details in the appendix A.2.1. In particular, we require that the LGM numeraire for the foreign currency, $N_f(t)$ converted to the domestic currency, discounted using the domestic numeraire is a martingale. More succinctly,

$$Y(t) = \frac{N_f(t)x(t)}{N_d(t)}$$

must have 0 drift.

We will however require a second condition or equation in order to isolate both of the drift terms. Subsequently, we consider the Zero Coupon Bond based on the foreign interest rate $P_f(t,T)$. This

too should be a martingale once we convert to the domestic currency and divide by the domestic numeraire. In other words, we also impose that

$$U(t) = \frac{P_f(t,T)x(t)}{N_d(t)}$$

has 0 drift.

From this the drift components are identified in [12] as

$$\begin{aligned}\gamma_f(t) &= -H_f(t)\alpha_f^2(t) + H_d(t)\alpha_d(t)\alpha_f(t) - \sigma_x(t)\alpha_f(t)\rho_{xf}\\ \mu_x(t) &= r_d(t) - r_f(t) + H_d(t)\alpha_d(t)\sigma_x(t)\rho_{dx}.\end{aligned}$$

It is also shown in [12], that the value of the foreign and domestic Zero Coupon Bonds divided by the corresponding numeraires are

$$\tilde{P}_i(t,T) = P_i(0,T)\exp\left\{-H_i(T)z_i(t) + \frac{1}{2}H_i^2(T)\xi_i(t)\right\}$$

where $i = \{f, d\}$. The derivation is the same as in the single currency case with equation (2.2.5), now with quantities for each currency. The value of the foreign bond under the domestic LGM measure is actually equivalent to the domestic case with the corresponding foreign parameters [12].

$$P_f(t,T) = \frac{P_f(0,T)}{P_f(0,t)}\exp\left(-(H_f(T) - H_f(t))z_f(t) - \frac{1}{2}(H_f^2(T) - H_f^2(t))\xi_f(t)\right)$$

The distributions and covariances of each process can also be determined. Quite simply z_d is gaussian with mean 0 due to its lack of drift, and has variance $\xi_d(t)$, identical to the single currency case. The foreign process z_f has a drift so has non-zero mean.

$$\mathbb{E}[z_f(t)] = \mathbb{E}[z_f(0) + \int_0^t \gamma_f(s)ds + \int_0^t \alpha_f(s)dW_f(s)] = z_f(0) + \int_0^t \gamma_f(s)ds$$

Where we have used the fact that Itô integrals of square-integrable functions have expectation 0, [26, Theorem 3.2.1]. Then by using Itô's Isommetry, we can find the variance to be $\xi_f(t)$, we provide a brief derivation in appendix A.2.2. So this is also a Gaussian random variable with the given mean and variance. Finally, since the FX process $x(t)$ is a Geometric Brownian Motion, it is lognormally distributed. So $\ln(x(t))$ is normally distributed. We can write the expression for $\ln(x(t))$ as follows

$$\begin{aligned}\ln(x(t)) =& \ln\left(\frac{P_f(0,t)}{P_d(0,t)}\right) - \frac{1}{2}\int_0^t \sigma_x^2(s)ds + \int_0^t \xi_d(s)H_d(s)H_d'(s)ds - \int_0^t \xi_f(s)H_f(s)H_f'(s)ds\\ &+ \rho_{dx}\int_0^t \alpha_d(s)H_d(s)\sigma_x(s)ds - \int_0^t (H_f(t) - H_f(s))\gamma_f(s)ds\\ &+ \int_0^t (H_d(t) - H_d(s))\alpha_d(s)dW_d(s) + (H_d(t) - H_d(0))z_d(0)\\ &- \int_0^t (H_f(t) - H_f(s))\alpha_f(s)dW_f(s) - (H_f(t) - H_f(0))z_f(0)\\ &+ \int_0^t \sigma_x(s)dW_x(s) \quad [12].\end{aligned}$$

We provide a derivation of this in appendix A.2.3. The mean of the log of the FX rate is also written in [12] as

$$\begin{aligned}\mathbb{E}[\ln(x(t))] =& \ln\left(\frac{P_f(0,t)}{P_d(0,t)}\right) - \frac{1}{2}\int_0^t \sigma_x^2(s)ds + \int_0^t \xi_d(s)H_d(s)H_d'(s)ds - \int_0^t \xi_f(s)H_f(s)H_f'(s)ds\\ &+ \rho_{dx}\int_0^t \alpha_d(s)H_d(s)\sigma_x(s)ds - \int_0^t (H_f(t) - H_f(s))\gamma_f(s)ds\\ &+ H_d(t)z_d(0) - H_f(t)z_f(0)\end{aligned}$$

and the variance is found by applying Itô's Isometry

$$\begin{aligned}\texttt{Var}[\ln(x)] = &\int_0^t (H_d(t) - H_d(s))^2 \alpha_d(s) ds \\ &-2\rho_{fd} \int_0^t (H_d(t) - H_d(s))(H_f(t) - H_f(s))\alpha_d(s)\alpha_f(s) ds \\ &+2\rho_{xd} \int_0^t (H_d(t) - H_d(s))\alpha_d(s)\sigma_x(s) ds \\ &-2\rho_{xf} \int_0^t (H_f(t) - H_f(s))\alpha_f(s)\sigma_x(s) ds \\ &+ \int_0^t (H_f(t) - H_f(s))^2 \alpha_f^2(s) ds \\ &+ \int_0^t \sigma_x^2(s) ds\end{aligned}$$

[12].

The covariance between each process can also be calculated. The covariances of the increment of each process over $[u, t]$ is given in [12] by the following equations.

$$\begin{aligned}\texttt{Cov}[\Delta z_d, \Delta z_f] &= \rho_{fd} \int_u^t \alpha_i(s)\alpha_d(s) ds \\ \texttt{Cov}[\Delta z_i, \Delta \ln(x)] &= \rho_{di} \int_u^t \alpha_f(s)(H_d(t) - H_d(s))\alpha_d(s) \\ &\quad - \rho_{fi} \int_u^t \alpha_i(s)(H_f(t) - H_f(s))\alpha_f(s) ds \\ &\quad + \rho_{ix} \int_u^t \alpha_i(s)\sigma_x(s) ds \\ &\quad i = \{f, d\}\end{aligned}$$

One distinct advantage of the LGM model at this stage is the analytical tractability of the model, each of the integrals above can be calculated explicitly. This also allows for exact simulation of the processes, meaning we avoid any discretisation error introduced by methods such as Euler-Maruyama or Milstein. Furthermore the expressions for the Zero Coupon Bonds rely on the observable value for the Zero Coupon Bond from the discount curve, meaning the model is automatically calibrated to a yield curve [12].

Some calibration steps are still required though, the mean reversion and volatility parameters for the interest rates can be calibrated from European Swaptions [12]. As we mentioned in Section 1.1.2, there often exists an analytical formula for European Swaptions, the LGM model is no exception. Calibrating the FX process requires FX forwards and European FX options. Finally in order to determine the correlation parameters historical data is typically used to provide an estimation [12]. In particular, the variance and covariations of past Zero-Coupon Bond prices are computed. The historical covariations $\texttt{HistCov}_{i,j}$ between currencies i and j are given by

$$\texttt{HistCov}_{i,j}(S, T) = \frac{1}{\Delta} \sum_{k=1}^{N} (P_i(t_k, t_k + T) - P_i(t_{k-1}, t_{k-1} + T)) \cdot (P_j(t_k, t_k + S) - P_j(t_{k-1}, t_{k-1} + S))$$

Where $t_N - t_0 = \Delta$. By calculating the covariations based on model parameters, $f_{i,j}$, as

$$f_{i,j}(t, \Delta) = \frac{\frac{1}{\Delta} \int_t^{t+\Delta} \sigma_i(s)\sigma_j(s) ds}{\sqrt{\frac{1}{\Delta} \int_t^{t+\Delta} \sigma_i^2(s) ds \cdot \frac{1}{\Delta} \int_t^{t+\Delta} \sigma_j^2(s) ds}}$$

We can then find the correlation parameter between interest rates as

$$\rho_{i,j} = \frac{1}{f_{i,j}(t, \Delta)} \frac{\texttt{HistCov}_{i,j}(t, \Delta)}{\sqrt{\texttt{HistCov}_{i,i}(t, \Delta)\texttt{HistCov}_{j,j}(t, \Delta)}}$$

[12]. We can calculate the correlation between the log of the FX process and the individual interest rates in a similar manner, by simply replacing one of the Zero-Coupon Bonds with the log of the FX where appropriate.

$$\texttt{HistCov}_j(t,\Delta) = \frac{1}{\Delta}\sum_{k=1}^{N} \ln\left(\frac{X(t_k)}{X(t_{k-1}}\right) \cdot (P_j(t_k, t_k+\Delta) - P_j(t_{k-1}, t_{k-1}+\Delta)$$

$$f_{x,j}(t,\Delta) = \frac{\frac{1}{\Delta}\int_t^{t+\Delta} \sigma_x(s)\sigma_j(s)ds}{\sqrt{\frac{1}{\Delta}\int_t^{t+\Delta} \sigma_x^2(s)ds \frac{1}{\Delta}\int_t^{t+\Delta} \sigma_j^2(s)ds}}$$

Meaning we can calculate the correlation between the FX and the interest rate for currency j as

$$\rho_{xj} = \frac{1}{f_{x,j}(t,\Delta)} \frac{\texttt{HistCov}_{x,j}(t,\Delta)}{\sqrt{\texttt{HistCov}_{x,x}(t,\Delta)\texttt{HistCov}_{j,j}(t,\Delta)}}$$

[12].

Fortunately, many of the calibration steps had already been performed prior to our work, so we excluded any excessive detail on how this is performed.

Chapter 3

Hagan's Rollback Method of Pricing

We will first detail the method introduced by Hagan in [1] to price Bermudan Swaptions, using the "Rollback" method, specifically for the LGM model. This is a backward induction method that works similarly to pricing American options on a binomial tree, or with finite difference methods. We adapt from [28], the standard steps for such backward induction methods are as follows:

Algorithm 1: Backward Induction Algorithm

Let x_k be the state variable for the driving stochastic process, and t_j the time variable, then $V_{k,j}$ is the corresponding option value, for $k \in \{0, ..., n\}$, $j \in \{0, ..., m\}$. $P_{k,j}$ is the payoff of the option.

$V_{k,m} = P_{k,m}$

for $j \in 1, ..., m$ **do**

  for $k \in 1, ..., n$ **do**

    $V_{k,j-1} = \max\left\{\mathbb{E}\left[V_j e^{-r(t_j - t_{j-1})} | \mathcal{F}_{j-1}, X = x_k\right], P_{k,j-1}\right\}$

Repeat to find V_0 the initial price of the option

This algorithm takes the maximum between the continuation value, given by the conditional expectation, and the payoff function evaluated at a given exercise date. In the case of Bermudan options, we restrict our choices of t_j to the finite number of exercise points. Due to the discrete nature of this method, we can actually price it more accurately than an American version. We provide a more in-depth description of the algorithm later in Algorithm 2.

The issue of mismatched risk-neutral training data and real-world test data for training a model [15] is also circumvented as no model needs to be trained for this method. Monte Carlo methods typically involve a regression step that can create issues when working with real data. In that case simulations would be generated in the risk-neutral measure, but when we introduce real data there is no guarantee that it will be risk-neutral. This problem tends to arise semi-frequently when trying to price these products, and so this project was born of the necessity for an alternative more accurate method.

3.1 Single-Currency Rollback on Bermudan Swaptions

In section 2.2 we defined the functions $H(t)$ and $\xi(t)$, these can be assumed known after calibrating to market data. We also saw that the model depends on the single factor z_t. We denote the value of the Swaption at time t for rate factor z as $V^{full}(t, z)$. Throughout the algorithm we work under the LGM measure, so we now divide by the numeraire $N(t, z)$, so that we have $V(t, z) = \frac{V^{full}(t,z)}{N(t,z)}$ as the "reduced value" of the option. At $t = 0$ the value of the numeraire is 1, so we do not have to worry about converting back to the full quantity at the end of our calculation [1].

3.1.1 Discretisation

We first define a grid for the process z, this is considered in terms of standard deviations, λ_z is the number of points per standard deviation, and N_z is the number of standard deviations. Given exercise time t_j^{ex}, the step size at that time is given as $h_j = \frac{\sqrt{\xi_j}}{\lambda_z}$, we double the quantity $m_z = N_z\lambda_z$ to be the number of points on the grid. Subsequently, our grid is defined by

$$z_k^{(j)} = h_j(k - m_z), k = 0, 1, ..., 2m_z, j = 1, 2, ..., J.$$

Then $V_{j,k} = V(t_j^{ex}, z_k^j)$ is the option value at the grid point corresponding to exercise time t_j^{ex} and the rate process z_k^j, and similarly the payoff of the Swaption at the same grid point is denoted as $P_{j,k}$ [1].

A Gaussian random variable y is discretised in a similar way to the factor z, such that

$$h_y = \frac{1}{\lambda_y}, m_y = N_y\lambda_y$$
$$y_i = h_y(i - m_y), \text{for } i = 0, 1, ..., 2m_y.$$

We will use this y later in order to approximate certain conditional expectations.

3.1.2 Backward Induction

It is optimal to exercise these options when the immediate payoff exceeds the value of holding the option until the next exercise date. At the final exercise date the value of not exercising is clearly 0, as we will not enter the contract. Accordingly, we start at the final exercise date t_J^{ex} and work backwards.

$$V_{J,k} = \max\{P_{J,k}, 0\}$$

We now have to determine at the earlier time point whether we should exercise now or wait until the next exercise date. We can express the future value of the option when we choose not to exercise as $\mathbb{E}[V(T, z_T)|F_t]$, this is commonly referred to as the continuation value. Using the martingale property, and the fact that z_T is Gaussian with 0 mean and variance $\xi(T)$ we have that

$$\mathbb{E}[V(T, z_T)|\mathcal{F}_t] = V(t, z_t)$$
$$V(t, z_t) = \frac{1}{\sqrt{2\pi\Delta\xi}} \int_{-\infty}^{\infty} e^{-(z_T - z_t)^2/2\Delta\xi} V(T, z_T) dz_T$$
$$\Delta\xi = \xi(T) - \xi(t)$$

Where we assume the initial value of z_T is z_t [1]. Performing a change of variable, $z_T = z_t + \sqrt{\Delta\xi}y$ we find the continuation value at a given grid point is

$$V^+(t_{j-1}^{ex}, z_k^{j-1}) = \frac{1}{\sqrt{2\pi}} \int_{\infty}^{\infty} e^{-y^2/2} V(t_j^{ex}, z_k^{j-1} + y\sqrt{\xi_j - \xi_{j-1}}) dy \quad [1] \tag{3.1.1}$$

This is the present value of the option if we choose not to exercise immediately at time t_{j-1}^{ex} and grid point z_k^{j-1}.

In order to step back over a single time point we use the inductive formula

$$V_{j-1,k} = \max\{P_{j-1,k}, V_{j-1,k}^+\}, \quad k = 0, 1, ..., 2m_z, j = 1, 2, ..., J$$

based on the logic that we should exercise now if it is more profitable than waiting until the next exercise date. We repeat this working backwards along the grid until we reach the initial time point and find our price estimate $V_{0,0}$ [1]. However, we still have the problem of evaluating the integral in (3.1.1), we explore in the following section how to approximate this.

3.1.3 Weighted Sum

It is suggested in [1] that the integral in (3.1.1) be approximated by a weighted sum. Though it is not specified in [1], this weighted sum is an application of the Gaussian quadrature numerical

integration technique [29]. The weights are given in [1] as

$$\begin{aligned} w_i &= \int_{y_i-h_y}^{y_i+h_y} \left(1 - \frac{|y_i - y|}{h_y}\right) \frac{e^{-y^2/2}}{\sqrt{2\pi}} dy \\ &= \left(1 + \frac{y_i}{h_y}\right) \Phi(y_i + h_y) - 2\frac{y_i}{h_y}\Phi(y_i) + \left(1 - \frac{y_i}{h_y}\right) \Phi(y_i - h_y) \\ &\quad + \frac{1}{h_y}\left(\phi(y_i + h_y) - 2\phi(y_i) + \phi(y_i - h_y)\right) \end{aligned}$$

$i = 1, 2, ..., 2m_y - 1$

$$\begin{aligned} w_{2m_y} = w_0 &= \int_{y_0}^{y_0+h_y} \left(1 - \frac{|y_i - y|}{h_y}\right) \frac{e^{-y^2/2}}{\sqrt{2\pi}} dy + \int_{-\infty}^{y_0} \frac{e^{-y^2/2}}{\sqrt{2\pi}} dy \\ &= \left(1 + \frac{y_0}{h_y}\right) \Phi(y_0 + h_y) - \frac{y_0}{h_y}\Phi(y_0) + \frac{1}{h_y}\left(\phi(y_0 + h_y) - \phi(y_0)\right) \end{aligned}$$

Where $\Phi(x)$ and $\phi(x)$ are the normal CDF and PDF respectively. These weights are then normalized by first calculating the weighted moments of y

$$M_0 = \sum_{i=0}^{2m_y} w_i, \quad M_2 = \sum_{i=0}^{2m_y} w_i y_i^2, \quad M_4 = \sum_{i=0}^{2m_y} w_i y_i^4$$

and defining

$$A := \frac{M_4 - M_2}{M_0 M_4 - M_2^2}, \quad B := -\frac{M_2 - M_0}{M_0 M_4 - M_2^2}.$$

Then we define the normalized weights as $w_i^{norm} := (A + By_i^2)w_i$ [1]. This ensures that

$$\sum_{i=0}^{2m_y} w_i^{norm} = 1$$

The weighted sum to approximate the integral is then

$$\begin{aligned} V_{j-1,k}^{+} &= \sum_{i=0}^{2m_y} w_i^{norm} V_{j,k'(i)}, j = 2, 3, ..., J \\ k'(i) &= \frac{h_{j-1}(k - m_z) + y_i\sqrt{\xi_j - \xi_{j-1}}}{h_j} + m_z \end{aligned}$$

Here $k'(i)$ is not an integer, so we will have to linearly interpolate over the values $V_{j,k}$ at this point in each iteration, for each k. At $j = 1$ we can calculate the price by

$$\begin{aligned} V_{0,0} &= \sum_{i=0}^{2m_y} w_i^{norm} V_{1,k'(i)} \\ k'(i) &= \frac{y_i\sqrt{\xi_1}}{h_1} + m_z \quad [1]. \end{aligned}$$

3.1.4 Bermudan Payoff Function

A callable Bermudan Swaption gives the owner the right but not the obligation to exit from the swap at a time of their choosing from a set of predetermined exercise dates. So payments are received and paid until the owner exercises the option, at which point no more payments are made. As we mentioned previously we can take advantage of a parity between "Exercise into" and callable Swaptions. Accordingly, we can price this as an "Exercise into" Swaption and subtract this from the price of the underlying swap to find the price of the Callable Swaption.

The fixed and floating legs respectively pay amounts

$$\begin{aligned} &N\tau(S_{j-1}, S_j)R^{fix} \text{ at times } S_j, j = 1, 2, ..., m \\ &N\tau(T_{i-1}, T_i)R_i^{float} \text{ at times } T_i, i = 1, 2, ..., n \end{aligned}$$

where N is the notional amount, and R^{fix} and R^{float} are the fixed and floating rates, the notation is otherwise the same as Section 1.1.2. Typically the floating rate payment is the LIBOR rate. In [1] the floating rate would be presented as a margin plus a basis spread. This basis spread is the difference between the forward and the current spot rates, over the unobserved period. However, Hagan's paper was written in 2004, and only used a single curve to price these products, but Lichters' book [12] claims that we generally now require multiple curves in order to price these kinds of products, one for discounting and one for forward projection of the rates. Subsequently, we will replace the payoff formulation in [1] with a more relevant formulation similar to [12] and [18].

Pricing the "Exercise Into" part of the Swaption, the payoff can be considered as the value of the Swap at the given exercise time. So we proceed by pricing the floating and fixed legs separately. The price of the floating leg at a future date depends on the forward rate, we will denote the relevant yield curve as fwd, and the discount curve as $disc$. Then the floating leg is valued at exercise time t_j^{ex} as

$$\sum_{i=j_{first}}^{n} \left(\tilde{P}^{disc}(t_j^{ex}, T_i) N \left(\left(\frac{\tilde{P}^{fwd}(t_j^{ex}, T_{i-1})}{\tilde{P}^{fwd}(t_j^{ex}, T_i)} - 1 \right) \left(\frac{1}{\tau(T_{i-1}, T_i)} \right) + s \right) \right) \tau(T_{i-1}, T_i)$$

Where $T_{j_{first}}$ is the time of the first floating rate payment after t_j^{ex}, and s is a constant spread that is applied in practice. We can similarly price the fixed leg of the swap as

$$\sum_{i=j_{first}}^{m} \tilde{P}^{disc}(t_j^{ex}, S_i) \tau(S_{i-1}, S_i) N R_{fix}$$

Again $S_{j_{first}}$ is the first fixed payment after the exercise date. We have used the formulation of the Zero-Coupon Bond divided by the numeraire $\tilde{P}(t, T)$. Implementing this in our Rollback method allows us to price a Bermudan Swaption in the single currency case.

The value of the underlying swap at any exercise date receiving the fixed leg is the fixed leg minus the floating leg evaluated at the exercise date. For a Callable Swaption receiving the fixed leg, we price the underlying swap receiving the fixed leg and the "Exercise Into" option for a swap paying the fixed leg, then the value of the Callable Swaption is the former minus the price of entering the latter. We do the reverse for a Callable Swaption paying the fixed leg. We now present the algorithm for the single currency Rollback method.

Algorithm 2: RollBack Method

Set up grid z_k, t_j for $k \in \{0, ..., 2m_z\}$, $j \in \{0, ..., J\}$
$P_{J,k} = P(t_J^{ex}, z_k^J)$
$V_{J,k} = \max\{P_{J,k}, 0\}$
for $j = 1, ..., J$ **do**
  for $k = 0, 1, ..., 2m_z$ **do**
    for $i = 1, 2, ..., 2m_y - 1$ **do**
      $y_i = h_y(i - m_y)$
      $w_i = \left(1 + \frac{y_i}{h_y}\right)\Phi(y_i + h_y) - 2\frac{y_i}{h_y}\Phi(y_i) + \left(1 - \frac{y_i}{h_y}\right)\Phi(y_i - h_y)$
      $+\left(\frac{1}{h_y}\right)(\phi(y_i + h_y) - 2\phi(y_i) + \phi(y_i - h_y))$
    $w_0 = w_{2m_y} = \left(1 + \frac{y_0}{h_y}\right)\Phi(y_0 + h_y) - \frac{y_0}{h_y}\Phi(y_0) + \frac{1}{h_y}(\phi(y_0 + h_y) - \phi(y_0))$
    $M_0 = \sum_{i=0}^{2m_y} w_i$
    $M_2 = \sum_{i=0}^{2m_y} w_i y_i^2$
    $M_4 = \sum_{i=0}^{2m_y} w_i y_i^4$
    $A = \frac{M_4 - M_2}{M_0 M_4 - M_2^2}$
    $B = -\frac{M_2 - M_0}{M_0 M_4 - M_2^2}$
    $w_i^{norm} = (A + By_i^2)w_i$ for $i = 1, 2, ..., 2m_y$
    if $j = 1$ **then**
      Calculate the vector $k'(i) = \frac{y_i}{\sqrt{\xi_1} h_1} + m_z$ for $i = 0, 1, ..., 2m_y$
    else
      Calculate the vector $k'(i) = \frac{h_{j-1}(k - m_z) + y_i\sqrt{\xi_j - \xi_{j-1}}}{h_j} + m_z$ for $i = 0, 1, ..., 2m_y$
    LinearInterpolate$(V_{j,k'(i)})$ for $i = 0, 1, ..., 2m_y$
    $V_{j-1,k}^+ = \sum_{i=0}^{2m_y} w_i^{norm} V_{j,k'(i)}$
    $P_{j-1,k} = P(t_{j-1}^{ex}, z_k^{j-1})$
    $V_{j-1,k} = \max\{P_{j-1,k}, V_{j-1,k}^+\}$
return $V_{0,0}$

[1]

3.2 Cross-Currency Rollback on Bermudan Swaptions

The extension of this method to the cross-currency case has not been studied previously. The effectiveness of this method hinges on the analytical tractability of the payoff function for the Cross-Currency Bermudan Swaption.

We will first derive the payoff function, reusing much of the notation from [1], and then discuss whether we can employ a similar method to what was used in [1], which we just described in Section 3.1.

3.2.1 Payoff Function

As we mentioned previously the Callable Swaption can be priced using an "Exercise Into" Swaption for a Swap in the opposite position and the actual underlying Swap at that time. We will consider the viewpoint of the option holder receiving the foreign amount and paying the domestic amount, and focus on pricing the "Exercise Into" Swaption, similar to [1]. At the exercise date t_j^{ex} the option holder will have the opportunity to enter the Swap contract. Here we focus on the simple cases where the Swaptions are not Marked to Market, so we do not need to consider Notional Resets. In this section we adapt and combine much of the notation and formulations from [1], [3], [12], and [18]. As we discussed in Section 1.2 if they choose to exercise, this involves exchanging a notional amount at the start of the Swap. So we would receive $\hat{N}_f$, the foreign notional amount, and pay $\hat{N}_d$, the domestic notional amount. Since we work in the LGM measure, we must divide

everything by the domestic LGM numeraire $N_d(t)$. So the initial value of the notional exchange is

$$\frac{\hat{N}_f x(t_j^{ex}) - \hat{N}_d}{N_d(t_j^{ex})}. \tag{3.2.1}$$

Here $x(t)$ is the foreign exchange rate process, we also discussed in Section 1.2 how we choose a single currency to value the product, accordingly we convert everything to the domestic currency as needed.

We value the foreign and domestic legs separately again. The value of a coupon at any time T_k is calculated similarly to before. Then the leg corresponding to currency i after exercising at t_j^{ex} is

$$PV(Cpn, \Omega_i = Float, s_i; t_j^{ex}) =$$
$$= \sum_{k=j_{i,first}}^{n_i} \hat{N}_i \tilde{P}_i^{disc}(t_j^{ex}, T_k^i, z_i(t_j^{ex})) \left(\left(\frac{\tilde{P}_i^{fwd}(t_j^{ex}, T_{k-1}^i, z_i(t_j^{ex}))}{\tilde{P}_i^{fwd}(t_j^{ex}, T_k^i, z_i(t_j^{ex}))} - 1 \right) \frac{1}{\tau(T_{k-1}^i, T_k^i)} + s_i \right) \tau(T_{k-1}^i, T_k^i) \tag{3.2.2}$$

$\tilde{P}_i^{disc}(t, T, z_i(t))$ and $\tilde{P}_i^{fwd}(t, T, z_i(t))$ are the stochastic Zero Coupon Bond prices from t to T based on the discount and forward rate curves respectively for currency i. These are of course also divided by the currency i numeraire, where $i = \{f, d\}$ for foreign and domestic. We also have a constant spread s_i. We attach index i to the payment dates T_k^i to highlight that payment dates for the foreign and domestic legs are not necessarily the same. Accounting for the change of currency and numeraire on the foreign leg, the value of the Swap when exercised at t_j^{ex} is

$$-PV(Cpn, \Omega_{FOR} = Float, s_{FOR}; t_j^{ex}) \frac{N_f(t_j^{ex})}{N_d(t_j^{ex})} x(t_j^{ex}) + PV(Cpn, \Omega_{DOM} = Float, s_{DOM}; t_j^{ex})$$

Either or both of the legs could also be based on a fixed rate in which case the value of the coupons is

$$PV(Cpn, \Omega_i = Fixed, r_i; t_j^{ex}) = \sum_{k=j_{i,first}}^{n_i} \hat{N}_i \tilde{P}_i^{disc}(t_j^{ex}, T_k^i, z_i(t_j^{ex})) r_i \tau(T_{k-1}^i, T_k^i). \tag{3.2.3}$$

At this stage, we would like to highlight the dependence on all three processes z_d, z_f, and $\ln(x)$. The only piece remaining is the final notional exchange, where the initial endowments are essentially returned. So the value of this discounted back from final time T_{n_i} to the exercise date is given as

$$-\hat{N}_f \frac{\tilde{P}_f^{disc}(t_j^{ex}, T_{n_f}) N_f(t_j^{ex})}{N_d(t_j^{ex})} x(t_j^{ex}) + \hat{N}_d \tilde{P}_d^{disc}(t_j^{ex}, T_{n_d}) \tag{3.2.4}$$

Alternatively using our definitions from Section 1.2

$$PV(Exch, \Omega_{DOM}; t_j^{ex}) = \hat{N}_d \tilde{P}_d^{disc}(t_j^{ex}, T_{n_d}) - \frac{\hat{N}_d}{N_d(t_j^{ex})}$$
$$PV(Exch, \Omega_{FOR}; t_j^{ex}) = -\hat{N}_f \frac{\tilde{P}_f^{disc}(t_j^{ex}, T_{n_f}) N_f(t_j^{ex})}{N_d(t_j^{ex})} x(t_j^{ex}) + \frac{\hat{N}_f x(t_j^{ex})}{N_d(t_j^{ex})}$$

It should be noted that all of these expressions are also divided by the domestic numeraire. So their sum is the reduced value of the payoff for an "Exercise Into" Swaption receiving the domestic leg. For a Swaption based on receiving the foreign leg, we simply take the negative of each part.

3.2.2 Adapting the Rollback Method

Previously the Rollback algorithm depended on a single stochastic process and required two grids for the process z, and for the random Gaussian increment between time steps y when approximating the conditional expectation. Now we have three random processes, so we require grids for at least z_d, z_f, and $\ln(x)$.

The first step was to calculate the final payoff and determine the value of the Swaption one step prior using a conditional expectation and the General Pricing Formula from 2.1.4. So we

are interested in approximating the same conditional expectation we saw in the single currency case, but now with the cross currency model, so letting $P(T_i, x, z_f, z_d)$ be the new payoff function defined above we are interested in calculating

$$\begin{aligned}\mathbb{E}[P(T_i, z_d(T_i), z_f(T_i), \ln(x(T_i)))|\mathcal{F}_{T_{i-1}}] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{1}{\pi^{3/2}} \det(\Delta\Sigma_i)^{-1/2} \cdot \\ \cdot \exp\left(-\frac{1}{2}(\Delta\vec{z_i} - \Delta\vec{\mu_i})^T(\Delta\Sigma_i)^{-1}(\Delta\vec{z_i} - \Delta\vec{\mu_i})\right) \cdot \\ \cdot P(T_i, z_d(T_i), z_f(T_i), \ln(x(T_i)))d\Delta\vec{z_i}\end{aligned}$$

Since each of z_d, z_f, and $\ln(x)$ are normal, their joint distribution is multivariate normal with corresponding mean vector $\Delta\vec{\mu}$ and covariance matrix $\Delta\Sigma$ for an increment of the process. We now denote the vector of z_f, z_d, and $\ln(x)$ as $\vec{z}$. We also integrate with respect to $\Delta\vec{z_i}$, as this is the vector of stochastic increments between T_{i-1} and T_i. The Rollback method depends on these increments, so we will consider the means and covariances in that context going forward. These will have a slightly modified representation from what we saw in Section 2.3, but only minor adjustments are required to calculate them using the expressions there.

Recall that in the single currency case, this expectation had to be calculated at every grid point of z. Here this would now need to be calculated for every combination of grid points for z_d, z_f, and $\ln(x)$. This is already computationally quite slow and involved. Previously in order to apply the quadrature method we considered $z_d(T_i)$ conditioned on the information at T_{i-1} and re-wrote it as $z_d(T_{i-1}) + Y\sqrt{\xi(T_i) - \xi(T_{i-1})}$ where Y is Gaussian.

In order repeat this here we must account for the new correlation between each process. So we start by finding the Cholesky decomposition of the covariance matrix such that $\Sigma = A^T A$. Then we can generate three independent standard Gaussians $\vec{Y} = (Y_d, Y_f, Y_x)$ such that $\vec{\mu} + A\vec{Y}$ has the same dependence structure as an increment between T_{i-1} and T_i of z_d, z_f and $\ln(x)$ [30, Section 1.8]. Then we can write the conditional expectation as follows

$$\begin{aligned}\mathbb{E}[P(T_i, x, z_f, z_d)|\mathcal{F}_{T_{i-1}}] = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \frac{1}{\pi^{3/2}} \det(\mathbb{I}_3)^{-1/2} \exp\left(-\frac{1}{2}(\vec{Y})^T\mathbb{I}_3^{-1}(\vec{Y})\right) \cdot \\ \cdot P(T_i, z_d(T_{i-1}) + (\Delta\vec{\mu} + \vec{Y}A)_d, z_f(T_{i-1}) + (\Delta\vec{\mu} + \vec{Y}A)_f, \ln(x(T_{i-1})) + (\Delta\vec{\mu} + \vec{Y}A)_x)d\vec{Y}\end{aligned}$$

Following the steps for the single currency case we would require separate grids for Y_d, Y_f and Y_x. Meaning in total we require six grids, for each random process $z_d, z_f, \ln(x), Y_d, Y_f$, and Y_x. Algorithm 2 requires several intermediate calculations at each iteration such as calculating moments, weights, and performing linear interpolation. Applying this to the three-dimensional case we are now working with would lead to a very computationally inefficient algorithm.

Unfortunately due to the time-sensitive nature of this thesis, and the requirements from modern financial institutions for fast and efficient algorithms, we will not be able to proceed any further with implementing or testing this procedure. We will instead turn our focus to a Monte-Carlo based approach.

3.3 A Potential Solution

We would also like to discuss some potential methods for overcoming this problem that would have required more research than we realistically had time for.

PDE methods for this type of problem have been previously considered as we mentioned at the beginning of this paper. They have generally suffered the same fate we witnessed here, of suffering from significant increases in computational complexity with increases in dimensionality. Although one methodology, that has been researched in the general area of solving PDEs, may be of use in these high dimensional cases and in the context of the problem we present above.

The method we are referring to is the use of Cubatures on Wiener space [31]. We will briefly outline the method in [31] and explain how it can be applied. In the PDE context this is a unique method of approximating the solution of parabolic PDE's of arbitrary dimensions. Essentially, for a PDE with differential operator $L = V_0 + \frac{1}{2}(V_1^2 + ... + V_d^2)$ where V_i are vector spaces of $\mathbb{R}^N$, with

the following dynamics

$$\frac{\partial u}{\partial x} = -Lu(t,x)$$
$$u(T,x) = f(x).$$

A stochastic process can be used to approximate a solution. We first define the Wiener space $C_0^0([0,T],\mathbb{R}^d)$ as the set of continuous functions on $[0,T]$ that are $\mathbb{R}^d$ valued and start at 0. Then defining $\mathcal{F}$ and $\mathbb{P}$ as the necessary Borel σ-field and probability measure respectively we have a probability space $\{C_0^0([0,T],\mathbb{R}^d),\mathcal{F},\mathbb{P}\}$.

So given a vector of Brownian Motions $B = (B_t^1,...,B_t^d)$ we define the stochastic process

$$d\xi_{t,x} = \sum_{i=0}^{d} V_i(\xi_{t,x}) \circ dB_t^i.$$

Then defining the mapping

$$\Phi_{T,x} : C_0^0([0,T],\mathbb{R}^d) \mapsto \mathbb{R}^N, \quad \omega \mapsto \xi_{t,x}(\omega)$$

we can rewrite the solution of the PDE as

$$u(0,x) = \int_\Omega f(\Phi_{T,x}(\omega))\mathbb{P}(d\omega) \tag{3.3.1}$$

[31].

In our case, this Wiener integral is what we are trying to solve or approximate. More specifically the function $f(\Phi_{T,x}(\omega))$ will be our payoff function, ω is the set of Brownian Motions W_d, W_f, and W_x. The method outlined in [31] is specifically to approximate these types of integrals.

It is described in [31] how to approximate this integral using cubatures. First, the cubatures are defined below.

Definition 3.3.1 (Cubature)**.** For a positive measure μ on $\mathbb{R}^d$, and $x_1, x_2, ..., x_n$ in the support of μ. If we also consider the set of polynomials $P \in \mathbb{R}_m[X_1,...,X_d]$ and a set of positive weights $\lambda_1, \lambda_2, ...\lambda_n$ we can then define the Cubature as

$$\int_{\mathbb{R}^d} P(x)\mu(x)dx = \sum_{i=1}^{n} \lambda_i P(x_i)$$

[31].

When $d = 1$ this is equivalent to the quadrature formula which was used in the Rollback method, so we suspect this could be a natural extension to solving our problem. The following Theorem is also defined to support the existence of a cubature approximation.

Theorem 3.3.2. *With μ and P defined as before. If $\int |P(x)|\mu(dx) < \infty$ for all $P \in \mathbb{R}_m[X_1, X_2, ..., X_d]$ then there exists points $x_1, ..., x_n$ and weights $\lambda_1, ..., \lambda_n$ where $n \leq \dim \mathbb{R}_m[X_1, ..., X_d]$ such that Definition 3.3.1 holds. [31].*

We now define the stochastic Taylor expansion

Definition 3.3.3 (Stochastic Taylor Expansion)**.** For a smooth function f, and stochastic process ξ we have that

$$f(\xi_{t,x}) = f(x) + \int_0^t \sum_{i=0}^{d} V_i f(\xi_{s,x}) \circ dB^i(s).$$

Defining also

$$\mathcal{A}_m := \{(i_1,...,i_k) \in \{0,...,d\}^k, k + \texttt{card}\{j, i_j = 0\} \leq m\}$$

based on the fact that

$$\int_{0<t_1<...<t_k<t} \circ dB_{t_1}^{i_1} ... \circ dB_{t_k}^{i_k} \overset{\mathbb{P}}{=} \sqrt{t}^{k+\texttt{card}\{j,i_j=0\}} \int_{0<t_1<...<t_k<1} \circ dB_{t_1}^{i_1} ... \circ dB_{t_k}^{i_k}.$$

Then for a smooth and bounded function f

$$f(\xi_{t,x}) = \sum_{(i_1,...,i_k)\in A_m} V_{i_1}...V_{i_k}f(x)\int_{0<t_1<...<t_k<t} \circ dB^{i_1}_{t_1}... \circ dB^{i_k}_{t_k} + R_m(t,x,f).$$

Where R_m is the remainder and for a constant C depending on d and m is subject to

$$\sup_{x\in\mathbb{R}^N}\sqrt{\mathbb{E}(R_m(t,x,f)^2)} \leq Ct^{(m+1)/2}\sup_{(i_1,...,i_k)\in\mathcal{A}_{m+2}\setminus\mathcal{A}_m}||V_{i_1}...V_{i_k}f||_\infty$$

[31].

In [31] a Cubature formula is defined by a set of paths $\omega_1, ..., \omega_n$ on the Wiener space of degree m, and a set of weights $\lambda_1, ..., \lambda_n$. Defining a new probability measure $\mathbb{Q}_T$

$$\mathbb{Q}_T = \sum_{i=1}^{n}\lambda_i\delta_{\omega_{T,i}}$$

From this a new probability measure $\tilde{\mathbb{Q}}$ is constructed on the Wiener space such that the expectation of $f(\xi_{T-t,x})$ under $\mathbb{Q}$ and $\tilde{\mathbb{Q}}$ are very close. Doing so allows us to convert the problem of solving the associated PDE into one of solving a number of ODEs. Taking a weighted average of their solutions approximates the solution of the PDE and our relevant stochastic integral.

it is proved in [31] that

$$\sup_{x\in\mathbb{R}^n}\left|\mathbb{E}_{\mathbb{Q}_T}\left(f(\xi_{T,x}) - \sum_{i=1}^{n}\lambda_i f(\Phi_{T,x}(\omega_{T,i}))\right)\right| \tag{3.3.2}$$

is bounded above depending on T, but steps can be taken to minimize this error or bound.

Where $\omega_{T,i} : [0,T] \mapsto \mathbb{R}^{d+1}$ is a specific reweighting of existing paths $\omega_1, ..., \omega_n$ and $\delta_{\omega_{T,i}}$ is the increment along the path. By treating the integral in (3.3.1) as an expectation we approximate it by the relation in (3.3.2). The difference between this expectation and the same expectation under the original measure is proved to be bounded above as well in [31].

From here we can more easily define the random variables $(Y_i)_{0\leq i\leq k}$ as the random variables satisfying

$$\mathbb{P}(Y_{i+1} = \Phi_{(t_i-t_{i-1}),x}(\omega_{(t_i-t_{i-1}),i})|Y_i = x) = \lambda_i$$

Then it is stated in [31, Page 175] that

$$\mathbb{E}(f(Y_k)|Y_0 = x) = \sum_{i_1=1}^{n}...\sum_{i_k=1}^{n}\lambda_{i_1}...\lambda_{i_k}f(\Phi_{T,x}(\omega_{s_1,i_1}\otimes...\otimes\omega_{s_k,i_k})) = \mathbb{E}_{Q_T^k}(f(\xi_{T,x}))$$

At this point this also more closely resembles the problem we face of determining the conditional expectation in the Rollback method.

Implementation of this can be represented as a n-ary tree, where the sum of various nodes representing certain ODEs will approximate the PDE, according to [31].

A great amount of detail has been omitted here, as we simply sought to introduce the idea and consider how it might apply in our case. Again due to the time-sensitive nature of our research, it is very difficult to proceed with methods that are so unfamiliar and uncertain in their effectiveness. At this point, we could not say for certain how complex it would be to implement this methodology, it is even noted in [31] that depending on the size of n and k it may be necessary to use Monte Carlo simulations of $\mathbb{E}_{Q_T^k}(f(\xi_{T,x}))$ anyway. Alongside this, the early exercise feature can cause kinks or discontinuities when it is optimal to exercise early. This is noted in Hagan's paper [1] also in the single currency case, but he also provides steps that can reduce any inaccuracies arising from this . It is currently unclear how this would affect the methodology we described.

Chapter 4

American Monte Carlo

In this section, we will detail the method we finally used to price the Cross-Currency Bermudan Swaption. Monte Carlo is a common approach for pricing these complex non-standardized options due to its tractability. In particular, Monte Carlo is well renowned for being flexible in accommodating different payoff schemes and additional complications such as path dependencies quite well [32]. The simulation of stochastic processes also means that the random nature we observe in real markets can be well represented. It also allows for the calculation of Greeks using methods such as finite difference estimators [33]. Most importantly perhaps, when it is not possible to obtain an analytical expression for the price of the product, as in our case, Monte Carlo is one of the few toolkits that is available to tackle such problems. While this method has been commonly used for these types of products, to the best of our knowledge, it has not been thoroughly assessed on Cross-Currency Bermudan Swaptions with the LGM model. Provided in [12] are some details of the procedure, but little to no assessment of the accuracy or convergence of the method is provided.

We also sought to implement this method for the single currency Bermudan Swaption, this has been researched more extensively than the cross-currency case such as in [34] and [35], but to the best of our knowledge it has not been numerically assessed and compared to Hagan's Rollback [1]. The implementation for the single currency case is almost identical once the payoff function has been adjusted.

4.1 Monte Carlo

We will first give a general outline of the standard Monte Carlo method and principles. We start by noting that the methodology has a strong reliance on the Law of Large numbers.

Theorem 4.1.1 (Law of Large Numbers)**.** *Consider a sample of n independent, identically distributed random variables $X_1, X_2, ..., X_n$ with mean μ and variance σ^2. The sample mean is defined*

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_i$$

Then for any $\epsilon > 0$

$$\lim_{n\to\infty} \mathbb{P}(|\bar{X}_n - \mu| > \epsilon) = 0$$

[36].

The Monte Carlo method is most widely used for the estimation of statistical quantities such as the mean and variance of random variables. Modern computing allows for the fast simulation of a large number of samples of a random variable, calculating the sample average of these simulations provides an accurate estimation of the mean due to the Law of Large Numbers [37]. In our case, the quantity of interest is of course the price of the Cross-Currency Bermudan Swaption. Driven by a number of underlying processes, we can simulate the price as a function of these random variables and use Monte Carlo to calculate the mean of this function giving us an estimate of the price.

If we wish to calculate a confidence interval for our estimate we rely on the Central Limit Theorem.

Theorem 4.1.2 (Central Limit Theorem)**.** *Given a random sample $X_1, X_2, ..., X_n$ from some distribution with mean μ and standard deviation σ. As $n \mapsto \infty$, $\bar{X}$ has a normal distribution.*

In other words

$$\lim_{n\mapsto\infty} \mathbb{P}\left(\frac{\bar{X}-\mu}{\sigma/\sqrt{n}} \leq z\right) = \Phi(z)$$

where Φ is the CDF of the normal distribution [38].

By first calculating the sample variance estimate $s_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \bar{X}_n)^2$, for a given confidence level α the confidence interval is

$$\bar{X}_n \pm \frac{Z_\alpha s_n}{\sqrt{n}}$$

where Z_α is such that $\mathbb{P}(-Z_\alpha \leq Z \leq Z_\alpha) = 1 - \alpha$ [39].

4.2 Path Simulation

In order to price a product using Monte Carlo we must first run a large number of simulations of the underlying processes or assets that drive the price of the product. In our case we will be simulating the LGM processes z_d, z_f and $\ln(x)$ that drive the Cross-Currency Bermudan Swaptions' price. One distinct advantage of the LGM model in this case is the ability to simulate exact trajectories of these processes. We have already seen the closed forms or distributions of each in Section 2.3. Each process is Gaussian due to the structure of our Itô integrals, as we discussed in Section 2.1.3, and so can be simulated using randomly generated Gaussian random variables.

We first calculate the mean and variance of each process for the time increment $[s, t]$, so using the expressions in Section 2.3, define the mean vector

$$\vec{\mu} = \begin{pmatrix} \mathbb{E}[z_d(t) - z_d(s)] \\ \mathbb{E}[z_f(t) - z_f(s)] \\ \mathbb{E}[\ln(x(t)) - \ln(x(s))] \end{pmatrix}.$$

We also calculate a vector for the individual variances $\vec{\mathbb{V}}$

$$\vec{\mathbb{V}} = \begin{pmatrix} \mathbb{V}[z_d(t) - z_d(s)] \\ \mathbb{V}[z_f(t) - z_f(s)] \\ \mathbb{V}[\ln(x(t)) - \ln(x(s))] \end{pmatrix}.$$

Calculate the correlation matrix Σ of the increments of three processes using the covariance and variance expressions we saw previously

$$\Sigma = \begin{pmatrix} 1 & Corr[\Delta z_d, \Delta z_f] & Corr[\Delta z_d, \Delta \ln(x)] \\ Corr[\Delta z_d, \Delta z_f] & 1 & Corr[\Delta \ln(x), \Delta z_f] \\ Corr[\Delta z_d, \Delta \ln(x)] & Corr[\Delta \ln(x), \Delta z_f] & 1 \end{pmatrix}$$

we then find the Cholesky decomposition A of this matrix by solving

$$\Sigma = A \cdot A^T.$$

Finally, we can simulate all three processes at once by simulating three independent standard Normal random variables $\vec{N}$ and using the following relation

$$\begin{pmatrix} z_d(t) - z_d(s) \\ z_f(t) - z_f(s) \\ \ln(x(t)) - \ln(x(s)) \end{pmatrix} = \vec{\mu} + \vec{\mathbb{V}} A^T \vec{N}$$

This process of simulating correlated normal random variables from independent standard normal random variables is explained in [30, Section 1.8], the simulation specifics for the LGM model are also briefly formulated in [12]. This generates a single increment of the three processes between time points s and t, by simulating a sequence of increments and cumulatively adding them, we can simulate an entire path.

We can actually observe the martingale property of the quantities

$$\tilde{P}(t,T,z_d(t)) = \frac{P(t,T,z_d(t))}{N_d(t)} = P(0,T)\exp\left(-H_d^2(T)z_d(t) - \frac{1}{2}H_d^2(T)\xi_d(T)\right)$$
$$Y(t) = \frac{N_f(t)x(t)}{N_d(t)}$$
$$U(t) = \frac{P_f(t,T)x(t)}{N_d(t)}$$

which we mentioned in Section 2.3, we demonstrate this in Section 5.2.1 of our results.

In our simulations we assume the mean reversion parameters λ_f and λ_d are constant, and that the volatilities $\sigma_f(s), \sigma_x(s)$, and $\sigma_d(s)$ are piecewise constant functions. The implementation of all the relevant mean, variance and covariance calculations is actually quite involved, often requiring piecewise functions that are not so easily generalized. Alongside this many of the integrals to be calculated are quite similar but have slight variations making the implementation extremely error-prone. Fortunately, it is possible to check each of these calculations using numerical integration.

4.3 Longstaff-Schwartz

Just like Hagan's Rollback, the Longstaff-Schwartz method [2] is also a backwards induction method and follows a similar structure at a high level. Once again we focus on pricing an "Exercise into" Bermudan Swaption, as this can be used to also evaluate the Callable Bermudan Swaption using the parity relation we described in Section 1.2.3. We follow the algorithm used in [2] and present the formulations relevant to our problem below.

As we did with the Rollback method we start by calculating the final payoff, we use the same payoff function as in equation (3.2.2), along with the notional exchanges (3.2.1) and (3.2.4). Where we can substitute our simulated values of z_d, z_f and $\ln(x)$. Here we make two slight alterations first we treat the initial notional exchange as a strike price, and remove it from the payoff function. Secondly, we consider the version of the payoff that is not yet divided by the domestic numeraire. Then the payoff $\frac{P(T,z_d,z_f,\ln(x))-K}{N_d(T,z_d)}$ is equal to the previously defined payoff, where K is the new strike. We assume another time grid covering only the exercise dates $\{t_1^{ex}, t_2^{ex}, ..., t_J^{ex}\}$. So allowing $V(t_J^{ex}, \vec{z})$ to be the value of the contract at final exercise time t_J^{ex} for simulated values $\vec{z}$, we begin by setting

$$V(t_J^{ex}, \vec{z}(t_J^{ex})) = \max(P(t_J^{ex}, z_d(t_J^{ex}), z_f(t_J^{ex}), \ln(x(t_J^{ex}))) - K, 0).$$

Where clearly the investor will not exercise the option if the payoff is negative. Then we can perform the first backwards iteration step where we aim to calculate $V(t_{J-1}^{ex}, \vec{z}(t_{J-1}^{ex}))$. We follow as before taking advantage of the General Pricing Formula, in Theorem 2.1.4, in order to determine the value of continuing the contract if we do not exercise, denoting the continuation value as $C(t_{J-1}^{ex}, \vec{z}(t_{J-1}^{ex}))$

$$C(t_{J-1}^{ex}, \vec{z}(t_{J-1}^{ex})) = \mathbb{E}\left[V(t_J^{ex}, \vec{z}(t_J^{ex}))\frac{N_d(t_{J-1}^{ex}, z_d(t_{J-1}^{ex}))}{N_d(t_J^{ex}, z_d(t_J^{ex}))}\middle|\mathcal{F}_{J-1}\right]$$

Then since the previous time point t_{J-1}^{ex} is also an exercise date, the investor must decide whether to exercise and receive the immediate payoff or wait until the next date. Subsequently, the value of the contract is then

$$V(t_{J-1}^{ex}, \vec{z}(t_{J-1}^{ex})) = \max\{C(t_{j-1}^{ex}, \vec{z}(t_{J-1}^{ex})), P(t_{J-1}^{ex}, \vec{z}(t_{J-1}^{ex})) - K\}$$

When it is optimal to exercise early the value of the immediate payoff will exceed the value of staying in the contract. When it is not optimal to exercise early the continuation value of the contract will exceed the immediate payoff. The decision is then the same at each exercise date, so we continue iterating backwards in this way until reaching the final time, where $V(0, \vec{z}(0))$ is the value of the contract. Taking the average of $V(0, \vec{z}(0))$ over every path provides the Monte Carlo estimate of the price [2].

To see why this is the price, consider the algorithm in each case where it is and is not optimal

to exercise. If it is optimal to exercise we rely again on Theorem 2.1.4, in particular, we have that

$$\begin{aligned}\mathbb{E}\left[V(t_i^{ex},\vec{z}(t_i^{ex}))\frac{N_d(t_{i-1}^{ex},z_d(t_{i-1}^{ex}))}{N_d(t_i^{ex},z_d(t_i^{ex}))}\middle|\mathcal{F}_{i-1}\right] &= \mathbb{E}\left[P(t_i^{ex},\vec{z}(t_i^{ex}))\frac{N_d(t_{i-1}^{ex},z_d(t_{i-1}^{ex}))}{N_d(t_i^{ex},z_d(t_i^{ex}))}\middle|\mathcal{F}_{i-1}\right]\\ &= V(t_{i-1}^{ex},\vec{z}(t_{i-1}^{ex}))\end{aligned}$$

As, if we exercise at this point, the value at the previous time is simply the discounted value of the payoff. If instead we encounter that it is optimal to wait rather than exercise early we have that

$$\begin{aligned}\mathbb{E}\left[V(t_i^{ex},\vec{z}(t_i^{ex}))\frac{N_d(t_{i-1}^{ex},z_d(t_{i-1}^{ex}))}{N_d(t_i^{ex},z_d(t_i^{ex}))}\middle|\mathcal{F}_{i-1}\right] &= \mathbb{E}\left[C(t_i^{ex},\vec{z}(t_i^{ex}))\frac{N_d(t_{i-1}^{ex},z_d(t_{i-1}^{ex}))}{N_d(t_i^{ex},z_d(t_i^{ex}))}\middle|\mathcal{F}_{i-1}\right]\\ &= \mathbb{E}\left[\mathbb{E}\left[V(t_{i+1}^{ex},\vec{z}(t_{i+1}^{ex}))\frac{N_d(t_{i-1}^{ex},z_d(t_{i-1}^{ex}))}{N_d(t_{i+1}^{ex},z_d(t_{i+1}^{ex}))}\middle|\mathcal{F}_i\right]\middle|\mathcal{F}_{i-1}\right]\\ &= V\left(t_{i-1}^{ex},\vec{z}(t_{i-1}^{ex})\right)\end{aligned}$$

In this case, this holds by simple repeated application of the tower property for conditional expectations, since $C(t_i^{ex},\vec{z}(t_i^{ex}))$ is itself a conditional expectation.

By ensuring we maximize the value of the option at each exercise date, we guarantee that we receive the maximal price of the option once the algorithm is complete. We know this to be the actual price of the option, or at least an approximation in this case, as any rational investor will make decisions to maximize their profit, leading to the same decision criteria at each exercise date [2].

4.3.1 Regression

Aside from the simulations of the correlated processes, one of the distinguishing features of this method from Hagan's Rollback is the use of regression in order to calculate the continuation value. The functional form of $\mathbb{E}\left[V(t_i^{ex},\vec{z}(t_i^{ex}))|\mathcal{F}_{i-1}\right]$ is estimated using regression in [2].

There is a certain amount of freedom in the choice of basis functions. In the Longstaff-Schwartz paper [2] it is observed that basic polynomials, Laguerre, Hermite, Legendre polynomials and many more provide accurate results. However, it should be noted that as we increase the number of basis functions the number of simulations must grow exponentially also in order to avoid overfitting according to [40].

Typically $V(t_i^{ex},\vec{z}(t_i^{ex}))$ is regressed on the underlying Swap value at the previous time point or the nearest dated Swap rate, this is done in both [12] and [41]. In both these cases only polynomials up to the third order are used, though only [12] considered a cross-currency case.

Regressing directly on the value function V directly tends to lead to an upward bias according to [2], due to the fact that the max function is convex.

4.3.2 Optimal Exercise Boundary

The method we described above is actually not identical to the algorithm originally described by Longstaff-Schwartz [2]. Instead of performing the regression step on all available paths, in [2] it is only performed on paths that have a positive cashflow at the given time. In other words, one would set

$$V(t_i^{ex},\vec{z}(t_{i-1}^{ex},\omega)) = \begin{cases}0 \texttt{ if not optimal to exercise or non-positive cashflow}\\ P(t_{i-1}^{ex},\vec{z}(t_{i-1}^{ex},\omega)) \texttt{ if optimal to exercise}\end{cases}$$

corresponding to a single path ω.

Alongside this, an array is also kept to identify optimal exercise times for each path. So once the iteration is complete we have an array of all the cashflows at each time point, and an array identifying the exercise times for every path. By discounting all the cashflows appropriately we can calculate the price of the product. In [2] it is argued that this formulation is computationally more efficient, and provides a better estimate of the continuation value by considering only positively valued cashflows. It also helps to minimize the upward bias that was previously mentioned.

4.3.3 Algorithms

We present below the algorithms for the method we employed, in which we perform the regression on all available paths.

Algorithm 3: American Monte Carlo

Set up grid of exercise times:

$\{t_1^{ex}, t_2^{ex}, ..., t_J^{ex}\}$

Generate paths $\vec{z} = \begin{pmatrix} z_d \\ z_f \\ \ln(x) \end{pmatrix}$:

for $\{j = 1; j \leq J; j{+}{+}\}$ **do**

$\vec{N}$ =GenerateRandomNormal(3)

$$\vec{\mathbb{V}}_j = \begin{pmatrix} \mathbb{V}[z_d(t_j^{ex}) - z_d(t_{j-1}^{ex})] \\ \mathbb{V}[z_f(t_j^{ex}) - z_f(t_{j-1}^{ex})] \\ \mathbb{V}[\ln(x(t_j^{ex})) - \ln(x(t_{j-1}^{ex}))] \end{pmatrix}$$

$$\vec{\mu}_j = \begin{pmatrix} \mathbb{E}[z_d(t_j^{ex}) - z_d(t_{j-1}^{ex})] \\ \mathbb{E}[z_f(t_j^{ex}) - z_f(t_{j-1}^{ex})] \\ \mathbb{E}[\ln(x(t_j^{ex})) - \ln(x(t_{j-1}^{ex}))] \end{pmatrix}$$

$$\Sigma_j = \begin{pmatrix} 1 & Corr[\Delta z_d, \Delta z_f] & Corr[\Delta z_d, \Delta \ln(x)] \\ Corr[\Delta z_d, \Delta z_f] & 1 & Corr[\Delta \ln(x), \Delta z_f] \\ Corr[\Delta z_d, \Delta \ln(x)] & Corr[\Delta \ln(x), \Delta z_f] & 1 \end{pmatrix}$$

A =CholesyDecompostion(Σ_j)

$\Delta \vec{z}_j = \vec{\mu}_j + A\vec{\mathbb{V}}_j^{\,1/2}\vec{N}$

$\vec{z}(t_j^{ex}) = \vec{z}(t_{j-1}^{ex}) + \Delta \vec{z}_j$

Calculate final payoff:

$V(t_J^{ex}, \vec{z}(t_J^{ex})) = \max\{P(t_J^{ex}, \vec{z}(t_J^{ex})) - K_J, 0\}$

Perform backwards Iteration:

for $\{j = J; j > 1; j{-}{-}\}$ **do**

Calculate continuation value:

model=FitRegressionModel$\Big(X = P(t_{j-1}^{ex}, \vec{z}(t_{j-1}^{ex})), Y = V\left(t_j^{ex}, \vec{z}(t_j^{ex})\right) \frac{N_d(t_{j-1}^{ex}, z_d(t_{j-1}^{ex}))}{N_d(t_j^{ex}, z_d(t_j^{ex}))}\Big)$

$C(t_{j-1}^{ex}, \vec{z}(t_{j-1}^{ex}))$ =model$\big(P(t_{j-1}^{ex}, \vec{z}(t_{j-1}^{ex}))\big)$

$V(t_{j-1}^{ex}, \vec{z}(t_{j-1}^{ex})) = \max\left\{P(t_{j-1}^{ex}, \vec{z}(t_{j-1}^{ex})) - K_{j-1}, C(t_{j-1}^{ex}, \vec{z}(t_{j-1}^{ex}))\right\}$

Price=mean$\Big(\frac{V(t_1^{ex}, \vec{z}(t_1^{ex}))}{N(t_1^{ex}, z_d(t_1^{ex}))}\Big)$

[2]

As we mentioned, we want the immediate value of the payoff, but our previous definition from equations (3.2.1), (3.2.2), (3.2.4), and (3.2.3) is implicitly divided by the numeraire, so in order to obtain the appropriate value one can simply multiply by the domestic numeraire at that time. We introduce a time dependency on the strike K as the foreign notional has to be converted to the domestic currency at that time using the FX process, which has a time dependency.

In order to use this method for the single-currency Swaption we replace $\vec{z}$ with just the domestic process z_d, and set the strike to be 0 as there is no initial exchange in a single currency swap. In the single-currency algorithm, there is no correlation structure so these steps can be omitted as well.

Chapter 5

Implementation and Results

We implemented the American Monte Carlo method for pricing both the single-currency and cross-currency Bermudan Swaptions, as well as Hagan's Rollback method in the single-currency case. We based our implementation and results around two existing trades one for the single-currency Bermudan and one for the cross-currency Bermudan, and we will discuss each case separately.

5.1 Single-Currency Bermudan Swaption

5.1.1 Context and Data

We considered an actual contract that began in November 2021, and value it at the twelfth of June 2023, as this is when we began working on our implementation, until the end date in 2031. Since there is a large number of details and parameters we present this in Table 5.1 below. The spread

Table 5.1: Table of Single Currency Bermudan Swaption Trade Details

Detail	**Value**
Notional	50,000,000 Euro
Floating Rate Payment Frequency	3 Months
Fixed Rate Payment Frequency	1 Year
Floating Rate Curve	3-Month-Euribor
Discount Rate Curve	Euro-Short-Term-Rate-OIS
Spread	-0.0033
Fixed Rate	0.00192
Floating Leg Daycount Convention	ACT/360
Fixed Leg Daycount Convention	ACT/365
Frequency of Early Exercise Opportunity	1 Year
Fixed Leg	Receive
Floating Leg	Pay
"Exercise Into" or Callable	"Callable"

is applied to the forward rate from the floating leg, and each exercise date aligns with the fixed payment dates.

We also require parameters for the LGM model, fortunately, this calibration was performed prior to this project using European Swaptions over the same period based on the three-month Euribor rate. We use a constant mean reversion parameter $\lambda = 0.034498$ and a piecewise constant volatility function $\sigma(t)$, specified in the following table.

Table 5.2: Table of Piecewise Volatility Values

Date	$\sigma(t)$
20/11/2023	0.010695
20/11/2024	0.011696
20/11/2025	0.011394
20/11/2026	0.010507
22/11/2027	0.010346
20/11/2028	0.010049
19/11/2029	0.009547
20/11/2030	0.008510

At any given date the value of σ used is the earliest date listed after the given date, we use a flat extrapolation where necessary.

The only other data required for this project was the relevant yield curve data. This was available as a CSV file containing an ID for each curve, maturity dates and corresponding yield values. We then used linear interpolation in order to calculate the yields at the necessary dates. This same data is used outside of this project but typically with a different kind of interpolation which leads to some slight differences in pricing for these Swaptions. Alongside this there are additional notification dates that occur a few days before exercise dates so the necessary trade can be prepared. Different conventions for counting days can lead to minor differences as well. These additional details do not have a significant bearing on the price, but will still lead to some minor discrepancies when we compare to the quoted price. Bearing this in mind we expect these methods to at least converge to a price that is some fixed difference from the true price. Subsequently, we can still make a valid analysis of these methodologies.

The quoted price of the relevant "Exercise Into" Swaption is 329,476 Euro, and the Callable Swaption has a price of -9,285,649 Euro, where the price is negative since the party receiving the fixed leg will likely make a loss, and so they have to be paid to enter the contract. The quoted price for the underlying Swap is -8,956,173 Euro. Since the Callable Swaption is equivalent to holding the underlying Swap and the opposite "Exercise Into" Swaption, we only need these two quantities to price the Callable, as we discussed in Section 1.2.3. Note that since calling the Swaption is equivalent to *entering* the "Exercise Into" we actually *deduct* the price of the "Exercise Into" contract from the Swap, as we would have to pay this price to enter into such a position.

5.1.2 Programming Framework

We used Python to implement both the American Monte Carlo and Rollback methods as described in Sections 4 and 3 respectively. The program was structured by a number of classes. This began with a simple coupon class containing the necessary dates and time interval calculations. This was then inherited by a class for pricing a standard Swap which contained information about the contract details and relevant curves. We then use this class in the implementation of the Rollback method directly for managing all the relevant information and payoff calculations for each leg.

We also use this Swap class for the American Monte Carlo method, but we require an LGM Simulator class to simulate individual paths of the process z in the LGM framework. This involves the generation of necessary random variables, performing the Cholesky decompositions, and simulation of the entire path. This is derived from an LGM Engine class which contains all the necessary mean, and variance calculations for the LGM model. We in fact use the same simulation method for both the cross-currency and single-currency case, but only use the process corresponding to the domestic currency for the single-currency Bermudan. By construction, this is independent of the other two processes so poses no issue in this regard.

5.1.3 Rollback

With the Rollback method, there are no criteria for selecting the parameters $N_z, N_y, \lambda_z, \lambda_y$ presented in Hagan's paper [1], but it is stated that empirically values of λ_z between 16 and 32 and λ_y between 10 and 16 work well, while N_z and N_y are suggested to take values between 4 and 5.5. Fortunately in this case we already know the value of the trade, as it has already been made, so we can find the parameters that work best quite simply, but in a standard real-world context we

would obviously not have this luxury. One would generally have to rely on calibration based on the price of similar products in the past.

We generated results over different values for each grid of z and present them below. We first vary the values of N_z and λ_z and fix $\lambda_y = 16$ and $N_y = 5$ in Figure 5.1.

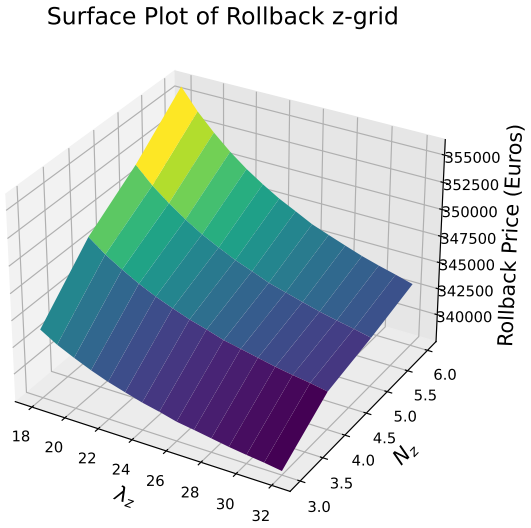

Figure 5.1: Surface Plot of Rollback Values over z-grid

Based on the quoted price, we would suspect that $N_z = 3$ and $\lambda_z = 32$ may be the parameter choice that leads to the closest estimate. Looking at the exact values in Table 5.3 this corresponds to a price of 337,750.62 Euro. We will explore this now and reassess this once we consider our Monte Carlo method. So now fixing those parameters for the z-grid we vary the parameters for the y-grid and plot this in Figure 5.2.

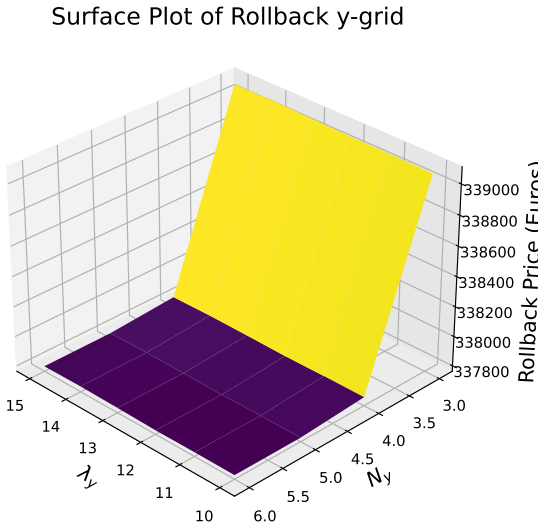

Figure 5.2: Surface Plot of Rollback Values over y-grid

Here we find that a larger number of standard deviations and points per standard deviation leads to a closer estimate of the price, which is to be expected. Since y represents a normal random

variable and is used to calculate the stochastic increments along the path of z, by allowing this to vary over a large number of standard deviations we can effectively observe some of the more extreme values that might occur and include this in our pricing. We see that the price increases quite sharply when we set the number of standard deviations to be below 4, which supports this claim. One potential source of error coming from this method is since y influences the change between each time step, it is possible that our volatility structure does not match the evolution of volatility over time that would actually be observed in the market. Particularly since we use piecewise constant volatilities a certain amount of stationarity is assumed, so a greater degree of variance would be required in order to more closely align the model to the market. This inconsistency may be appearing here implicitly in our choice of parameters. One other thing to note from Figure 5.2 is that the choice of λ_y or the number of points per standard deviation appears to have little effect on the price in the y-grid.

We also present below a table of the exact values generated that correspond to Figures 5.1 and 5.2 in Tables 5.3 and 5.4 respectively.

Table 5.3: Table of Rollback Values Varying z-grid

λ_z \ N_z	3	4	5	6
20	342156.43	346583.61	349217.28	352499.21
24	339962.51	343776.76	345623.52	347904.78
28	338677.34	342120.62	343492.03	345168.86
32	337750.62	340952.17	342014.86	343299.32

Table 5.4: Table of Rollback Values Varying y-grid

λ_y \ N_y	3	4	5	6
10	339066.13	337820.79	337790.12	337789.79
12	339070.37	337821.31	337790.46	337790.12
14	339081.87	337831.03	337800.07	337799.73
16	339033.34	337781.66	337750.62	337750.28

From the tables it would appear the closest price we can generate to the original price of the "Exercise Into" option, 329,476 Euro, is 337,750.28 Euro. The original callable price was -9,285,649 Euro, and the corresponding price from the Rollback is -9,303,738.91. This was obtained using our calculated underlying Swap price of -8,965,988.63 Euro. We present Table 5.5 below with the absolute and relative errors for each of these quantities.

Table 5.5: Table of Single Currency Rollback Errors and Relative Errors

Underlying Swap	
Absolute Error	9815.63
Absolute Error Relative to Price	0.1096%
Absolute Error Relative to Notional	0.020%
Exercise Into Swaption	
Absolute Error	8274.28 Euro
Absolute Error Relative to Price	2.5113%
Absolute Error Relative to Notional	0.016%
Callable Swaption	
Absolute Error	18089.91
Absolute Error Relative to Price	0.195%
Absolute Error Relative to Notional	0.036%

The most reliable measure of error for these types of products is typically relative to the notional, as the prices of each component tends to vary significantly in magnitude such as with the "Exercise Into" Swaption. Each price generated by the Rollback is within 5 basis points of the actual quoted price which given the differences we described earlier is actually quite accurate.

The Rollback method takes between one and four seconds and increases somewhat linearly as the total number of grid points increases. It took two seconds to compute the price we quoted. We present a surface of the computation time over the choices of λ_z, N_z to support this in the Appendix in Figure B.1.

5.1.4 American Monte Carlo

We implement the same payoff function for the underlying Swap that was used in the Rollback implementation. With the Monte Carlo method we can test whether our simulations are correct using the General Pricing Formula 2.1.4. Under the LGM measure associated with the numeraire $N(t)$, we require that the value of the Swap, previously denoted by the payoff $P(t, z_t)$, when divided by the numeraire and taken inside an expectation should equal the time 0 value of the remaining swap payments. In other words, we can check the following relation

$$\mathbb{E}\left[\frac{P(t, z_t)}{N(t)}\right] = V_t(0, z_0)$$

The payoff structure of the Swap at time t will differ from time 0, as we may pass a number of payment dates within $[0, t)$. So using the General Pricing Formula for a payoff at a future time t will not recover the original Swap price, but the present value of the remaining Swap, which we call $V_t(0, z_0)$. So we compare the present value of the remaining Swap payments at time t, using the analytical pricing formula for a Swap, to the value generated by our simulations. To simulate the Swap value at time t, we generate a number of paths of z and use this to calculate the payoff and the numeraire. We can then generate our Monte Carlo estimate for $\mathbb{E}\left[\frac{P(t,z_t)}{N(t)}\right]$. We test this by generating a sample of 800,000 paths and testing at every exercise date.

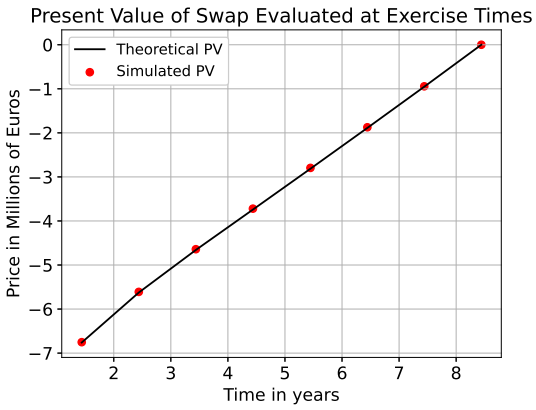

Figure 5.3: Swap Present Value

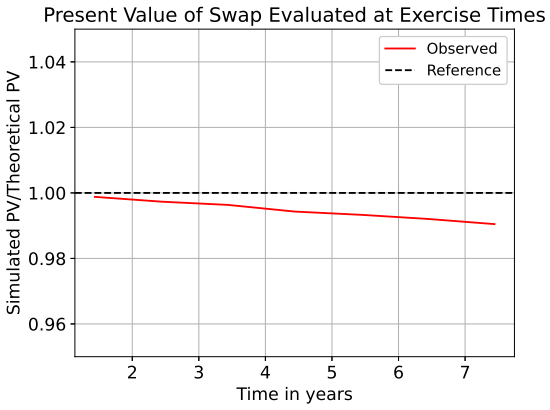

Figure 5.4: Swap Simulated/Theoretical Value

Assessing Figure 5.3, the points in red are our simulated present value estimates, evaluated at the individual exercise dates for the Swaption. The black line is the theoretical present value calculated using an analytical formula. It is quite clear to see that the value generated by our simulations is in line with what we would expect theoretically. So we can be confident that our Monte Carlo method is effective and our simulations are working correctly.

Though it is not immediately clear from this graph how large the error might be, so we supplement this with Figure 5.4. In this plot, we took the previously calculated simulated and theoretical present values and divided one by the other, plotted in red. We would expect this to be equal to 1, which is represented by the black dashed reference line. Here we can clearly observe a decay in the accuracy of the General Pricing Formula as time progresses. This is observed for the LGM model in [12] as well in fact, but for the price of a Zero Coupon Bond. It is explained in [12] why these inaccuracies arise over time and we explain these in Appendix A.2.4. We also suspect this is happening because of our choice of volatility structure. By using constant volatility values we will not be able to capture the evolution of the volatility over time. Piecewise constant volatilities overcome this to some extent but still introduces some level of inaccuracy, and this could be affecting the price. After seven years we observe an error of 1%.

Using our Monte Carlo method with two million simulations iterating only over exercise dates as necessary we produced a price of 348,380.49 Euro for the "Exercise Into" Swaption and subsequently a price of -9,314,369.12 Euro for the Callable Swaption. The 95% confidence interval for the Callable Swaption price is $[-9315345.69, -9313332.02]$, and for the"Exercise Into" estimate is $[347343.39, 349357.06]$. Since we have already assessed the price of the underlying Swap we will only consider the error of the Callable and "Exercise Into" Swaption errors relative to the market price in Table 5.6.

Table 5.6: Table of Single Currency Monte Carlo Errors and Relative Errors

Exercise Into Swaption	
Absolute Error	18904.49 Euro
Absolute Error Relative to Price	5.7377%
Absolute Error Relative to Notional	0.038%
Callable Swaption	
Absolute Error	28720.12
Absolute Error Relative to Price	0.309%
Absolute Error Relative to Notional	0.057%

We see errors of a similar order to the Rollback method from Table 5.5. However, we do note that the errors for the Monte Carlo estimate are slightly higher. This could be due to our choice of parameters for the Rollback, or due to the overestimation we mentioned for Longstaff-Schwartz.

Reassessment of Rollback method

At this point, we will briefly reassess our choice of parameters in the Rollback method. The closest estimate we can find to the Monte Carlo estimate is 347,904.78 Euro, based on Table 5.3, corresponding to $N_z = 6$ and $\lambda_z = 24$. We present again a table of the values generated by the rollback method while varying parameters over the y grid.

Table 5.7: Table of Rollback Values Varying y-grid

λ_y \ N_y	3	4	5	6
10	346756.16	347869.85	347894.20	347894.42
12	346685.37	347803.28	347827.81	347828.03
14	346771.52	347891.85	347916.50	347916.71
16	346758.02	347880.06	347904.78	347905.00

We can see that the majority of these values, save for when we use less than 4 standard deviations, fall within 95% confidence interval. We can see to some extent that there is less variation among the prices where $N_y = 5, 6$ and $\lambda_y = 14, 16$, than the prices where $N_y = 3, 4$ and $\lambda_y = 10, 12$, in other words the lower right and upper left corners of Table 5.7 respectively. So we might conclude that we are converging to some stationary point on the grid as we use more points in total and consider a greater number of standard deviations. By considering more standard deviations we are able to consider more extreme values within our grid which should give us a more accurate price. This is a standard result that the more points we use in the grid the greater our accuracy, but this need not always be the case [42]. This has not previously been studied so explicitly for the Rollback method either to the best of our knowledge. For this reason, we would suspect that the method is converging to a value of 347,905 Euro. From here we will refer to this as our benchmark Rollback price. The Rollback price is still marginally lower than the Monte Carlo price, so the overestimation bias due to Longstaff Schwartz is likely presenting itself here.

Monte Carlo Convergence and Error

Before assessing the error we will consider the convergence of the Monte Carlo method relative to the quoted market price, the rollback price closest to the market price, and one of the closest rollback prices to our Monte Carlo estimate. We present this in Figure 5.5.

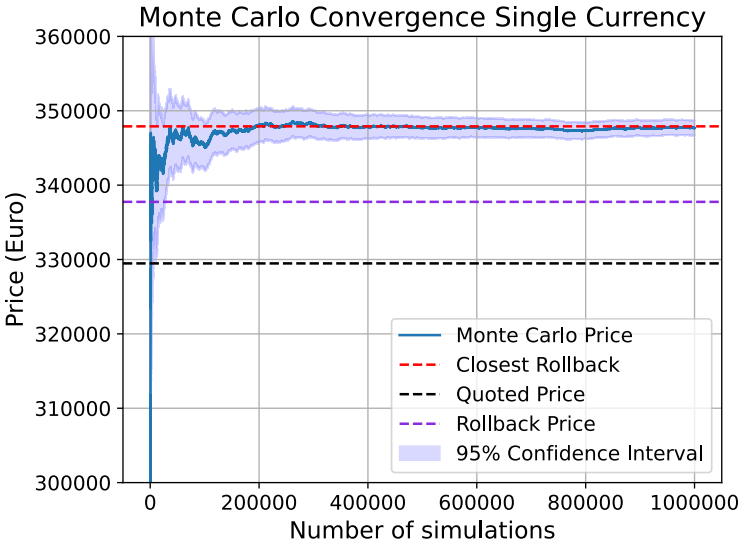

Figure 5.5: Convergence of Estimate of Exercise Into Swaption Price (Single Currency)

In Figure 5.5 we have plotted in blue the value of our Monte Carlo estimate as the number of simulations increases. Similarly, we calculated the 95% confidence interval of this estimate which is represented by the shaded blue area. The red dashed line represents the Rollback value from Table 5.7 that is closest to the Monte Carlo estimate, in particular, this corresponds to $N_y = 6$ and $\lambda_y = 16$, giving a price of 347,905 Euro. The purple line represents the Rollback price we mentioned previously as being closest to the market price, and the black line is the quoted market price.

From this, we might expect that the true price based on our implementation is actually this value of 347,905 Euro, since both methodologies are based on the same contract, data and payoff structures. This is also contained in the confidence interval based on two millions simulations. We can see that the Monte Carlo method appears to have converged after roughly 300,000 simulations. Even around 100,000 simulations, the estimate is quite close. We can assess the rate of convergence more accurately by plotting the error against the number of simulations on a log-log scale in Figure 5.6.

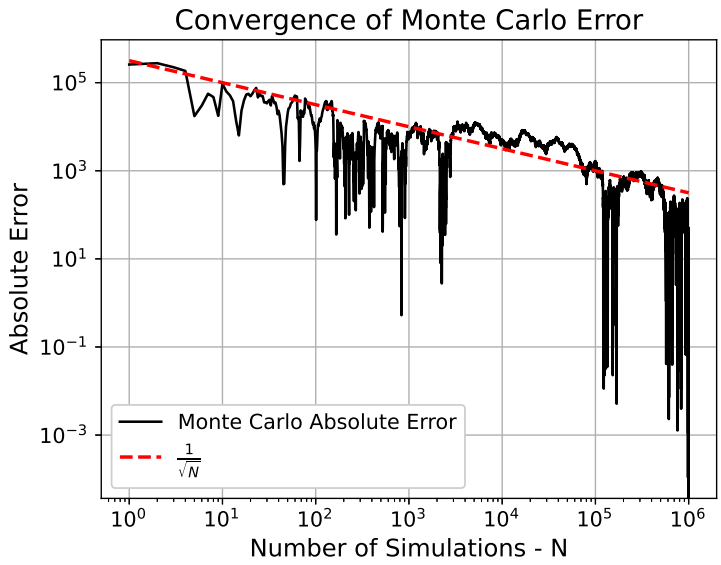

Figure 5.6: Monte Carlo Error Convergence Plot (Single Currency)

The reference line is $\frac{1}{\sqrt{N}}$ where N is the number of simulations, with an appropriate shift

applied. From this we can quite clearly see that this is the rate of convergence of our Monte Carlo estimate. In other words in order to halve the size of our error we must quadruple the number of simulations. This is to be expected and is a standard result as mentioned in [43, Chapter 7], so we have confidence that this method is effective. The sudden drops at higher numbers of simulation are simply due to random noise where the error was close to 0.

Following our discussion of figure 5.6 we use the rollback price of 347,905 Euro as a benchmark, and use this to assess the error of our Monte Carlo estimate in Table 5.8.

Table 5.8: Table of Monte Carlo Errors from Closest Rollback Benchmark

Number of Simulations	10,000	50,000	100,000	500,000	1,000,000
Absolute Error	6276.07	1105.06	2722.16	215.01	223.27
Error Relative to Price	1.804%	0.3176%	0.7824%	0.0618%	0.0642%
Error Relative to Notional	0.0126%	0.0022%	0.0054%	0.0004%	0.0004%
Computation Time (seconds)	0.5559	0.7318	1.0654	6.2315	12.2505

Here, we can see more clearly just how small the error is even with only 10,000 samples we manage to achieve an error of just over 1 basis point relative to the notional. The absolute error more than halves if we increase this to 50,000 or 100,000 samples. After 500,000 simulations the absolute error is within 300 Euro, just over 6 basis points relative to the price, and is less than a tenth of a basis point relative to the notional. The Monte Carlo method then clearly converges quite quickly to an acceptable level of accuracy on par with the Rollback method.

In terms of computation times, the Monte Carlo-based method will likely have converged within 2 seconds of computation time. This is faster than our new benchmark price computed by rollback which takes 3.1750 seconds to compute.

There is a trade-off between accuracy and computation speeds with these. Rollback converges to the long-running Monte Carlo and does not have any stochastic dependencies so always produces the same output for a given calibration. This can be desirable if we are able to calibrate the model to produce a very close estimate. However, Monte Carlo will run more quickly with lower numbers of simulations but will still be subject to more random error and potential overestimation. In order to converge to a level of accuracy on par with the rollback it could take up to twice as long. Though based on Table 5.8 we suspect it could achieve similar levels of speed and accuracy at 300,000 or 400,000 simulations.

Choice of Basis Functions

We explored different sets of polynomials that could be used as basis functions, specifically power series, Laguerre polynomials, and Hermite polynomials. We chose these simply as a subset of the suggestions made in Longstaff Schwartz's original paper [2], and the ease of implementation. The definition of each of these is provided in Appendix A.3. We explore each of these over multiple degrees of freedom as well. The quality of the regression, for the continuation value calculation, is assessed by comparing the predicted regression value to the equivalent value generated by the Rollback method. We consider the regression performed at the first exercise date, where the underlying swap is used as the independent variable.

In Figures 5.7, 5.8, 5.9 we plot in black the benchmark Swaption value at the given exercise date from the Rollback method. We then repeat the regression on a given class of polynomials and increase their degree from one to six. We used 500,000 simulations for each plot.

We first present the plot for the standard power series in Figure 5.7, followed by the Laguerre polynomials in Figure 5.8, and finally Figure 5.9 illustrates the Hermite polynomials.

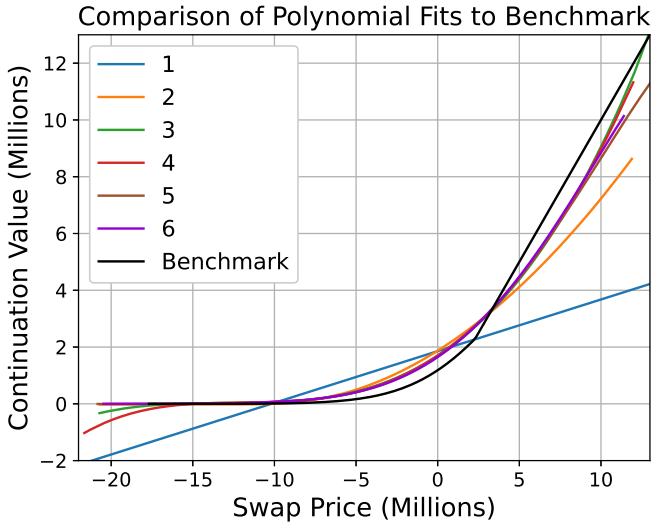

Figure 5.7: Comparison of Polynomial Fits to Benchmark (Single Currency)

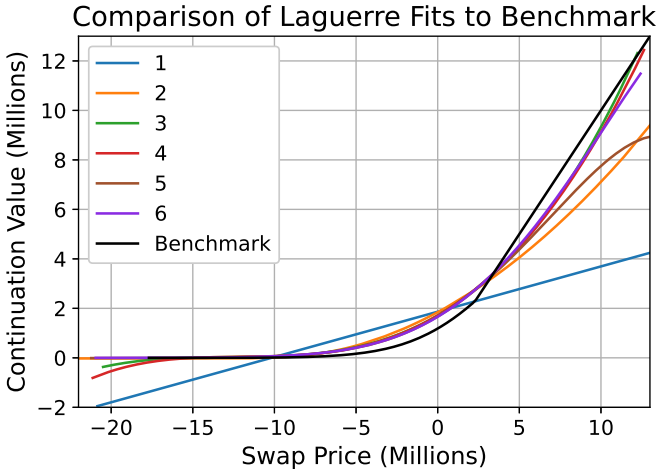

Figure 5.8: Comparison of Laguerre Polynomial fits to Benchmark (Single Currency)

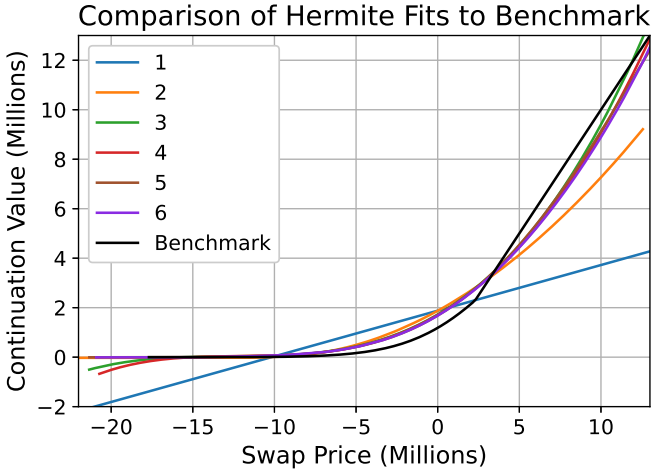

Figure 5.9: Comparison of Hermite Polynomial fits to Benchmark (Single Currency)

It would appear that there is little difference between the Hermite and power series polynomials, both fit close to the curve. In the Laguerre polynomials, we see an interesting case with the fifth-degree polynomial, it diverges from the benchmark just beyond a swap value of 10 million. This is due to overfitting, if we were to overlay the set of underlying points for the regression we would find the line passing through a number of outliers. This leads us to believe that more than 4 terms is likely unnecessary for this step.

From simple graphical inspection, we can see that the Hermite polynomials tend to gravitate towards the benchmark more consistently than the power series polynomials as the degree increases. Generally from all three plots we can see that 3 or 4 degrees of freedom is typically sufficient for our regression. Since Hermite polynomials seem to offer the most consistent fits we will proceed with these.

One other interesting thing to point out is that the benchmark has a very slight kink around 2 million where it becomes linear, this is the point where it is optimal to exercise early and the value changes to the value of the underlying swap. Indeed we can see that after the value of the swap reaches roughly two million the benchmark is exactly the line $y = x$, meaning the Swaption value is the value of the immediate payoff. The comparison between the continuation value and immediate payoff occurs after the regression step, which is why the benchmark sits above the continuation values after this point.

We might have suspected that Hermitian polynomials would work well as they are often connected to Brownian Motion and probability theory according to [44]. Gauss-Hermite Quadratures [45] are also a popular numerical integration technique, where the Hermite polynomials are used in the calculation of the weights. Some similarities can be drawn in this sense between the quadrature in our Rollback method, and the regression step here. For these reasons, the Hermite polynomials are likely well-suited to the types of problems we are dealing with. Though perhaps a future area of research should be a more fundamental approach to this.

We also tabulate the Root-Mean-Square-Error(RMSE) and R-Square values from the Hermite polynomials at each degree in Table 5.9. If we consider the Swaption valuations from our Monte Carlo estimate as y_i and our predicted value from regression as $\hat{y}_i$ for simulation i, with P degrees of freedom, these are defined as

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N - P}}$$

$$R^2 = 1 - \left(\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \bar{y})^2} \right)$$

The RMSE gives us an absolute measure of how close the regression fit is to the actual data, while the R^2 statistic gives us an indication of how much of the variance in our data can be explained by the relation implied by our regression model [46].

Table 5.9: Table of Regression Statistics (Single Currency)

Polynomial Degree	1	2	3	4	5	6
RMSE	416354.07	336798.19	320338.86	321875.86	322105.49	322486.78
R^2	80.14%	84.14%	85.68%	83.52%	83.07%	83.05%

From Table 5.9, we can quite clearly see that the RMSE begins to increase slowly after the third degree, the R^2 statistic decreases after this point as well. So we can conclude that 3 degrees of freedom is sufficient for our regression. These observations are generally in keeping with standard practice for these types of problems in the literature, we have previously mentioned that [12] and [41] use standard power series polynomials up to the third degree in their work. In fact based on this trend we might suspect that after three degrees of freedom our model is overfit [47]. Our RMSEs are not overly large, but are not small either in the context of a 50 million notional. The R^2 statistic suggests 85% of the variance in our data can be explained by our model [47]. Overall the model seems to perform quite well and manages to predict the continuation value effectively. We also considered the RMSE and R^2 statistics for the power series and Laguerre polynomials and found these to be almost identical.

Following this we also briefly assess a potential alternative regression problem, in which we replace the underlying Swap value with the simulated values of the process z. Since this is the

driving process of the price it should be possible to estimate some functional form that can accurately predict the continuation value. However we would like to emphasize that the z process is typically not measurable in the real world, and so a modeling decision like this is only really valid in a simulation context.

We begin by presenting a plot of the estimated continuation values against the Rollback benchmark under the same conditions as before, with Hermite polynomials in Figure 5.10.

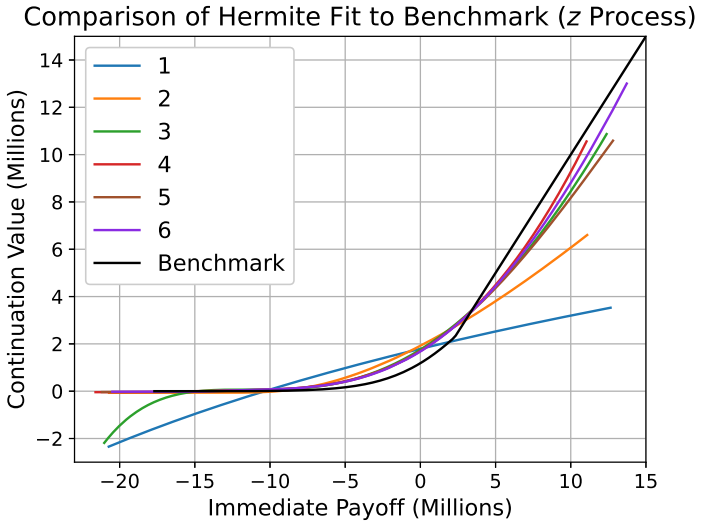

Figure 5.10: Comparison of Hermite z Regression fits to Benchmark (Single Currency)

Note that the x-axis is the immediate payoff or Swap price implied by the values of z we observed in order to compare to the Rollback benchmark. The predicted values fit the Benchmark quite closely. This is expected as the price of the Rollback is driven by the same process, and it is ultimately this process that drives the price of the underlying Swap and Swaption at future times. From the graph, the fit does not appear to be significantly better than those in Figure 5.9. This fit leads to slightly higher RMSE values and lower R^2 values. We present the exact results in Appendix B.1.2. The fit is only marginally worse than when we regress on the Swap value. Ultimately we prefer to regress on the Swap values, since the underlying z process itself is not exactly tangible or measured and offers no significant improvement in accuracy.

5.2 Cross-Currency Bermudan Swaption

Now that we have assessed how well Monte Carlo and Rollback work in the single-currency case, we will see if our findings are consistent for the cross-currency case. We can of course only assess the Monte Carlo method as we discussed earlier. To the best of our knowledge, the combination of the LGM model, the American Monte Carlo method, and cross-currency Swaptions has not been thoroughly studied on real trades or with real data. Some numerical examples have been assessed in [12], but the accuracy and convergence of the method were not considered. Alongside this, our given trade is also an unexpected edge case problem that demonstrates some of the shortcomings of the method when dealing with options that are deep-out-of-the-money. This topic has been assessed in papers such as [48], and [49] using Black Scholes, but we believe the analysis has not been performed on the LGM model, we also aim to provide a unique analysis of how this affects our Monte Carlo estimate.

5.2.1 Context and Data

We will outline the trade details for the cross-currency Bermudan Callable Swaption. We considered another real trade that was committed in June 2021. We consider the valuation of the trade in June 2023, when we began this project, the contract then ends in June 2025. We provide Table 5.10 for the details of the trade.

Table 5.10: Table of Trade Details (Cross Currency)

Detail	**Domestic Value**	**Foreign Value**
Notional	50,000,000 GBP	70,702,500 USD
Payment Frequency	1 Year	3 Months
Floating Rate Curves	-	3 Month USD LIBOR Rate
Fixed Rates	0.55%	-
Discount Rate Curves	GBP-USD 3 Month Cross-Currency Basis Curve	USD Secured Overnight Financing Rate (SOFR)
Spread	-	-0.05%
Daycount Convention	30/360	ACT/360
Frequency of Early Exercise	1 Year	-
Receive/Pay	Receive	Pay
"Exercise Into" or Callable	"Callable"	-
Mean Reversion Parameters	0.0256	0.0425

The set of volatilities are presented in Table 5.11.

Table 5.11: Table of Piecewise Volatility Values (Cross Currency)

Date	$\sigma_d(t)$	$\sigma_f(t)$	$\sigma_x(t)$
13/6/2023	0.018227	0.017742	0.105397
20/6/2023	0.018227	0.017742	0.071180
12/7/2023	0.018227	0.017742	0.073067
10/8/2023	0.018227	0.017742	0.077053
12/9/2023	0.018227	0.017742	0.074186
18/9/2023	0.018227	0.015437	0.083769
12/12/2023	0.018227	0.015615	0.08473
12/3/2024	0.018227	0.015537	0.086889
12/6/2024	0.018227	0.016736	0.089060
20/6/2024	0.016898	0.016736	0.094436
16/9/2024	0.016898	0.013720	0.094436
12/12/2024	0.016898	0.01867	0.087593
17/3/2025	0.016898	0.010540	0.087593
12/6/2025	0.016898	0.010540	0.097803

The correlation matrix between the Brownian Motions from the model calibration is presented in Table 5.12.

Table 5.12: Correlation Matrix Brownian Motions (Cross Currency)

	W_d	W_f	W_x
W_d	1	0.170645	0.221809
W_f	0.170645	1	0.655462
W_x	0.221809	0.655462	1

We also have from calibration that the initial value of the FX process is 0.799616. The market quoted price of the "Exercise Into" Bermudan Swaption is 779 GBP, the price of the underlying Swap as of the 12th of June 2023 is -11,423,473 GBP. This is again negative because the value of the domestic leg, which we are receiving, is lower than the value of the foreign leg we are paying. Subtracting one from the other we get the price of the Callable Bermudan Swaption as -11,424,252 GBP.

As we discussed previously in Section 2.3, there are several martingales that are used to determine the drift of the FX and Foreign processes. In particular, under the Cross-Currency LGM model, we require that the following processes are martingales

$$\tilde{P}_d(t,T,z_d), \quad Y(t) = \frac{N_f(t)x(t)}{N_d(t)}, \quad U(t) = \frac{P_f(t,T)x(t)}{N_d(t)} \quad [12].$$

We can assess this by simulating each process, and then calculating the expected value of each using Monte Carlo. This should yield the value of each process at time 0 because of the martingale

property. In particular the relations

$$\mathbb{E}\left[\tilde{P}_d(t,T,z_d)\right] = P_d(0,T)$$
$$\mathbb{E}\left[\frac{N_f(t)x(t)}{N_d(t))}\right] = \frac{N_f(0)x(0)}{N_d(0)} = x(0)$$
$$\mathbb{E}\left[\frac{P_f(t,T)x(t)}{N_d(t)}\right] = \frac{P_f(0,T)x(0)}{N_d(0)} = P_f(0,T)x(0)$$

should be observable. These quantities can all be calculated directly from our market data and parameter choices. We perform the necessary simulations and expectation calculations and plot how this progresses as time, t, increases for fixed T. We also divide by the time 0 value of these processes, so that our plots should be centered around unity. We present this in Figures 5.11, 5.12, 5.13.

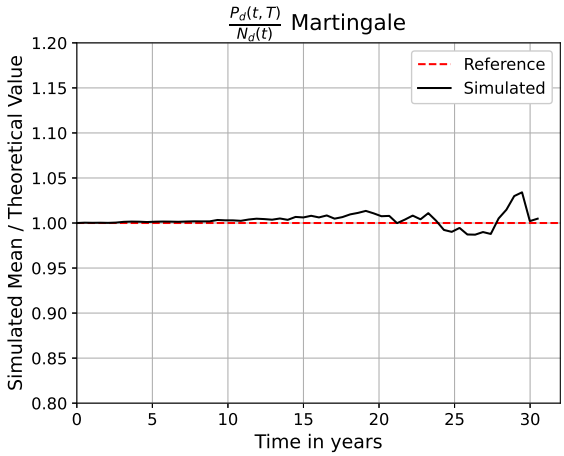

Figure 5.11: Domestic Zero Coupon Bond Martingale

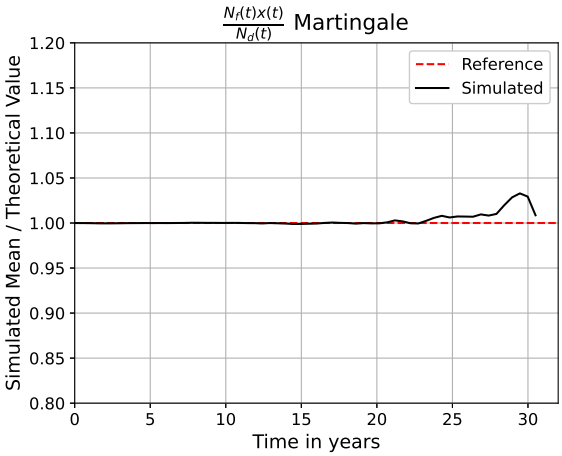

Figure 5.12: Y Martingale

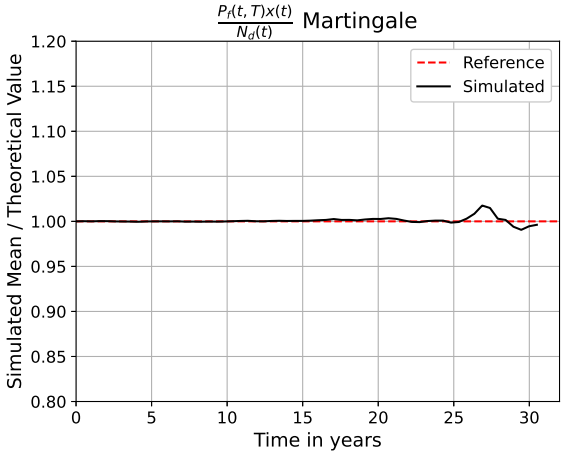

Figure 5.13: U Martingale

Figure 5.14: Plot of Martingales Expectation divided by Time Zero Valuation

We used 800000 samples for our Monte Carlo estimate of the mean. We can see that each plot remains quite close to 1, and we only start noticing significant departures around 20 years or more. A similar analysis is performed in [12] but only for the Zero-Coupon Bond. Since the martingale effect seems to be working in our simulations, we can be confident that our model has been implemented correctly. We do notice that the size of the departures increases as time increases. Similar to what we observed for the swap price in the single currency case in Section 5.1.4. It is

explained in [12] why these inaccuracies arise over time and we explain this in Appendix A.2.4. We also mentioned before that the choice of piecewise constant volatilities may lead to inaccuracies in the long run. For the sake of our contract which expires in two years the current level of accuracy is sufficient.

5.2.2 American Monte Carlo

Monte Carlo Accuracy

We will now assess the error of our Monte Carlo method. We ran 10,000,000 simulations, which is significantly more than we used previously and we will explain why later as this is the beginning of a greater point. Using again Hermite polynomials with three degrees of freedom and regressing on the underlying Swap value, as this appeared to be effective in the single currency case. We generate a price for the "Exercise Into" Option of 1501.11 GBP, we also price the Swap at -11,429,653.06 GBP, as such the price of the Callable Swaption is -11,431,154.17 GBP. We note that our Swap price again differs from the quoted price due to differences in interpolation choices for yield curves, here we use linear interpolation. Other minor details may differ such as the decision to exercise being made several days before the actual exercise date. This is done so contract holders can prepare the necessary payments in advance. Despite this our price is still quite close, only differing by roughly 6000 GBP on a 50 million notional. We present a table of errors for our final price estimates of each product in Table 5.13.

There we can see that our estimates are again quite accurate relative to the notional, the price of the Callable Swaption has an error of 1 basis point. If we could guarantee the same interpolation steps this would improve even further. Even better than this, the error of the price of the "Exercise Into" Swaption relative to the notional is only one tenth of a basis point. The error does appear quite large relative to the price, but this is simply due to the fact that the price is so low.

Table 5.13: Table of Errors and Relative Errors (Cross-Currency)

Underlying Swap	
Absolute Error	6180.06
Absolute Error Relative to Price	0.05%
Absolute Error Relative to Notional	0.01%
Exercise Into Swaption	
Absolute Error	722.11
Absolute Error Relative to Price	92.697%
Absolute Error Relative to Notional	0.001%
Callable Swaption	
Absolute Error	6902.17
Absolute Error Relative to Price	0.06%
Absolute Error Relative to Notional	0.01%

Where we have compared relative to the domestic notional, as all prices are converted to GBP.

Monte Carlo Convergence and Error

We will begin looking at the convergence of our method by considering Table 5.14, which presents the errors of our estimate for different numbers of simulations.

Table 5.14: Table of Monte Carlo Errors (Cross Currency)

Number of Simulations	100,000	500,000	1,000,000	5,000,000	10,000,000
Observed Price	1440.11	1510.46	1468.04	1502.44	1501.11
Absolute Error	661.11	731.46	689.04	723.44	722.11
Error Relative to Price	84.87%	93.9%	88.5%	92.9%	92.7%
Error Relative to Notional	1.3×10^{-5}	1.5×10^{-5}	1.4×10^{-5}	1.4×10^{-5}	1.4×10^{-5}
Computation Time (seconds)	2.719	3.365	4.196	10.445	17.529

From here we can see that the price has still not quite converged after 1 million simulations, only around 5 million do we start to notice any kind of consistency, we will explore why so many

simulations are needed once we assess the convergence. But at this stage we manage to obtain relatively high levels of accuracy even for only 100,000 simulations, we achieve accuracy within 2 tenths of a basis point, which is more than acceptable. The level of accuracy seemingly decreases as the number of simulations decreases, which may seem counterintuitive for now. This is also due to a lack of convergence until around 5 million simulations. The computation times are actually shorter than in the single currency case, this is because we have fewer time steps to iterate over in this trade. In general, we would expect longer computation times for similar trades as the simulation is now much more involved. In particular, we now require the calculation of more complicated means and variances for the foreign and FX processes. Alongside this, we must calculate the Cholesky decomposition for the correlation structure at each step. The simulation of 2 additional normal random variables will of course increase the amount of memory required as well.

The 95% confidence interval for our price estimate is $[1500.89, 1501.33]$ after 10 million simulations. This is a very tight confidence interval primarily due to the fact that the standard deviation of our samples is quite small, as we saw in Table 5.14, the absolute error is typically within 1000 GBP. Recalling the formula for the confidence interval as $\bar{X}_N \pm 1.96\frac{\hat{\sigma}_N}{\sqrt{N}}$ for N simulations, it is easy to see how this tight interval would occur.

In order to properly understand what is happening in the convergence of our method we should explain a number of implementation details for creating convergence plots. Typically to plot the convergence of a Monte Carlo estimate one would iteratively calculate a cumulative mean as the number of simulations increases and plot this. This could be done by first generating the simulations and then calculating the cumulative mean, or repeating the process and tracking the historical mean at each step. In the standard case there is no real difference, but in the context of American Monte Carlo we have to consider a regression step. If we run all our simulations and then calculate the cumulative mean then we end up using all of our simulations in our regression step rather than incrementally increasing the number of data points we use with the number of simulations. Not increasing the number of regression data points sequentially is a simple change but can actually be quite telling in this case. So we begin by presenting the correct calculation of the Monte Carlo convergence but then later present this "spurious" method in order to aid in our discussion.

We begin by assessing the convergence of the error on a log-log scale using the price of 1501.11 as our Benchmark. We used 5 million simulations and present this in Figure 5.15.

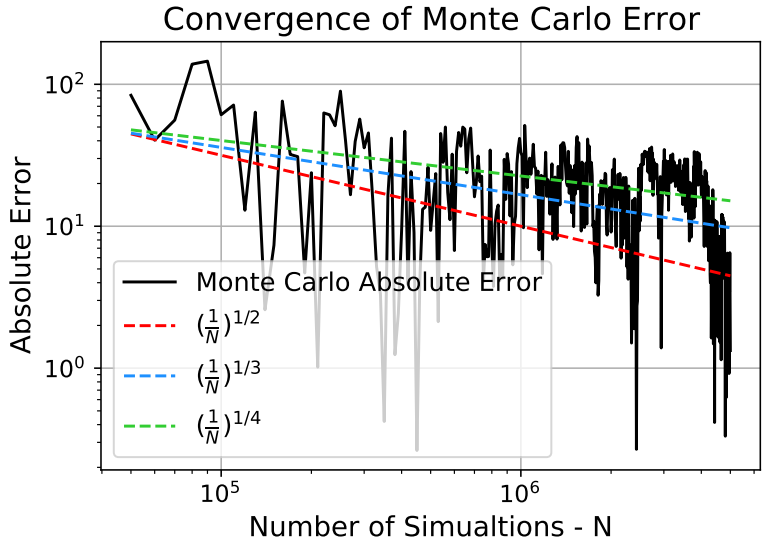

Figure 5.15: Monte Carlo Error log-log Convergence Plot

From this we can see that the rate of convergence might differ from the standard results of $\frac{1}{\sqrt{N}}$. We include in blue and green respectively slopes of $\frac{1}{\sqrt[3]{N}}$ and $\frac{1}{\sqrt[4]{N}}$ to help in our assessment. These do appear to be a closer estimate of the convergence of this method than $\frac{1}{\sqrt{N}}$. In particular, we suspect that the rate of convergence is $\frac{1}{\sqrt[4]{N}}$. We will now also consider the plot of the convergence of the price in GBP over the number of simulations in Figure 5.16.

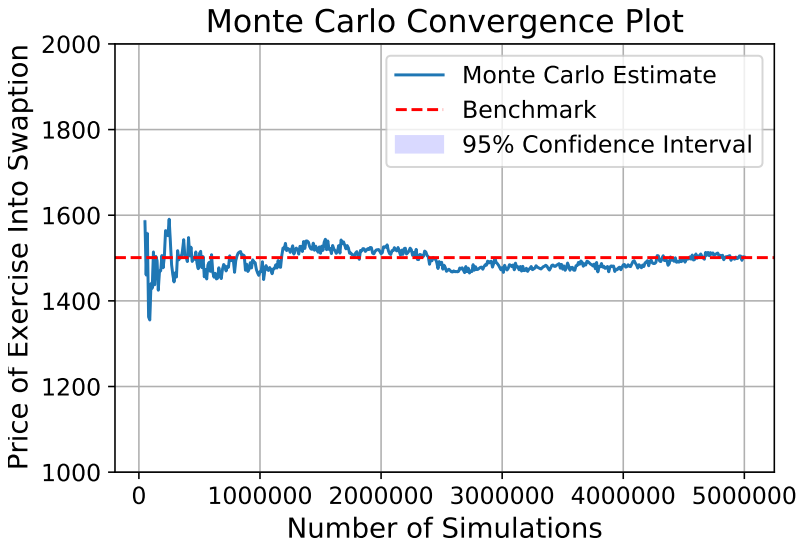

Figure 5.16: Monte Carlo Estimate Convergence Plot

There are a number of interesting things to observe in Figure 5.16. What may initially appear as a mistake is the lack of a 95% confidence interval. We assure the reader that it is present but is simply so small it cannot be seen on this scale. Often being as wide as only 1 or 2 GBP, it seems to be far too small for the level of convergence we observe. This brings us to the second observation that there is very little variability in the price but that it is clearly still slow to converge. This slower convergence is in line with what we observe in Figure 5.15. If the standard rate of convergence is $\frac{1}{\sqrt{N}}$ then to halve the error we would have to quadruple the number of simulations, if our new rate of convergence is $\frac{1}{\sqrt[4]{N}}$ then in order to halve the error one would have to use 16 times as many simulations. This result and type of analysis have not been thoroughly studied in the literature on this topic to the best of our knowledge. We eventually observe convergence just before 5 million simulations are reached. We would expect this slow rate of convergence to manifest a wider confidence interval as well though. There is clearly more to be said about this so as we mentioned we will turn our attention to the spurious calculation of these convergence plots.

To reiterate, when we refer to these spurious plots we mean performing the American Monte Carlo calculation with a fixed number of simulations, so that each regression step uses the total number of simulations, then calculating the cumulative mean across all our paths and plotting this. In order to properly assess some of the strange results we have observed we will plot the convergence of our estimate over two different numbers of total simulations. We first use 100,000 simulations and repeat the experiment 5 times, we present the results in Figure 5.17. Later we will perform a similar test with 10 million simulations.

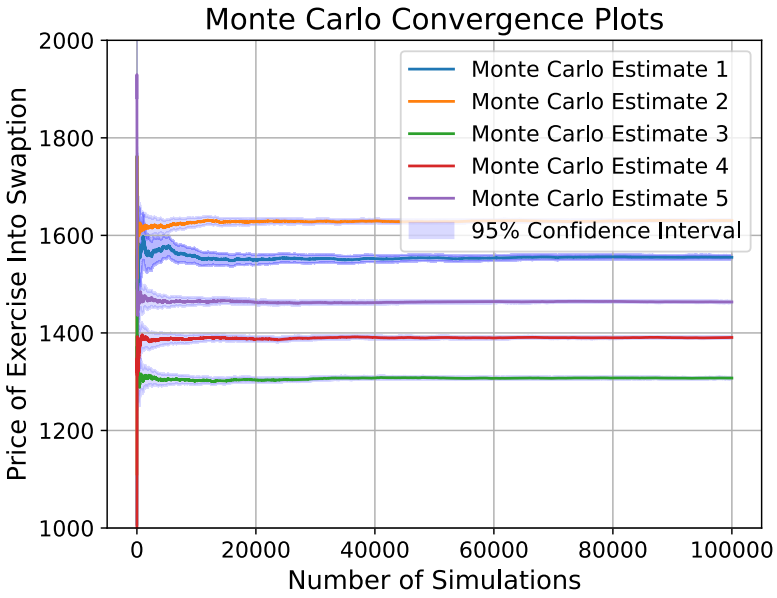

Figure 5.17: Spurious Monte Carlo Convergence Plot 100,000 Simulations

What we observe is quite interesting, particularly that each experiment appears to converge to a different estimate, and that despite the wide range of estimates the confidence intervals for each are extremely narrow and do not overlap. It would appear that our method converges but they have converged much more quickly than in Figure 5.16. We suspect there is some other error that is affecting our estimates. Even if we were to argue that our simulations had not converged, we would expect the confidence intervals to be much larger to reflect this. In order to assess this we will look more closely at the regression step.

Regression Assessment

We explored a number of basis functions again, as well as the quantities we regress on. We first considered Hermite and power series polynomials over degrees 1 to 6. However, we no longer have a benchmark continuation value to compare against. So we are restricted to several regression statistics and plotting the fit of the regression lines to the data. We used 50,000 simulations for this assessment.

We present the plots of the observed values alongside the predicted values from our various regression fits, starting first with the standard polynomial fit in Figures 5.18 and 5.19, immediately followed by the Hermite polynomial fits in Figures 5.20 and 5.21.

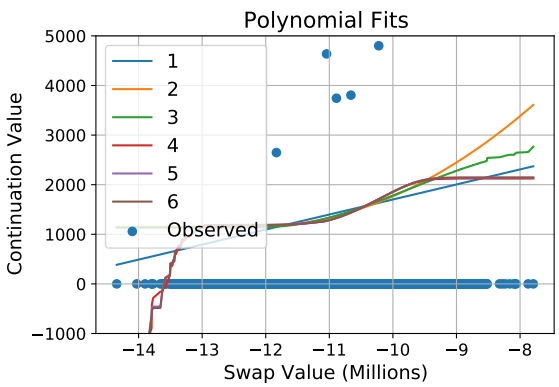

Figure 5.18: Polynomial Fit Zoomed in

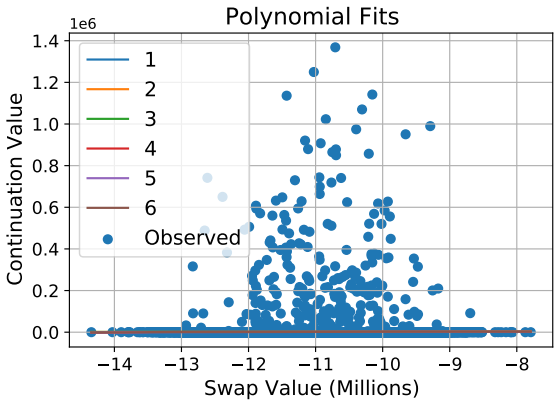

Figure 5.19: Polynomial Fit Zoomed Out

This is the regression step from the last exercise date to the next latest. At that step we set the value of the option to be the maximum between the Swap value and 0. The points are the

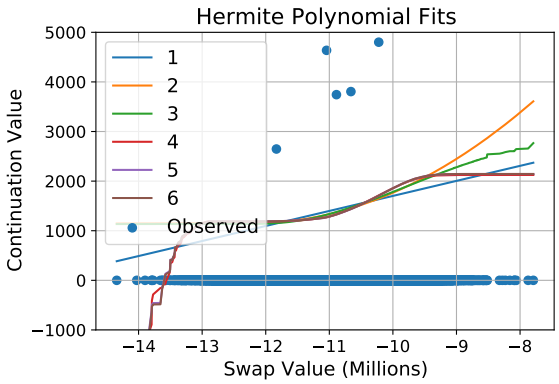

Figure 5.20: Hermite Fit Zoomed in

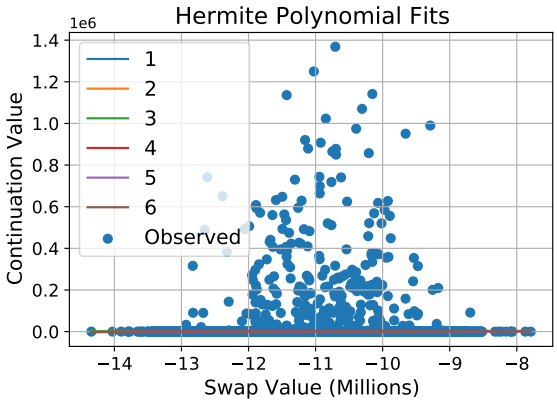

Figure 5.21: Hermite Fit Zoomed Out

observed values, and the lines are our regression fits. We can see from the plots that the majority of the points are 0. In fact of the 50,000 there are only roughly 300 non-zero observations. We can see this from the plot on the right. As we saw before the value of the Swap is negative and the "Exercise Into" price is close to 0, as a result, many of the valuations in our simulations are still 0 after the small number of time steps remaining in this trade. This poses an interesting problem of sparse data when performing the regression step. This makes it difficult to train the model. From Figure 5.21, we can see that our regression functions are set relatively close to 0. Figure 5.20 is a zoomed-in version of the same plot. From this, we can see that the regression estimates actually sit above 0, as their fit is influenced by the non-zero points. The reason the points in Figures 5.19 and 5.21 appear so random is because in order for the Swaption to have a positive continuation value there needs to be a significant change in z_d, z_f, and $\ln(x)$, in the time increment between one valuation date and the next, regardless of the current value of the Swap. Essentially in this case the positive valuations are almost entirely by chance, giving these models low predictive power.

In standard statistical analysis, these might be considered outliers, but here we understand that they are necessary in order to assess the possible extreme values that could be observed, which will affect our valuation. So we cannot remove them. This problem arises because it is almost never optimal to exercise this option. This is an edge case that we believe has not been studied for these products previously to any great extent and highlights a very important shortcoming of the Monte Carlo Longstaff Schwartz methodology. This lack of data affects the quality of our predictions and influences our price in this case.

Returning to Figure 5.17, we now have a better understanding of what is occurring. The Monte Carlo estimates are converging quite quickly leading to the narrow confidence intervals, but the error of the estimate is being dominated by the regression error, due to the sparse data. When these Monte Carlo based methods are used typically a minimal number of simulations are used in order to reduce computation time, 20,000 simulations is typically more than sufficient for this many time steps. From what we see here though due to the excess of zeros in our data the convergence of our estimate is not consistent.

In order to try and overcome this problem we tested a different regression method specifically designed for these types of problems. We used a Double Hurdle regression that was originally proposed in [4]. These models essentially involve first predicting the probability that any given observation will be different from 0. The most common approaches to this are using Logit and Probit regression models, and this is suggested in [4] also. A second standard regression model is then trained, but only on the data corresponding to non-zero observations. The idea is to determine the probability of an observation being 0 and predict the value of a non-zero observation. It is stated in [4] that this is a particularly useful approach for dealing with data of mostly zeros. We implemented a logit model and used power series polynomials of various degrees for the non-zero value part of the regression. We present the plot of the fits below in Figures 5.22 and 5.23.

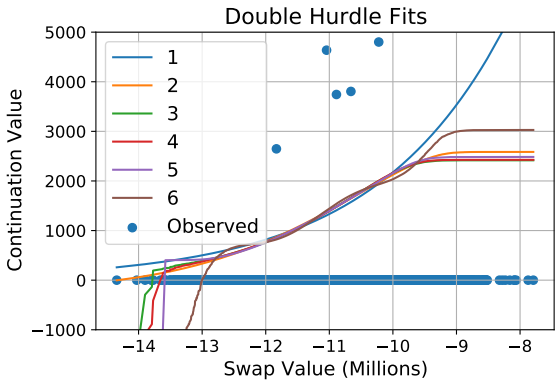

Figure 5.22: Double Hurdle Fit Zoomed in

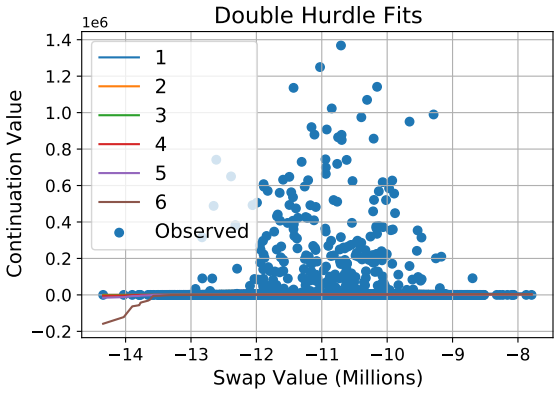

Figure 5.23: Double Hurdle Fit Zoomed Out

From brief graphical inspection the model seems to offer little to no improvement in fit over the two standard polynomial models. Though the models do not fit the data well, it is still effective to the extent that we would expect it to predict values close to but not exactly 0, since we can not completely dismiss the idea of a positive payoff on this product.

We will conduct a more quantitative investigation of the quality of all these fits using the R^2 statistic and Root-Mean-Square-Error as we did before. Starting with the power series polynomials regressed on the value of the underlying swap in Table 5.15.

Table 5.15: Table of Regression Statistics (Polynomial, Underlying Swap)

Polynomial Degree	1	2	3	4	5	6
RMSE	27620.13	27620.01	27619.47	27619.47	27619.06	27619.03
R^2	0.02%	0.03%	0.03%	0.03%	0.04%	0.04%

As we might have expected the R^2 values indicate that the fit of our regression model is quite poor. These values suggest that almost none of the variance in our data can be explained by our model [46]. However, the RMSE values are of a similar magnitude to what we observe in the single currency case in Table 5.9. Little to no improvement is offered by increasing the degree of our polynomial either. This clearly calls for a more effective method for modelling or predicting this type of data.

Looking now to the fit of the Hermite polynomials, we present the same statistics in Table 5.16 below.

Table 5.16: Table of Regression Statistics (Hermite, Underlying Swap)

Polynomial Degree	1	2	3	4	5	6
RMSE	27620.13	27620.01	27619.47	27619.47	27619.06	27619.03
R^2	6×10^{-5}	0.03%	0.03%	0.03%	0.04%	0.04%

We observe that the RMSE values are very similar, as are the R^2 values, to Table 5.15. As we might have expected based on Figures 5.18 and 5.20.

Finally, we repeat this for our new double hurdle model, here we tested regressing on both the underlying Swap value, and the driving processes $z_d, z_f, \ln(x)$. These are presented respectively in Tables 5.17 and 5.18.

Table 5.17: Table of Regression Statistics (Double Hurdle, z_d, z_f, x)

Polynomial Degree	1	2	3	4	5	6
RMSE	27624.21	27622.77	27624.63	27635.73	27802.65	34691.55
R^2	-5×10^{-6}	9×10^{-5}	-3×10^{-5}	-0.01%	-1.30%	-57.71%

Table 5.18: Table of Regression Statistics (Double Hurdle, Underlying Swap)

Polynomial Degree	1	2	3	4	5	6
RMSE	27620.47	27620.45	27621.80	27624.86	27622.94	27853.58
R^2	0.03%	0.03%	0.02%	-5×10^{-5}	8×10^{-5}	-1%

From Tables 5.17 and 5.18 we can quite clearly see that again the fit to the data is quite poor, what is most striking is perhaps the regression on the underlying processes, it appears to be significantly worse. We notice a large negative R^2 value for the polynomial at degree 6. Looking at Figure 5.23 we can see a more significant departure from 0 for the 6th degree polynomials that is not present in the other plots. This is likely what is causing such a poor fit, as it should remain around 0, where the majority of the data lies.

Even on the Swap based regression the fit is quite poor. It is actually worse than the standard regression approach. This is likely occurring due to the fact that by performing the regression only on the non-zero valued points we overfit the model much faster as we have significantly fewer data points. This is evidenced by the decrease in R^2 as the degrees of freedom increase [47]. From Figure 5.22 we can actually see that with 1 degree of freedom we have a similar shape to fits in the single currency case. However, this is still reduced by the low probability weightings for non-zero estimates.

There are many approaches to dealing with sparse data in a regression context such as data Imputation [50], where 0 values can be replaced with the mean or median of the data, though this has a tendency to introduce bias. Neural Networks may be more capable of identifying trends in data such as this. There is unfortunately not enough time for us to explore these ideas, so we instead recommend them as future areas of research in this problem.

As we mentioned previously an analysis of the Longstaff Schwartz approach for Out-of-the-Money options was performed in [48] and [49], where it was concluded that in these cases the American Monte Carlo method does not perform well on out-of-the-money options. This is in keeping with what we have found here and we can draw a similar conclusion, but we make additional notes about how it affects the convergence of our estimates and that this is due to the sparse data we have encountered. Despite these underlying issues this presents, our estimates are still quite accurate relative to the notional as we saw in Table 5.14.

Finally returning to Figure 5.15 we now understand that this slow rate of convergence is likely being caused by the regression step and sparse data. In order to confirm this we can consider the convergence when we consider an extremely high number of simulations.

Convergence with a High Number of Simulations

As we stated at the beginning of this section the Monte Carlo method converged to a single price after roughly 5 million simulations. We now better understand why this is required, in order to obtain a sufficient number of non-zero data points in our regression we require significantly more simulations than we would otherwise expect. We proceed by plotting the error against the number of simulations on a log-log scale in Figure 5.24, this time using 10 million simulations with our spurious convergence calculation, so that we maximise the number of points we use in our regression.

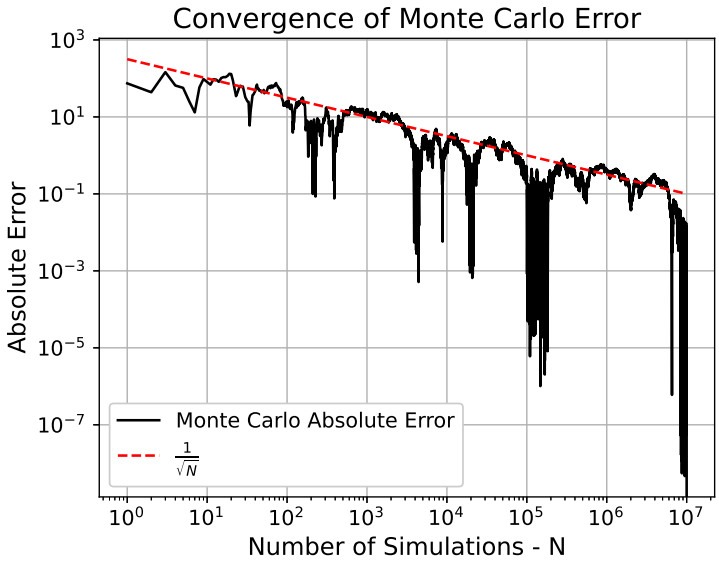

Figure 5.24: Spurious Monte Carlo log-log Error Convergence Plot

We can see that we now observe the standard result of a rate of convergence of $\frac{1}{\sqrt{N}}$ [43]. At the final point, the error drops to 0 as we are comparing our set of simulations against the final value generated from the same simulations. So clearly when we reduce the error introduced by the regression step as much as possible, the underlying Monte Carlo estimate behaves as normal.

So we will now present another spurious plot of the convergence of our estimate against the number of simulations in Figure 5.25, this time using 10 million simulations.

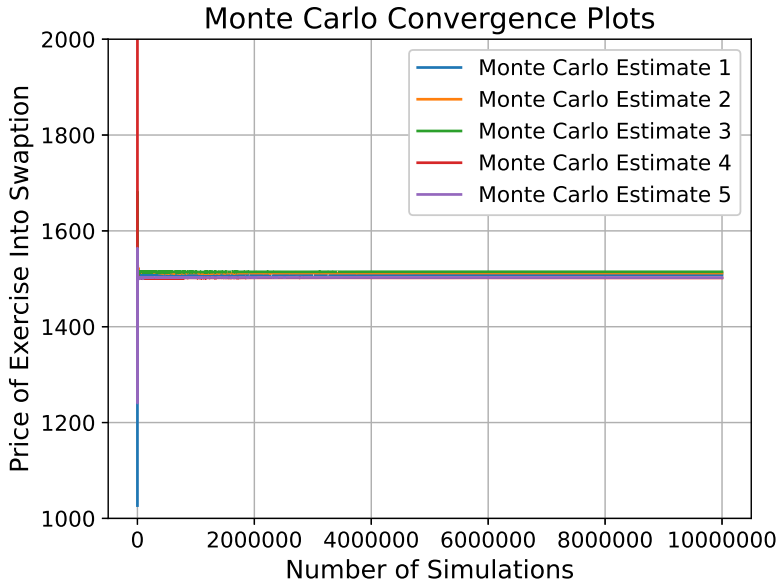

Figure 5.25: Spurious Monte Carlo Convergence with 10 Million Simulations

We see the method has converged and is nowhere near as volatile as when we used 100,000 simulations in Figure 5.17. With 500,000 simulations one would observe slightly reduced levels of variation to Figure 5.17. Increasing again to one million we could achieve an even tighter spread and we present this in Appendix B.2, around 5 million we have a very similar range of convergence to what we observe in Figure 5.25. In each case we observe that each set of simulations converges quickly, but the spread gets tighter between them as we increase the total number of simulations. This demonstrates that the strange results we were observing before can be attributed to the error introduced in the regression step. Because of this we suspect that in a standard case where the

underlying Swap is more in-the-money we would see convergence relatively quickly, potentially within 100,000 simulations as this appears to be a sufficient level of convergence if we were to ignore the effects of sparse data.

Finally, we will assess the regression statistics on 10 million simulations in Table 5.19.

Table 5.19: Table of Regression Statistics (Hermite, 10 million)

Polynomial Degree	1	2	3	4	5	6
RMSE	28873.89	28873.83	28873.83	28873.83	28873.83	28873.82
R^2	0.02%	0.02%	0.02%	0.02%	0.02%	0.02%

The RMSE values are of a similar magnitude to our previous results in Tables 5.16 and 5.9. However similar to what we observed in Table 5.16 the R^2 values are extremely low. The combination of the low RMSE and R^2 values suggests that we make predictions that are quite close to actual observed data points, but that there is little correlation between the dependent and independent variables in our data [47]. In other words we can make a fair guess at what the continuation value may be, but there is little or no relation between the past Swap value and the continuation value so individual predictions are poor. This is occurring due to the excess of 0's we observed, making it difficult to distinguish between paths with different Swap values but 0 continuation value. At 10 million simulations we expect roughly 60,000 points to be non-zero, which is still on the lower end of what we had required previously for convergence in a standard case. As we saw before as well in Figures 5.19, 5.21, and 5.23, the distribution of non-zero points is quite random as a positive valuation requires a large random shock on the processes of z_d, z_f and $\ln(x)$.

Chapter 6

Conclusion

In this thesis we sought to explore a number of methods of pricing Cross Currency Callable Bermudan Swaptions. We introduced the Linear Gauss Markov model by deriving it from the Hull-White model. In our results section we tested our implementation by demonstrating the martingale property for a number of quantities in both the single-currency and cross-currency case.

We then used this model to implement two different pricing methods in the single-currency context. Hagan's Rollback method [1] was considered first, and we began by exploring the choice of parameters. We generally found that accuracy and convergence improved as we used more points in each grid. This is a standard result for Finite Difference methods [51], which work similarly to Hagan's Rollback. This exploration also helped demonstrate how much the price can vary depending on our choice of parameters, offering a range of roughly 10,000 GBP. We also noted that computation times are between roughly 1 and 4 seconds increasing linearly as the total number of points increases. Following this we considered the implementation of the Rollback method with the cross-currency LGM model. We quickly learned that we would require three times as many grids if we were to proceed in an identical way to the single currency case. This makes the method quite intractable in terms of implementation and we can only imagine would lead to significantly longer computation times as well. So ultimately this method was unable to overcome the curse of dimensionality which plagues similar numerical methods [52]. We offered a potential alternative methodology using Cubatures on Wiener space [31], and we recommend this as a future area of research in the problem.

We changed our focus then to implementing a Longstaff-Schwartz Monte Carlo scheme [2] for both the single-currency and cross-currency Swaptions. In the single-currency case we provided a unique comparison with Hagan's Rollback method. We find that Monte Carlo is slightly faster but offers lower accuracy or less consistent estimates than the Rollback method. We also explored the choice of basis functions and found that Hermite polynomials of the third degree are quite effective, as is regressing on the underlying Swap price. Power series and Laguerre polynomials also work well, but more than three degrees of freedom seem to lead to overfitting. Regressing on the underlying stochastic processes $z_d, z_f, \ln(x)$ offers no improvement in predictions. In related literature we observe similar implementation with power series polynomials in [12] and [41], so our result is not surprising in this regard. In the cross-currency case we performed a similar analysis but quickly realized that we had discovered an interesting edge case for a deep-out-of-the-money option. We demonstrated how regression error created by data with excess zeros dominates the convergence of the Monte Carlo method. We implemented the Double Hurdle regression method for dealing with this but found it could not overcome the almost random relationship between the current swap value and non-zero continuation values. Dealing with excess zeros in data is a topic that has been studied heavily, so we recommend it as a future area of research in this context. We also observed that this leads to a slower rate of convergence, $\left(\frac{1}{N}\right)^{\frac{1}{4}}$, than is typically observed for Monte Carlo methods [43]. Once this error is compensated for by large numbers of simulations we do observe more standard convergence results. Such convergence analysis was not conducted in the relevant papers [49] and [48] that we mentioned in this section. The method will also be slower than the single-currency case due to the more involved simulations. Ultimately all three methods we implemented proved to be quite accurate typically offering accuracy within 5 basis points or less. Computation speeds could be improved by implementation in more efficient languages, though they already show promise by typically only requiring a few seconds to run.

Appendix A

Technical Proofs and Definitions

A.1 Single Currency LGM model

A.1.1 LGM Zero Coupon Bond

$$\begin{aligned}P(t,T) =&\mathbb{E}\left[e^{-\int_t^T r(s)ds}\middle|\mathcal{F}_t\right] = \mathbb{E}\left[e^{-(\int_0^T r(s)ds-\int_0^t r(s)ds)}\middle|\mathcal{F}_t\right] = \mathbb{E}\left[e^{-(\int_0^T \mu(s)ds-\int_0^t \mu(s)ds)-\int_t^T x_s ds}\middle|\mathcal{F}_t\right]\\ =&\mathbb{E}\left[\exp\left(\ln(P(0,T)) - \frac{1}{2}\int_0^T (H_T-H_s)^2\alpha_s^2 ds - \ln(P(0,t)) + \frac{1}{2}\int_0^t (H_t-H_s)^2\alpha_s^2\right.\right.\\ &\left.\left.-\int_t^T x_s ds\right)\middle|\mathcal{F}_t\right]\end{aligned}$$

Since x_t is a function of Itô integrals, it follows a normal distribution, since we are conditioning on time t the solution to the OU process now has an initial term such that

$$\begin{aligned}x(T) &= x_t e^{-(\beta(T)-\beta(t))} + e^{-(\beta(T)-\beta(t))}\int_t^T e_s^\beta \sigma_s dW_s\\ \mathbb{V}\left[x(T)|\mathcal{F}_t\right] &= e^{-2(\beta(T)-\beta(t))}\int_t^T \alpha_s^2 ds.\end{aligned}$$

Upon inserting this back into the original equation, recalling that $H(t) = \int_0^t e^{-\beta(s)}ds$, and using the Gaussian Moment Generating Function on the Itô Integral we get

$$P(t,T) = \frac{P(0,T)}{P(0,t)}\exp\left(-(H_T-H_t)e^{\beta_t}x_t - \frac{1}{2}\int_0^t \left[(H_T-H_s)^2-(H_t-H_s)^2\right]\alpha_s^2 ds\right).$$

[12]

A.1.2 LGM Drift and Variance

We first recall that the value of the Zero-Coupon Bond is given by the relation

$$P(0,t) = \mathbb{E}\left[e^{-\int_0^t r_s ds}\right]$$

Now following the steps mentioned to derive the LGM model we find

$$\begin{aligned} P(0,t) &= \mathbb{E}\left[e^{-\int_0^t \mu_s + x_s ds}\right] \\ \int_0^t x_s ds &= \int_0^t e^{-\beta_s} \int_0^s e^{\beta_u} \sigma_u dW_u ds \\ &= \int_0^t \int_0^s e^{-\beta_s} \alpha_u dW_u ds \\ &= \int_0^t \int_u^t e^{-\beta_s} \alpha_u ds dW_u \\ &= \int_0^t \alpha_u (H(t) - H(u)) dW_u \end{aligned}$$

Since this is an Itô Integral, this follows a normal distribution such that

$$\begin{aligned} \mathbb{E}\left[\int_0^t \alpha_u (H(t) - H(u)) dW_u\right] &= 0 \\ \mathbb{E}\left[(\int_0^t \alpha_u (H(t) - H(u)) dW_u)^2\right] &= \int_0^t (\alpha_u (H(t) - H(u)))^2 du \end{aligned}$$

Where we use the Îto Isometry in the second line. So we can make use of the Gaussian Moment Generating Function to find an expression for the Zero Coupon Bond in terms of the drift term.

$$\begin{aligned} P(0,t) &= e^{-\int_0^t \mu_s ds} e^{\frac{1}{2}\int_0^t \alpha(s)^2 (H(t) - H(s))^2 ds} \\ \int_0^t \mu(s) ds &= -\ln(P(0,t)) + \frac{1}{2}\int_0^t \alpha(s)^2 (H(t) - H(s))^2 ds \end{aligned}$$

By identifying x_t as an Ornstein Uhlenbeck process we have that

$$\begin{aligned} x_t &= e^{-\beta_t} \int_0^t \alpha_s dW_s \\ \mathbb{V}[r_t] = \mathbb{V}[x_t] &= e^{-2\beta_t} \int_0^t \alpha_s^2 ds \end{aligned}$$

due to the Itô Isometry. [12]

A.1.3 LGM Change of Numeraire

Here we make use of the toolkit described in [16, Section 2.3]. In particular, given two SDEs with different numeraires S and U and dynamics

$$\begin{aligned} dX_t &= \mu_t^S(X_t) dt + \sigma_t(X_t) C dW_t^S \\ dX_t &= \mu_t^U(X_t) dt + \sigma_t(X_t) C dW_t^U \end{aligned}$$

then the Radon Nikodym derivative is given as

$$\begin{aligned} \frac{dQ^S}{dQ^U} = \exp\Big(&-\frac{1}{2}\int_0^t ((\sigma_s(X_s)C)^{-1}[\mu_s^S(X_s) - \mu_s^U(X_s)])^2 ds \\ &+ \int_0^t (\sigma_s(X_s)C)^{-1}[\mu_s^S(X_s) - \mu_s^U(X_s)]) dW_s^U\Big) \end{aligned}$$

In our case

$$\begin{aligned} \mu_t^U &= 0 \\ \sigma_t &= \alpha(t) \\ \mu_t^S &= -H(t)\alpha^2(t) \end{aligned}$$

Upon substitution to this equation we find the Radon-Nikodym derivative in [12].

$$\frac{dQ^S}{dQ^U} = \exp\left(\int_0^t H_s \alpha_s dW_s - \frac{1}{2}\int_0^t H_s^2 \alpha_s^2 ds\right)$$

A.2 Distributions for Cross-Currency LGM

A.2.1 Drift Conditions

We follow the steps in [12] closely and define

$$Y(t) := \frac{N_f(t)x(t)}{N_d(t)}$$

and require it to be a martingale. Since $N_d(t)$ and $N_f(t)$ are functions of $z_d(t)$ and $z_f(t)$ respectively, then we can apply Itô's lemma directly, which gives the following SDEs.

$$\begin{aligned}
\frac{dN_d(t)}{N_d(t)} &= \left(r_d(t) + H_d(t)^2\alpha_d(t)^2\right)dt + H_d(t)\alpha_d(t)dW_d(t) \\
\frac{dN_f(t)}{N_f(t)} &= \left(r_f(t) + H_f(t)^2\alpha_f(t)^2 + H_f(t)\gamma_f(t)\right)dt + H_f(t)\alpha_f(t)dW_f(t)
\end{aligned}$$

We can now apply Itô's lemma to our expression for $Y(t)$ as a function of $N_f(t), N_d(t), x(t)$.

$$dY(t) = \frac{\partial Y}{\partial N_f}dN_f + \frac{\partial Y}{\partial x}dx + \frac{\partial Y}{\partial N_d}dN_d + \frac{1}{2}\frac{\partial^2 Y}{\partial N_f^2}(dN_f)^2 + \frac{\partial^2 Y}{\partial N_f \partial N_d}dN_f dN_d + \frac{\partial^2 Y}{\partial x \partial N_d}dx dN_d + \frac{\partial^2 Y}{\partial N_f \partial x}dN_f dx$$

Where some of the second derivatives that are automatically 0 have been excluded, simplifying this we get

$$\begin{aligned}
dY(t) = Y(t)[\,&\left(r_f(t) - r_d(t) + H_f^2(t)\alpha_f^2(t) + \mu_x(t) + H_f(t)\gamma_f(t)\right. \\
&+ H_f(t)\alpha_f(t)\sigma_x\rho_{fx} - H_d(t)\alpha_d(t)\sigma_x\rho_{dx} \\
&\left.-H_f(t)\alpha_f(t)H_d(t)\alpha_d(t)\rho_{df}\right)dt \\
&-H_d(t)\alpha_d(t)dW_d(t) + H_d(t)\alpha_d(t)dW_f(t) + \sigma_x dW_x(t)]\,.
\end{aligned}$$

By setting the drift equal to 0 we find our first condition

$$\begin{aligned}
\mu_x(t) =& r_d(t) - r_f(t) - H_f^2(t)\alpha_f^2(t) - H_f(t)\alpha_f(t)\sigma_x\rho_{fx} + H_d(t)\alpha_d(t)\sigma_x\rho_{dx} \\
&+ H_f(t)\alpha_f(t)H_d(t)\alpha_d(t)\rho_{df} - H_f(t)\gamma_f(t).
\end{aligned}$$

Fortunately, we can simplify this once we solve for the second drift expression $\gamma_f(t)$. We start by rewriting our expression for $U(t)$

$$U(t) = \frac{P_f(t,T)x(t)}{N_d(t)} = \tilde{P}_f(t,T)\frac{N_f(t)x(t)}{N_d(t)} = \tilde{P}_f(t,T)Y(t)$$

Where $\tilde{P}_f(t,T)$ is the foreign bond divided by the foreign numeraire. Its expression is the same as we previously derived in the single currency case

$$\tilde{P}_f(t,T) = \frac{P(t,T)}{N_f(t)} = P_f(0,T)\exp\left(-H_f(T)z_f(t) - \frac{1}{2}H_f^2(t)\xi(t)\right).$$

This is again a function of z_f and t, so we can apply Itô's lemma directly to $U(t)$, which ultimately gives

$$\begin{aligned}
\frac{dU(t)}{U(t)} &= H_f(t)\left(-\gamma_f(t) - H_f(t)\alpha_f^2(t) + H_d(t)\alpha_d(t)\alpha_f(t)\rho_{df} - \sigma_x(t)\alpha_f(t)\rho_{xf}\right)dt \\
&- H_d(t)\alpha_d(t)sW_d(t) - (H_f(T) - H_f(t))\alpha_f(t)dW_f(t) + \sigma_x(t)dW_x(t).
\end{aligned}$$

Setting the drift to be 0 leads us to the desired condition on $\gamma_f(t)$

$$\gamma_f(t) = -H_f(t)\alpha_f^2(t) + H_d(t)\alpha_d(t)\alpha_f(t)\rho_{df} - \sigma_x(t)\alpha_f(t)\rho_{xf}.$$

Once we substitute this back into our expression for $\mu(t)$ we find a simplified version of the condition as

$$\mu_x(t) = r_d(t) - r_f(t) + H_d(t)\alpha_d(t)\sigma_x(t)\rho_{dx}. \tag{A.2.1}$$

A.2.2 Variance of Foreign Process

$$\begin{aligned}
\texttt{Var}(z_f(t)) &= \mathbb{E}\left[\left(\int_0^t \gamma_f(s)ds - \int_0^t \alpha_f(s)dW_f(s)\right)^2\right] - \mathbb{E}\left[(\int_0^t \gamma_f(s)ds)^2\right] \\
&= \mathbb{E}\left[\left(\int_0^t \gamma_f(s)ds\right)^2\right] - 2\mathbb{E}\left[\int_0^t \gamma_f(s)ds\right]\mathbb{E}\left[\int_0^t \alpha_f(s)dW_f(s)\right] + \mathbb{E}\left[\left(\int_0^t \alpha_f(s)dW_f(s)\right)^2\right] \\
&\quad - \mathbb{E}\left[(\int_0^t \gamma_f(s)ds)^2\right] \\
&= \mathbb{E}\left[\int_0^t \alpha_f(s)^2 ds\right]
\end{aligned}$$

[12]

A.2.3 FX Process Expression

Since $x(t)$ is a geometric Brownian motion its solution will have the form

$$x(t) = x(0)\exp\left(\int_0^t \mu_x(s)ds - \frac{1}{2}\int_0^t \sigma_x^2(s)ds + \int_0^t \sigma_x(s)dW_x(s)\right).$$

We have a formula for $\mu_x(s)$ which depends on $r_d(s)$ and $r_f(s)$ in equation (A.2.1), from [12] these have the following forms

$$\begin{aligned}
r_d(t) &= f_d(0,t) + z_d(t)H_d'(t) + \xi_d(t)H_d'(t)H_d(t) \\
r_f(t) &= f_f(0,t) + z_f(t)H_f'(t) + \xi_f(t)H_f'(t)H_f(t)
\end{aligned}$$

Substituting this into the expression for μ, we get

$$\begin{aligned}
\ln(x(t)) =& \ln(e^{\int_0^t f_d(0,s) - f_f(0,s)ds}) + \int_0^t z_d(s)H_d'(s) + \xi_d(s)H_d'(s)H_d(s) \\
&- z_f(s)H_f'(s) - \xi_f(s)H_f(s)H_f'(s)ds + \rho_{dx}\int_0^t \alpha_d(s)H_d(s)ds - \frac{1}{2}\int_0^t \sigma_x^2(s)ds \\
&+ \int_0^t \sigma_x(s)dW_x(s) \\
=& \ln\left(\frac{P_f(0,t)}{P_d(0,t)}\right) + (H_d(t) - H_d(0))z_d(0) + \int_0^t (H_d(t) - H_d(s))dz_d(s) \\
&+ \int_0^t \xi_d(s)H_d'(s)H_d(s)ds - \int_0^t \xi_f(s)H_f(s)H_f'(s)ds \\
&- z_f(0)(H_f(t) - H_f(0)) - \int_0^t (H_f(t) - H_f(s))dz_f(s) \\
&+ \rho_{xd}\int_0^t \alpha_d(s)H_d(s)\sigma_f(s)ds - \frac{1}{2}\int_0^t \sigma_x^2(s)ds + \int_0^t \sigma_x(s)dW_x(s)
\end{aligned}$$

Applying integration by parts

$$\begin{aligned}
=& \ln\left(\frac{P_f(0,t)}{P_d(0,t)}\right) + (H_d(t) - H_d(0))z_d(0) + \int_0^t (H_d(t) - H_d(s))\alpha_d(s)dW_d(s) \\
&+ \int_0^t \xi_d(s)H_d'(s)H_d(s)ds - z_f(0)(H_f(t) - H_f(0)) - \int_0^t (H_f(t) - H_f(s))\gamma_f(s)ds \\
&- \int_0^t (H_f(t) - H_f(s))\alpha_f(s)dW_f(s) + \rho_{xd}\int_0^t \alpha_d(s)H_d(s)\sigma_x(s)ds - \frac{1}{2}\int_0^t \sigma_x^2(s)ds \\
&+ \int_0^t \sigma_x(s)dW_x(s) - \int_0^t \xi_f(s)H_f(s)H_f'(s)ds
\end{aligned}$$

[12].

A.2.4 Martingale Inaccuracies over time

In [12] this inaccuracy is explained in the context of the Zero Coupon Bond martingale by first observing

$$\mathbb{E}\left[\frac{1}{N(t)}\right] = P(0,t)$$

So rearranging this and writing the expectation analytically we have

$$\begin{aligned}
\frac{1}{P(0,t)}\mathbb{E}\left[\frac{1}{N(t)}\right] &= \mathbb{E}\left[\exp\left(-H_t z_t - \frac{1}{2}H_t^2\xi_t\right)\right] \\
&= \frac{1}{\sqrt{2\pi\xi_t}}\int_{-\infty}^{\infty}\exp\left(-H_t z - \frac{1}{2}H_t^2\xi_t - \frac{z^2}{2\xi_t}\right)dz \\
&= \frac{1}{\sqrt{2\pi\xi_t}}\int_{-\infty}^{\infty}\exp\left(-\frac{(z+H_t\xi_t)^2}{2\xi_t}\right)dz \\
&= \frac{1}{\sqrt{2\pi}}\int_{-\infty}^{\infty}\exp\left(-\frac{(x+H_t\sqrt{\xi_t})^2}{2}\right)dx \\
&= 1 \quad [12]
\end{aligned}$$

Where at the second last line a change of variables was performed such that $x = \frac{z}{\sqrt{\xi_t}}$. It is theorized that the driving factor behind the calculation of the expectation is less the standard normal random variable x, and more so the quantity $-H_t\sqrt{\xi_t}$. It is shown empirically in [12] that this can take values as large as 3 or 4, which moves the sampling away from the relevant region which should be centered around 0. In order to rectify this it is possible to apply a shift to $H(t)$. The LGM model has a "Shift Invariance" Property which we have not mentioned previously for the sake of relevance. So we can apply a shift of $-H(T)$ to $H(t)$ which can solve the problem, but the size and direction of the shift should be treated on a case-by-case basis according to [12]. Similar steps would have to be repeated for each of the martingales we outlined previously.

A.3 Monte Carlo

A.3.1 Polynomial Definitions

Standard Power series polynomials of degree n, $P_n(x)$, are defined

$$P_n(x) = \Sigma_{k=0}^{n} x^k.$$

Hermite Polynomials of degree n, $H_n(x)$, are defined by the following formula

$$H_n(x) = (-1)^n e^{x^2}\frac{d^n}{dx^n}e^{-x^2}.$$

Laguerre Polynomials of degree n, $L_n(x)$, are defined by a recursive formulation.

$$\begin{aligned}
L_0(x) &= 1 \\
L_1(x) &= 1 - x \\
L_{k+1}(x) &= \frac{(2k+1-x)L_k(x) - kL_{k-1}(x)}{k+1}.
\end{aligned}$$

Appendix B

Additional Results and Plots

B.1 Sinlge Currency

B.1.1 Rollback

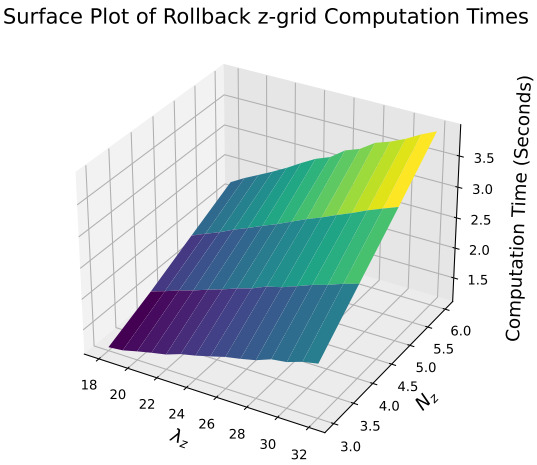

Figure B.1: Rollback Computation Time Surface over z-Grid

B.1.2 Monte Carlo Regressions

Table B.1: Single Currency z-Process Regression Statistics

Polynomial Degree	1	2	3	4	5	6
RMSE	436937.25	354154.05	325689.55	317422.58	320087.16	320454.17
R^2	77.76%	83.59%	83.54%	83.78%	83.36%	83.52%

B.2 Cross Currency

B.2.1 Convergence Plot

We plot the convergence of our Monte Carlo estimate 5 times with the number of regression data points fixed at 1 million.

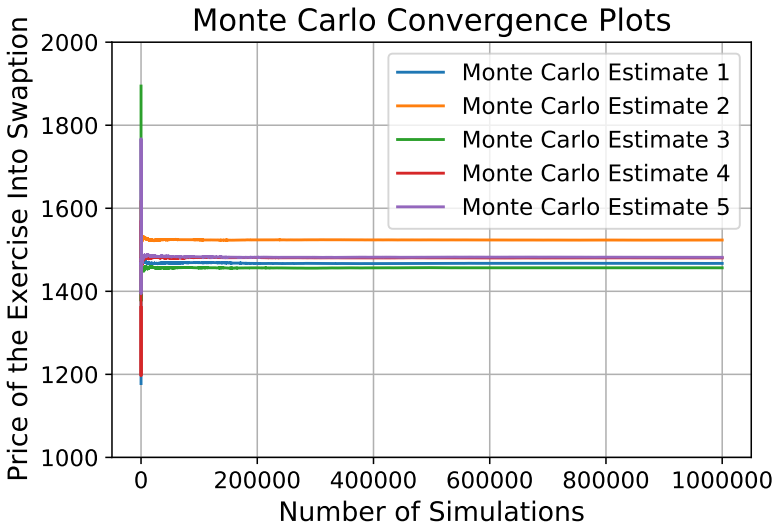

Figure B.2: Spurious Monte Carlo Convergence 1 Million Simulations

Bibliography

[1] Patrick Hagan. Methodology for callable swaps and bermudan " exercise into " swaptions. 01 2004.

[2] Francis A. Longstaff and Eduardo S. Schwartz. Valuing american options by simulation: A simple least-squares approach. *Review of Financial Studies*, 14, 2001.

[3] Nicholas Burgess. Cross currency swap theory & practice - an illustrated step-by-step guide of how to price cross currency swaps and calculate the basis spread. *SSRN Electronic Journal*, 2018.

[4] John G. Cragg. Some statistical models for limited dependent variables with application to the demand for durable goods. *Econometrica*, 39, 1971.

[5] Bank for International Settlements Monetary and Economic Department. Triennial central bank survey otc foreign exchange turnover in april 2022. 2022.

[6] Andreas Schrimpf and Vladyslav Sushko. Beyond libor: a primer on the new reference rates. *BIS Quarterly Review*, 2019.

[7] Duy Minh Dang, Christina C. Christara, Kenneth R. Jackson, and Asif Lakhany. A pde pricing framework for cross-currency interest rate derivatives. volume 1, 2010.

[8] Dariusz Gatarek and Juliusz Jabłecki. Between scylla and charybdis: The bermudan swaptions pricing odyssey. *Mathematics*, 9, 2021.

[9] Milind Sharma and Jonathan Stein. Valuation of cancelable cross currency bermudan swaps. *SSRN Electronic Journal*, 2011.

[10] Ahsan Amin. Multi-factor cross currency libor market models: Implementation, calibration and examples. *SSRN Electronic Journal*, 2011.

[11] Raoul Pietersz. Pricing models for bermudan-style interest rate derivatives. 2005.

[12] Roland Lichters, Roland Stamm, and Donal Gallagher. *Modern Derivatives Pricing and Credit Exposure Analysis*. 2015.

[13] Simon H Babbs. The valuation of cross-currency interest-sensitive claims with application to "diff" swaps. *Applications in Finance, Investments, and Banking*, pages 285–333, 1999.

[14] Jori Hoencamp, Shashi Jain, and Drona Kandhai. A semi-static replication approach to efficient hedging and pricing of callable ir derivatives. 2022.

[15] Attilio Meucci. " p " versus " q ": Differences and commonalities between the two areas of quantitative finance. *Ssrn*, 2011.

[16] D Brigo and F Mercurio. *Interest rate models-theory and practice: with smile, inflation and credit*, volume 82. 2006.

[17] Tomas Björk. *Arbitrage Theory in Continuous Time*. 2005.

[18] Ferdinando Ametrano and Marco Bianchetti. Everything you always wanted to know about multiple interest rate curve bootstrapping but were afraid to ask. *SSRN Electronic Journal*, 2013.

[19] G Grimmett and D Stirzaker. *Probability and random processes (3rd Edition)*. 2001.

[20] Nigel J. Cutland and Alet Roux. *Derivative Pricing in Discrete Time*, volume 79. 2013.

[21] Andrea Pascucci. Pde and martingale methods in option pricing. *Bocconi and Springer Series*, 2, 2011.

[22] Fischer Black and Myron Scholes. The pricing of options and corporate liabilities. *Journal of political economy*, 81(3):637–654, 1973.

[23] J.Michael Harrison and David M Kreps. Martingales and arbitrage in multiperiod securities markets. *Journal of Economic Theory*, 20(3):381–408, 1979.

[24] Nicolas Gisiger. Risk-neutral probabilities explained. *SSRN Electronic Journal*, 2011.

[25] Rene L. Schilling. *Measures, integrals and martingales*. 2005.

[26] S. D. Jacka and Bernt Oksendal. Stochastic differential equations: An introduction with applications. *Journal of the American Statistical Association*, 82, 1987.

[27] Ioannis Karatzas and Steven E. Shreve. *Brownian motion and stochastic calculus*, volume 113. Springer Science & Business Media, 1991.

[28] Paul Wilmott. *Paul Wilmott Introduces Quantitative Finance*, volume 8. 2002.

[29] John Karon, Richard W. Hamming, William S. Dorn, Daniel D. McCracken, and A. H. Stroud. Introduction to applied numerical analysis. *The American Mathematical Monthly*, 84, 1977.

[30] Reuven Y. Rubinstein and Dirk P. Kroese. *Simulation and the Monte Carlo Method: Third Edition*. 2016.

[31] Terry Lyons and Nicolas Victoir. Cubature on wiener space. *Proceedings: Mathematical, Physical and Engineering Sciences*, 460(2041):169–198, 2004.

[32] Phelim P. Boyle. Options: A monte carlo approach. *Journal of Financial Economics*, 4, 1977.

[33] Komputasi Greeks, Finite Difference, Monte Carlo, and Reduksi Variansi. Computing greeks by finite difference using monte carlo simulation and variance reduction techniques. *Bimipa*, 28, 2018.

[34] Morten Bjerregaard Pedersen. Bermudan swaptions in the libor market model. *SSRN Electronic Journal*, 2005.

[35] Tomohisa Yamakami and Yuki Takeuchi. Pricing bermudan swaption under two factor hull-white model with fast gauss transform, 2022.

[36] F.M. Dekking, C. Kraaikamp, H.P. Lopuhaä, and L.E. Meester. *A Modern Introduction to Probability and Statistics: Understanding Why and How*. Springer Texts in Statistics. Springer London, 2006.

[37] E. S. Page, Yu. A. Shreider, and G. J. Tee. The monte carlo method: The method of statistical trials. *Journal of the Royal Statistical Society. Series A (General)*, 130, 1967.

[38] Kenneth N. Berk Jay L. Devore. *Modern Mathematical Statistics with Applications - 2nd Edition*, volume 64. 2012.

[39] Christian P. Robert and George Casella. *Monte Carlo Statistical Methods*. Springer, 2004.

[40] Matthias Leclerc, Qian Liang, and Ingo Schneider. Fast monte carlo bermudan greeks. *Risk*, 22(7):84, 2009.

[41] Stefanus C. Maree and Jacques Du Toit. Pricing bermudan swaptions on the libor market model using the stochastic grid bundling method. 2015.

[42] Paolo Brandimarte. *Numerical Methods in Finance and Economics: A MATLAB-Based Introduction: Second Edition*. 2006.

[43] William H Press. *Numerical recipes in Fortran 77: the art of scientific computing*. Cambridge university press, 1992.

[44] A. Plucińska. A stochastic characterization of hermite polynomials. *Journal of Mathematical Sciences*, 89, 1998.

[45] Qing Liu and Donald A. Pierce. A note on gauss-hermite quadrature, 1994.

[46] Stephen M. Meyer and Michale S. Lewis-Beck. Applied regression: An introduction. *Journal of Policy Analysis and Management*, 1, 1981.

[47] Vicki A. Barbur, D. C. Montgomery, and E. A. Peck. Introduction to linear regression analysis. *The Statistician*, 43, 1994.

[48] Xisheng Yu and Qiang Liu. Canonical least-squares monte carlo valuation of american options: Convergence and empirical pricing analysis. *Mathematical Problems in Engineering*, 2014:13, 2014.

[49] Zhemin Wu. Pricing american options using monte carlo method, 2012.

[50] Roderick JA Little and Donald B Rubin. *Statistical Analysis with Missing Data*. Wiley, 2002.

[51] Daniel J. Duffy. *Finite Difference Methods in Financial Engineering: A Partial Differential Equation Approach*. John Wiley & Sons, 2006.

[52] Kenneth L Judd. *Numerical Methods in Economics*, volume 1. 1998.