Volatility Models in Option Pricing

Miguel Ângelo Maia Ribeiro

Thesis to obtain the Master of Science Degree in

Engineering Physics

Supervisors: Prof. Cláudia Rita Ribeiro Coelho Nunes Philippart
Prof. Rui Manuel Agostinho Dilão

Examination Committee

Chairperson: Prof. Full Name
Supervisor: Prof. Full Name 1 (or 2)
Member of the Committee: Prof. Full Name 3

Month Year

To my parents and sister

Acknowledgments

A few words about the university, financial support, research advisor, dissertation readers, faculty or other professors, lab mates, other friends and family...

Resumo

Inserir o resumo em Português aqui com o máximo de 250 palavras e acompanhado de 4 a 6 palavras-chave...

Palavras-chave: palavra-chave1, palavra-chave2,...

Abstract

Insert your abstract here with a maximum of 250 words, followed by 4 to 6 keywords...

Keywords: keyword1, keyword2,...

Contents

List of Tables

List of Figures

Nomenclature

Greek symbols

α Angle of attack.

β Angle of side-slip.

κ Thermal conductivity coefficient.

μ Molecular viscosity coefficient.

ρ Density.

Roman symbols

C_D Coefficient of drag.

C_L Coefficient of lift.

C_M Coefficient of moment.

p Pressure.

$\mathbf{u}$ Velocity vector.

u, v, w Velocity Cartesian components.

Subscripts

∞ Free-stream condition.

i, j, k Computational indexes.

n Normal component.

x, y, z Cartesian components.

ref Reference condition.

Superscripts

* Adjoint.

T Transpose.

Glossary

OTC Over-the-Counter market refers to all deals signed outside of exchanges.

Chapter 1

Introduction

1.1 Mathematical Finance

Mathematical finance, also known as quantitative finance, is a field of applied mathematics focused on the modeling of financial instruments. It is rather difficult to overestimate its importance since it is heavily used by investors and investment banks in everyday transactions. In recent decades, this field suffered a complete paradigm shift, following developments in computer science and new theoretical results that enabled investors to better understand the mechanics of financial markets.

With the colossal sums traded daily in financial markets around the world, mathematical finance has become increasingly important and many resources are invested in the research and development of new and better theories and algorithms.

1.2 Derivatives

Derivatives are currently one of the subjects most studied by financial mathematicians. In finance, a *derivative* is simply a contract whose value depends on other simpler financial instruments, known as *underlying assets*, such as stock prices or interest rates. They can virtually take any form desirable, so long as there are two parties interested in signing it and all government regulations are met.

The importance of derivatives has grown greatly in recent years. In fact, as of June 2017, derivatives were responsible for over $542 trillion worth of trades, in the Over-the-Counter (OTC) market alone [1], as can be seen in Figure 1.1 (the OTC market refers to all deals signed outside of exchanges). This growth peaked in 2008 but stalled after the global financial crisis due to new government regulations, implemented because of the role of derivatives in market crashes [2]. It is easy to see that mishandling derivatives can have disastrous consequences. However, when handled appropriately, derivatives prove to be very powerful tools to investors, as we will see shortly.

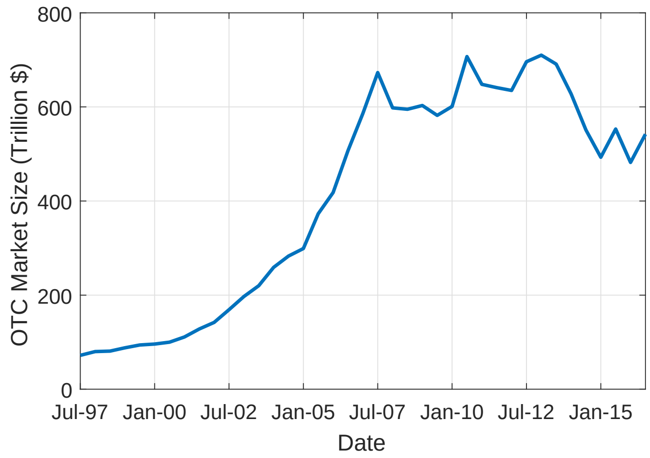


Figure 1.1: Size of OTC derivatives market since May 1996.

1.3 Options

Of all classes of derivatives, in this master thesis we will focus particularly on the most traded type [3]: *options*.

As the name implies, an *option* contract grants its buyer the *option* to buy (in the case of a *call* type option) or sell (for *put* options) its underlying asset at a future date, known as the *maturity*, for a fixed price, known as the *strike price*. In other words, when signing an option, buyers choose a price at which they want to buy/sell (call/put) some asset and a future date to do this transaction. When this date arrives, if the transaction is favorable to the buyers, they exercise their right to execute it.

The description above pertains only to *European* options. In this thesis, this type of contracts will be used for model calibration and validation. Other option types will also be considered, however. We shall approach *American* options, contracts that enable their buyers to exercise their right to buy/sell the underlying asset at any point in time *until* the maturity date. Other less common types, commonly known as *Exotic* options, will also be studied in some detail in the following sections.

It's important to emphasize the fact that an option grants its buyer the right to do something. If *exercising* the option would lead to losses, the buyer can simply decide to let the maturity date pass, allowing the option to expire without further costs. This is indeed the most attractive characteristic of options.

1.3.1 Why Options are Important

Options are very useful tools to all types of investors.

To hedgers (i.e. investors that want to limit their exposure to risk), options provide safety by fixing a minimum future price on their underlying assets - e.g. if hedgers want to protect themselves against a potential future price crash affecting one of their assets, they can buy put type options on that asset. With these, even if the asset's value does crash, their losses will always be contained because they can exercise the options and sell the asset at the option's higher strike price.

Options are also very useful to speculators (i.e. investors that try to predict future market movements). The lower price of options (when compared to their underlying assets) grants this type of investors great leveraging capabilities and, with them, access to much higher profits if their predictions prove true. The opposite is also true and a wrong prediction can equally lead to much greater losses.

Due to all their advantages, and unlike some other types of derivatives, options have a price. Finding the ideal price for an option is a fundamental concern to investors, because knowing their appropriate value can give them a chance to take advantage of under or overpriced options. Finding this price can be very difficult for some option types, however, and though a lot of research has been done towards this goal, a great deal more is still required.

Chapter 2

Background

2.1 Option Types

Before completely focusing on the mechanics of options, what influences their prices and how we can predict their behavior, we should begin by clearly defining the main option types, their characteristics, as well as their payoff functions. We will not only approach the two main types of options - European and American - but also other less common types, commonly referred to as Exotic options, such as Asian, Barrier and Binary options.

2.1.1 European Options

European options are the most traded type of option in the OTC market [4]. They are not only extremely useful to investors, but also very simple to study and comparatively easy to price. For all these reasons, they have been the subject of much research and are deeply understood. Furthermore, because of their high availability, they are very useful in model calibration and validation.

As stated before, call and put European options enable their buyers to respectively buy and sell the underlying asset *at the maturity* for the fixed strike price.

To understand the payoff function of such contracts, we'll use an example. In the case of a European call option, if at the maturity the market price of its underlying asset is greater than the strike, investors can exercise the option and buy the asset for the fixed lower strike price. They can then immediately go to the market and sell the asset for its higher value. Thus, in this case, the payoff of the option would be the difference between the asset's price and the option's strike price. On the other hand, if at the maturity the price of the asset decreases past the strike, the investor should let the option expire, since the asset is available in the market for a lower price. In this case, the payoff would be zero. The same reasoning can be made for European put type options, such that the payoff function of both option types can then be deduced as

$$\begin{aligned} \text{Payoff}_{Euro,\ call}(K,T) &= \max\left(S(T)-K,0\right); \\ \text{Payoff}_{Euro,\ put}(K,T) &= \max\left(K-S(T),0\right), \end{aligned} \tag{2.1}$$

where K is the option's strike price and $S(T)$ is the asset's price, $S(t)$, at the maturity, T. These functions are represented in Figure 2.1.

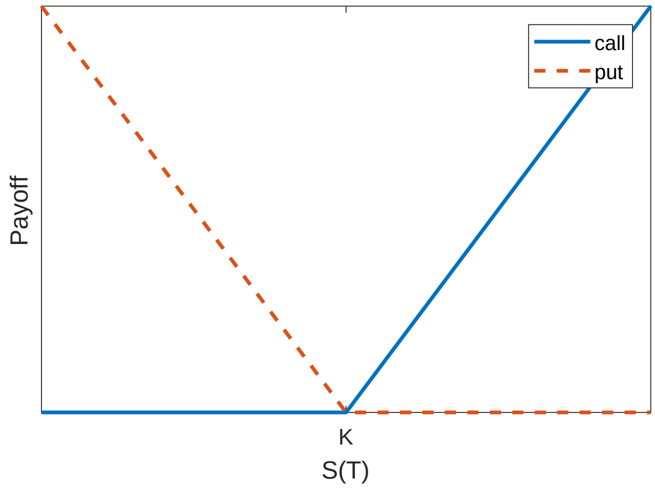


Figure 2.1: Payoff functions of European *call* and *put options*.

2.1.2 American Options

American options are more complex than European and thus harder to price. While European options dominate the OTC market, their American counterparts are the most traded type of option in exchanges [5]. Because of their great importance, many models have been developed to find the prices of these options [6].

American options grant the right to buy/sell (call/put) the underlying asset at any point in time *until the maturity date*. Following the logic used in the previous example to find the payoff functions of European calls and puts, we can deduce their American counterparts as

$$\begin{aligned} \text{Payoff}_{Amer,\ call}(K, t^*) &= \max\left(S(t^*) - K, 0\right); \\ \text{Payoff}_{Amer,\ put}(K, t^*) &= \max\left(K - S(t^*), 0\right), \end{aligned} \tag{2.2}$$

where we now define t^* (with $0 \leq t^* \leq T$) as the exercise date.

It should be obvious that the price of American options will always be greater or equal to the prices of similar European options. The reason behind this is the fact that, with European contracts our exercise decision is restricted to a single day, whereas with American options we have several other opportunities to make this choice.

2.1.3 Exotic Options

While European and American options are, by far, the most traded types, *Exotic options* should not be neglected. Not only does there exist a great number of Exotic option types, but these are also highly customizable, making this type of derivatives very useful for unconventional investment strategies. Due to their high complexity, these options are only traded in the OTC market, and not in exchanges [7]. We will explore three of the most common types of Exotic options, though many others exist.

Asian Options

An *Asian* option is an Exotic contract whose payoff depends on the *average value* of an underlying asset. This averaging procedure makes this type of derivative more stable than their European/American equivalents - e.g. if, at the last day of trading, an extreme market event occurs, affecting the price of the underlying asset, an European or American option might suffer radical changes to its value (either beneficial or detrimental), whereas the value of an Asian option should be much less affected by such sudden movements.

The payoff function of this type of contracts is given by

$$\begin{aligned} \text{Payoff}_{Asian,\ call}(K,T) &= \max\left(A(S,T) - K, 0\right); \\ \text{Payoff}_{Asian,\ put}(K,T) &= \max\left(K - A(S,T), 0\right), \end{aligned} \tag{2.3}$$

with $A(S,T)$ corresponding to the arithmetic average of the asset's prices until the maturity,

$$A(S,T) = \frac{1}{T}\int_0^T S(t)dt. \tag{2.4}$$

Other types of Asian option exist: some contracts have a floating strike (i.e. instead of the fixed strike, K, the strike is assumed as the average, $A(S,T)$, instead), and others use different types of averaging mechanisms, such as the geometric average. Despite this, we will assume that all Asian options follow the properties described in eqs.(2.3) and (2.4).

Barrier Options

A *Barrier option* behaves similarly to a European option with the difference that it only becomes active (or void) if the value of its underlying asset reaches a particular value, called the *barrier level*, B, at any point in time until the option's maturity.

There are four main types of Barrier option:

- *up-and-out*: the asset's price starts below the barrier (i.e. $S(0) < B$). If it increases past this threshold, the option becomes *worthless*;
- *down-and-out*: the asset's price starts above the barrier (i.e. $S(0) > B$). If it decreases past this threshold, the option becomes *worthless*;
- *up-and-in*: the asset's price starts below the barrier (i.e. $S(0) < B$). Only if it increases past this threshold does the option become *active*;
- *down-and-in*: the asset's price starts above the barrier (i.e. $S(0) > B$). Only if it decreases past this threshold does the option become *active*.

Because all of the previously described Barrier option types are handled similarly, we can easily adapt the models from one type to another. Thus, for simplicity, we will henceforth assume that all Barrier options are of the up-and-in type.

Using the up-and-in Barrier option type as an example, if the asset price, $S(t)$, remains below the barrier level B throughout the whole option duration, even if at the maturity the asset's value is higher than the strike price, the option's payoff would nonetheless be zero. On the contrary, if this threshold was surpassed at any point in this period, the option's payoff would be similar to that of its European equivalent.

The payoff function of this type of option is given by

$$\begin{aligned} \text{Payoff}_{Barr,\ call}(K,T) &= \begin{cases} \max\left(S(T)-K,0\right), & \text{if } \ \exists\, t<T \ : \ S(t)>B \\ 0, & \text{otherwise} \end{cases}; \\ \text{Payoff}_{Barr,\ put}(K,T) &= \begin{cases} \max\left(K-S(T),0\right), & \text{if } \ \exists\, t<T \ : \ S(t)>B \\ 0, & \text{otherwise} \end{cases}. \end{aligned} \tag{2.5}$$

Binary Options

As the name implies, a *Binary option* is a contract that has one of two outcomes: if at the maturity date the asset price is above/below the strike (call/put), the investor receives a fixed amount of money, C. If this event does not occur, the investor receives nothing.

The main difference between a Binary and an European option is the fact that the payoff of a Binary contract is fixed, regardless of the asset price at the maturity, which is not true for European options.

The payoff of this function is, thus,

$$\begin{aligned} \text{Payoff}_{Bin,\ call}(K,T) &= \begin{cases} C, & \text{if } \ S(T)>K \\ 0, & \text{otherwise} \end{cases}; \\ \text{Payoff}_{Bin,\ put}(K,T) &= \begin{cases} C, & \text{if } \ S(T)<K \\ 0, & \text{otherwise} \end{cases}. \end{aligned} \tag{2.6}$$

Though all the option types described before are used by banks and investors everyday, we will mainly focus on European options, for the reasons mentioned. One further reason for this choice is the fact that the data available for model calibration and validation pertains only to this option type. The remaining types will nonetheless be implemented and studied, though no benchmark will be used to verify the models' validity in these cases.

2.2 Option Prices and Payoffs

It is important to emphasize the difference between an option's payoff and its profit for investors. Because options grant the right to buy/sell some asset, no investors would exercise an option if this action was disadvantageous to them (i.e. negative payoff value). Thus, the payoff of an option is always positive (it can also, obviously, be zero). This might sound like an arbitrage possibility (i.e. the chance of making profit without risk - which is illegal), but in reality options have a price that investors have to pay in order to acquire them. This means that even if the option's payoff is positive, if this value is lower than the price an investor paid to buy the option, that investor will actually lose money. The profit of an option is thus the difference between its payoff and its price, which can be negative. With this concept in mind, we can price options by setting their expected profit to be the same as a risk-neutral investment, (e.g. bank deposit). The price of an option can thus be deduced as it's expected future payoff, discounted back to the present

$$\text{Price}(K,t^*) = e^{-rt^*}\mathbb{E}\left[\text{Payoff}(K,t^*)\right], \tag{2.7}$$

where t^* denotes the time at which the option is exercised and r corresponds to the risk-free interest rate, which we will approach in Section 2.3.

As an example, we now present the price functions of European call and put options. With eqs.(2.1) in mind, it should be clear that the value of these two types of contracts is given by

$$\begin{aligned} C(K,T)_{Euro} &= e^{-rT}\mathbb{E}\left[\max\left(S(T)-K,0\right)\right] = e^{-rT}\mathbb{E}\left[\left(S(T)-K\right)\mathbb{1}_{\{S(T)>K\}}\right]; \\ P(K,T)_{Euro} &= e^{-rT}\mathbb{E}\left[\max\left(K-S(T),0\right)\right] = e^{-rT}\mathbb{E}\left[\left(K-S(T)\right)\mathbb{1}_{\{S(T)<K\}}\right], \end{aligned} \tag{2.8}$$

with $C(K,T)$ and $P(K,T)$ being the values of European call and put options, respectively, and $\mathbb{E}[\cdot]$, $\mathbb{1}_{\{\cdot\}}$ corresponding to the expected value and indicator functions, respectively.

When selling or buying options, investment banks add some premium to this zero-profit price, to account for the risk taken. Though this premium is important to define, it is besides the scope of this work and will not be considered here.

2.3 Black-Scholes Formulae

Due to their high importance, options have been studied in great detail in the past. Probably the most important result in this field came from Fischer Black, Myron Scholes and Robert Merton, who developed a mathematical model to price European options - the famous Black-Scholes (BS) model [8] - still in use in present days [9].

This model states that the price of an European call or put option follows the partial differential equation (PDE)

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0, \tag{2.9}$$

where V is the price of the option, S is the price of the underlying (risky) asset, r is the risk-free interest rate and σ is the stock price volatility. The underlying asset is commonly referred to as *stock*, so these terms will be used interchangeably in the following sections.

The risk-free interest rate, r, is the interest an investor would receive from any risk-free investment (e.g. treasury bills). No investor should ever invest in risky products whose expected return is lower than this interest (e.g. the lottery), since there's the alternative of obtaining a higher (expected) payoff without the disadvantage of taking risks. In general, this rate changes slightly with time and is unknown, but Black *et al.*, in their original model (eq.(2.9)), assumed that it remains constant throughout the option's duration and that it is known. Some authors have suggested solutions to deal with this shortcoming, providing models to replicate the behavior of interest rates [10], but because option prices do not significantly depend on this value [9], in the remainder of this thesis we shall make the same assumptions as Black *et al.* and set this rate to some constant.

As for the stock price volatility, σ, since we will explore it to great extent in the next sections, suffice it to say that it is a measure of the future stock price movement's uncertainty.

Some companies decide to grant their shareholders a part of the profits generated, known as *dividends*. This action decreases the company's total assets, which decreases the value of stocks, changing option prices. Because this occurrence is based on human decisions, it is extremely hard to model. Furthermore, Black *et al.* assumed in their models that no dividends were paid throughout the option's duration. For both these reasons, we will henceforth set dividend payment to zero.

One other important assumption of the BS model is that stock prices follow a stochastic process, known as Geometric Brownian Motion, which can be defined as

$$dS(t) = rS(t)dt + \sigma S(t)dW(t), \tag{2.10}$$

with $\{W(t),\ t > 0\}$ defining a one-dimensional Brownian motion and where we define $S_0 = S(0)$ as the stock price at time $t = 0$. An example of such processes is represented in Figure 2.2.

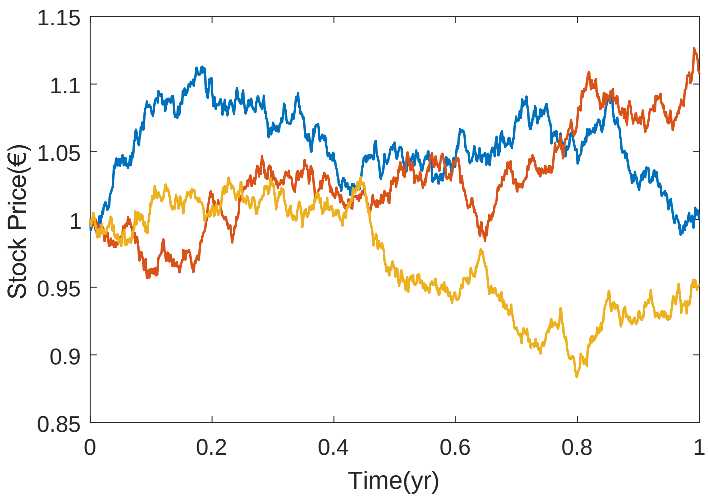


Figure 2.2: Example of three Geometric Brownian Motion processes with maturity $T = 1\,\mathrm{yr}$, interest rate $r = 0.01\,\mathrm{yr}^{-1}$, volatility $\sigma = 0.1\,\mathrm{yr}^{-1/2}$ and initial stock price $S_0 = 1\,€$.

With this result, pricing options is fairly straightforward - we simply need to solve the PDE in eq.(2.9) as we would for the diffusion equation's initial value problem [11]. The results published originally by

Black *et al.* state that, at time t, call and put options can be valued as

$$\begin{aligned} C(S(t),t) &= N(d_1)S(t) - N(d_2)Ke^{-r(T-t)}; \\ P(S(t),t) &= -N(-d_1)S(t) + N(-d_2)Ke^{-r(T-t)}, \end{aligned} \tag{2.11}$$

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

$$\begin{aligned} d_1 &= \frac{1}{\sigma\sqrt{T-t}}\left[\ln\left(\frac{S_t}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)(T-t)\right]; \\ d_2 &= d_1 - \sigma\sqrt{T-t}. \end{aligned} \tag{2.12}$$

From eq.(2.11) we can derive the relationship between $C(S,t)$ and $P(S,t)$, known as the *put-call parity*

$$C(S(t),t) = S(t) - Ke^{-r(T-t)} + P(S(t),t). \tag{2.13}$$

Because of this duality, we can always easily obtain the prices of put options from the prices of call options with the same underlying asset, maturity and strike. For this reason, some of the results presented in later sections only apply to call options, though we can just as easily find their put option equivalent.

2.4 Volatility

As mentioned, volatility is a measure of the uncertainty of future stock price movements. In other words, a higher volatility will lead to greater future fluctuations in the stock price, whereas a stock with lower volatility is more stable. This phenomenon is exemplified in Figure 2.3, where we can see the greater fluctuations of the high-volatility process (red) compared to the much smaller variations of the low-volatility process (orange).

Of all the parameters in the BS formula (eq.(2.9)), volatility is the only one we can't easily measure from market data. Furthermore, unlike the interest rate, volatility has a great impact on the behavior of stock prices and, consequently, on the price of options [9]. These two factors make volatility one of the most important subjects in all of mathematical finance and thus the focus of much research.

It should be noted that there are several types of volatility, depending on what is being measured. Some of these types will now be introduced and studied.

2.4.1 Implied Volatility

Implied volatility can be described as the value of stock price volatility that, when input into the BS pricer in eq.(2.11), outputs a value equal to the market price of a given option. In other words, it would be the stock price volatility that the seller/buyer of the option used when pricing it (assuming the BS model was used).

Because eq.(2.11) is not invertible, we need to use some numerical procedure (e.g. Newton's

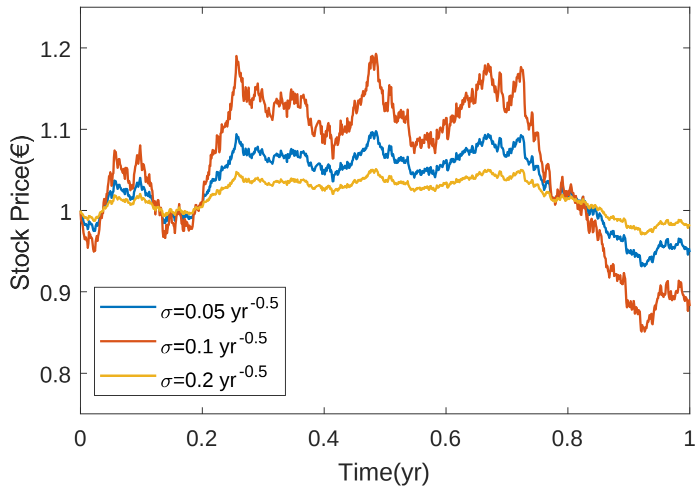


Figure 2.3: Example of three identical GBM processes with maturity $T = 1\,\text{yr}$, interest rate $r = 0.01\,\text{yr}^{-1}$ and initial stock price $S_0 = 1$€. The volatilities are $\sigma = 0.05\,\text{yr}^{-1/2}$, $\sigma = 0.1\,\text{yr}^{-1/2}$ and $\sigma = 0.2\,\text{yr}^{-1/2}$ for the orange, blue and red plot lines represented, respectively. To emphasize this effect, the underlying Brownian Motion $W(t)$ used to generate all three paths was the same.

method) to find the value of implied volatility that matches the market and model prices, i.e. we must find, numerically, the solution to the equation

$$C(\sigma_{imp}, S(t), t) - \overline{C} = 0, \tag{2.14}$$

where $C(\sigma_{imp}, \cdot)$ corresponds to the result of eq.(2.11) using σ_{imp} as (implied) volatility and $\overline{C}$ the price of the option observed in the market.

Because eq.(2.11) is monotonic w.r.t. the volatility, we can obtain the implied volatility of an option from its price and vice versa. This duality is so fundamental that investors often disclose options by providing their implied volatility instead of their price [12].

One important property of implied volatility is that, in the real-world, it depends on the strike price and the maturity. This should not occur in the ”Black-Scholes world”. Because the volatility is a property of the stock, if investors really used the the BS model to price their options, two options with the same underlying stock should have the same implied volatility, regardless of their strike prices or maturities (i.e. the same stock can't have two different volatilities at the same time). However, when observing real market data, this is in fact what is observed. The implied volatilities' dependence on the strike price can take one of two forms, known as *smile* and *skew*. An implied volatility smile presents higher volatilities for options with strikes farther from the current stock price (i.e. the shape of a smile). A skew, on the other hand, only presents higher volatilities in one of these directions (i.e. only for strikes either greater or smaller than the current stock price). Both phenomena are represented in Figure 2.4.

Because of their higher implied volatility, we can conclude that options with strikes different from the current stock price are *overpriced*. The reason behind this odd market behavior is related to the simple demand-supply rule [9]. On the one hand, some investors are risk-averse and want to hedge their losses

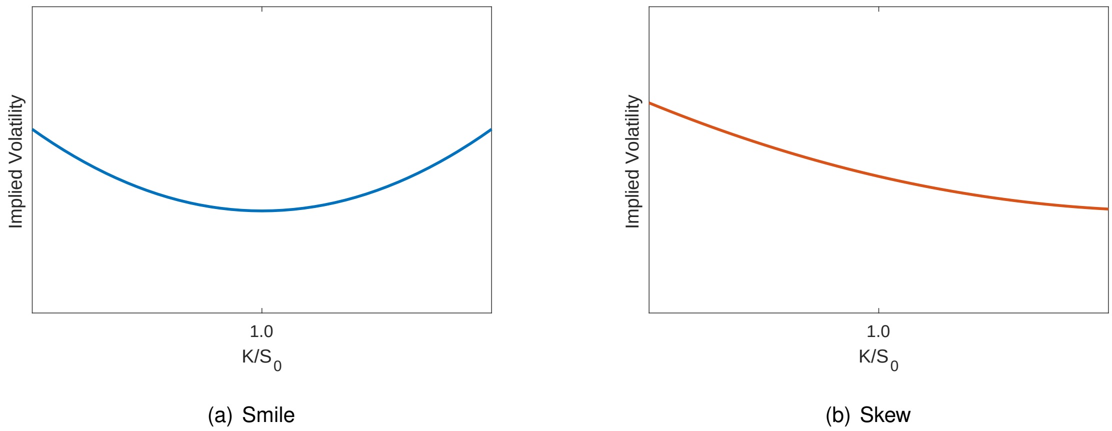


Figure 2.4: Representation of the implied volatility smile (left) and skew (right) functions.

in case of a market crash (as explained in subsection 1.3.1). They don't mind paying a higher price for an option if this means they would be relatively safe from potentially devastating market crashes. For this reason, the prices of call options with lower strikes increase, driving their implied volatility up. On the other hand, other investors are risk-seekers and want to take advantage of possible sudden price movements, buying the stocks for the lower strike prices. They don't mind paying higher prices for the chance of earning high profits and this drives the prices of high strike call options (and, consequently, their implied volatility) up. This fear-greed duality gives rise to the observed volatility smile. In the case of the volatility skew, only one of the two fenomena described occurs.

The presence of a smile instead of a skew, and vice-versa, is determined by the type of product serving as underlying asset. For example, Forex market options usually exhibit volatility smiles whereas index and commodities options usually present a volatility skew.

The dependence of the implied volatility on the maturity date is more complex, but in general it decreases with T.

It can also be shown that the implied volatility is the same for calls and puts [3], though the causes of the volatility smile/skew for put options are the opposite of the ones described before, for calls.

2.4.2 Local Volatility

In their original work, Black *et al.* assumed that volatility is constant throughout the whole option duration. From market data, it can be clearly seen that this is not the case. There may be times where new information reaches the market (e.g. the results of an election) and trading increases, driving volatility up. It is equally true that shortly before this information is known, trading may stall, and volatilities go down.

The BS model is therefore clearly incapable of completely grasping real-world trading. We should use a model where volatility is dynamic, measuring the uncertainty on the stock price movement at any point in time. However, as we saw in subsection 2.4.1, the market's view of volatility also depends on the strike price. The volatility should therefore be a function of both time and stock price: $\sigma(S, t)$. We call this model *local volatility* and the geometric Brownian motion from eq. 2.10 is transformed into

$$dS(t) = rS(t)dt + \sigma(S,t)S(t)dW(t), \tag{2.15}$$

where $\sigma(S,t)$ is some function of S and t.

We should note that finding the local volatility function is unnecessary when pricing European options - we can simply find the implied volatility of the option we're pricing from the data of similar options in the market and then use eq.(2.11), assuming a constant volatility, to find the price of the European option. However, for other contracts, such as American, Asian or Barrier options (among others), where the option's value depends on the intermediate stock prices, it is indeed crucial to appropriately model the volatility.

Because we can't directly measure the local volatility of a stock from market data, we need some models to find it. One of the most used of these is known as Dupire's formula.

Dupire's Formula

One of the most famous results in the modelling of the local volatility function was obtained by Dupire [13]. In his article, this author derives a theoretical formula for $\sigma(S,t)$, given by

$$\sigma(S,t) = \sqrt{\frac{\dfrac{\partial C}{\partial T} + rS\dfrac{\partial C}{\partial K}}{\dfrac{1}{2}S^2\dfrac{\partial^2 C}{\partial K^2}}}, \tag{2.16}$$

where $C = C(K,T)$ is the price of an European call option with strike price K and maturity T and where all the derivatives are evaluated at $K = S$ and $T = t$. A brief demonstration of this formula can be found in Appendix B.

As can be seen, we need to differentiate the option prices with respect to their strikes and maturities. To achieve this, we need first to gather, from the market, a large number of prices for options with different maturities and strikes. We then implement some interpolation on these values to obtain an option price surface (with K and T as variables). Finally, we calculate the gradients of this interpolated surface and input them into eq.(2.16) to obtain the local volatility surface. We can then sample the local volatility at each time step of our simulation.

Even before implementation, four potential sources of error can be found:

- First, it should be noted that markets only trade options with very specific maturities (e.g. 1, 2, 4 and 6-months maturity). For this reason, our data will be extremely sparse w.r.t. maturity and the interpolation generated may not correspond to reality.

- Furthermore, it can be shown that, for strikes much greater or much smaller than the current stock price, the option price's dependence on the strike is approximately linear. The second derivative in these regions would therefore be very close to zero. Because this second derivative is in the denominator of eq.(2.16), the obtained volatilities may explode for very large or very small strikes.

- There is also the problem of noise. Because we are interpolating very sparse data, even small fluctuations in the option market price may cause great variations in the option price interpolation. This can be specially problematic in regions where the second derivative is small, because, as discussed, the volatility is very sensitive to this value.

- Finally, some problems arise from the market itself. While most investors use some theoretical basis in their trades, the market is still governed by the demand-supply rule. If too many investors want to buy the option and few want to sell it, the option price will increase, even if it means that the option will be overpriced, and vice versa. Furthermore, the market is not very liquid for options with very large maturities or very large/small strikes (i.e. almost no trade occurs for these options) which causes the option prices to not truly follow the market's perception of future price movements.

All these problems must be taken into account when applying Dupire's model.

Fortunately, Dupire also developed an alternative local-volatility formula based on the implied volatility surface instead of the option price's, as seen in eq.2.16. The relation obtained is

$$\sigma(S,t) = \sqrt{\frac{\sigma_{imp}^2 + 2t\sigma_{imp}\dfrac{\partial \sigma_{imp}}{\partial T} + 2rSt\sigma_{imp}\dfrac{\partial \sigma_{imp}}{\partial K}}{\left(1 + Sd_1\sqrt{t}\dfrac{\partial \sigma_{imp}}{\partial K}\right)^2 + S^2 t\sigma_{imp}\left(\dfrac{\partial^2 \sigma_{imp}}{\partial K^2} - d_1\left(\dfrac{\partial \sigma_{imp}}{\partial K}\right)^2\sqrt{t}\right)}}, \tag{2.17}$$

where d_1 is given by

$$d_1 = \frac{\log(S_0/S) + \left(r + \frac{1}{2}\sigma_{imp}^2\right)t}{\sigma_{imp}\sqrt{t}}, \tag{2.18}$$

with S_0 being the stock price at $t = 0$. We define $\sigma_{imp} = \sigma_{imp}(K,T)$ as the interpolated surface of the implied volatilities evaluated at time T, and price K. All derivatives are evaluated at $K = S$ and $T = t$.

Because some of the shortcomings described for eq.(2.16) do not apply to eq.(2.17), and because the latter is more stable than the former [9], this new equation will be adopted in the implementation.

Two other problems can be identified in Dupire's local volatility assumption, however. First, it can be shown that if the local volatility surface truly matched reality, it should remain unchanged, i.e. the local volatility surface measured today and again in one month's time should, in theory, be the same. However, by studying market data, we can see that this is really not the case [9]. We can therefore conclude that the model doesn't completely correspond to reality and for that reason it shouldn't be used blindly. Furthermore, some authors have also pointed out that the volatility smile obtained from Dupire's local-volatility model doesn't follow real market dynamics [14]: it can be shown that when the price of the stock either increases or decreases, the volatility smile predicted by Dupire's model shifts in the opposite direction. The minimum of the volatility smile would therefore be offset and no longer correspond to the local stock price. The volatility smile dynamics obtained from the local-volatility model would thus be actually worse than if we assumed a constant volatility.

Despite its problems, Dupire's formula is still very much used in real-world practice and performs surprisingly well, as we will see in later chapters.

2.4.3 Stochastic Volatility

As stated before, the volatility is not constant, is not observable and is not predictable, despite our attempts to model it. This seems to indicate that volatility is itself also a stochastic process. Some research has been done into this hypothesis, and many models have been developed to replicate real-world volatilities. Though stochastic volatility is a sort of local volatility, for simplicity we will refer to them as separate things.

As before, we assume that the stock price follows a geometric Brownian motion

$$dS(t) = rS(t)dt + \sigma(S,t)S(t)dW_1(t), \tag{2.19}$$

but we further hypothesize that the volatility follows

$$d\sigma(S,t) = p(S,\sigma,t)dt + q(S,\sigma,t)dW_2(t), \tag{2.20}$$

where $p(S,\sigma,t)$ and $q(S,\sigma,t)$ are some functions of the stock price S, time t and of the volatility σ itself. We also assume that W_1 and W_2 are two Brownian motion processes with a correlation of ρ, i.e.

$$dW_1 dW_2 = \rho dt. \tag{2.21}$$

This correlation factor ρ can be explained by the relationship between prices and volatilities. Usually, when prices decrease, trade goes up and thus rises the volatility. The inverse is true when prices increase. This seems to indicate the existence of a negative correlation between stock prices and volatilities, though positive correlations are also possible.

Choosing the appropriate functions $p(S,\sigma,t)$ and $q(S,\sigma,t)$ is very important since the whole evolution of the stock price depends on them. All stochastic volatility models present a different version of these functions, and each may be more adequate for some types of assets. Furthermore, these functions have some parameters that we have to calibrate in order to best fit our model to market data, as we will see later.

We now present two of the most used stochastic volatility models - *Heston* and *SABR*.

Heston Model

show proof in appendix

Feller condition

One of the most popular stochastic volatility models is known as *Heston model*. It was developed in 1993 by Steven Heston [15] and it states that stock prices satisfy the relations

$$dS = rSdt + \sqrt{\nu}SdW_1, \tag{2.22}$$

$$d\nu = \kappa(\overline{\nu} - \nu)dt + \eta\sqrt{\nu}dW_2, \tag{2.23}$$

with ν corresponding to the stock price variance (i.e. $\nu = \sigma^2$) and again where W_1 and W_2 have a correlation of ρ. The original model used a drift parameter μ instead of the risk-free measure drift r presented here, but a measure transformation, using Gisarnov's theorem, can be easily implemented [16].

The parameters κ, $\overline{\nu}$ and η are, respectively, the mean-reversion rate (i.e. how fast the volatility converges to its mean value), the long-term variance (i.e. the mean value of variance) and the volatility of volatility (i.e. how erratic is the volatility process).

One of the reasons why the Heston model is so popular is the fact that there exists a closed-form solution for the prices of European options priced under this model. This closed form solution is given by

$$\begin{aligned} C_H(\theta; K, T) &= e^{-rT}\mathbb{E}\left[(S(T) - K)\,\mathbb{1}_{\{S(T)>K\}}\right] \\ &= e^{-rT}\left(\mathbb{E}\left[S(T)\mathbb{1}_{\{S(T)>K\}}\right] - K\mathbb{E}\left[\mathbb{1}_{\{S(T)>K\}}\right]\right) \\ &= S(0)P_1 - e^{-rT}KP_2, \end{aligned} \tag{2.24}$$

where $C_H(\theta; K, T)$ corresponds to the theoretical European call option price under the Heston model, assuming a parameter set θ, strike K and maturity T. The variables P_1 and P_2 are given by

$$P_1 = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left(\frac{e^{-iu\log K}}{iuS(0)e^{rT}}\phi(\theta; u - i, T)\right) du, \tag{2.25}$$

$$P_2 = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left(\frac{e^{-iu\log K}}{iu}\phi(\theta; u, T)\right) du, \tag{2.26}$$

where i is the imaginary unit and $\phi(\theta; u, t)$ is the characteristic function of the logarithm of the stock price process - the characteristic function of a random variable is the Fourier transform of the probability density function of that variable.

It is crucial to find the appropriate characteristic function $\phi(\theta; u, t)$ in order to evaluate the integrals in eqs.(2.25) and (2.26) and with them find the option price with eq.(2.24). In his original article, Heston derived this very characteristic function [15]. However, some authors showed that it contained discontinuities for large maturities and wasn't easily derivable [17], making eq.(2.24) very difficult to evaluate. These shortcomings led some other authors to propose several modified versions of this function [18, 19]. Most recently, Cui *et al.* [20] presented a characteristic function that not only doesn't have the previously mentioned discontinuities but is also easily derivable, given by

$$\phi(\theta; u, t) = \exp\left\{iu\left(\log S_0 + rt\right) - \frac{t\kappa\overline{\nu}\rho iu}{\eta} - \nu_0 A + \frac{2\kappa\overline{\nu}}{\eta^2}D\right\}, \tag{2.27}$$

with A and D given by

$$A = \frac{A_1}{A_2}, \tag{2.28}$$

$$D = \log C + \frac{(\kappa - C)t}{2} - \log\left(\frac{C+\xi}{2} + \frac{C-\xi}{2}e^{-Ct}\right), \tag{2.29}$$

where we introduce the variables A_1, A_2, C and ξ

$$A_1 = (u^2 + iu)\sinh\frac{Ct}{2}, \tag{2.30}$$

$$A_2 = C\cosh\frac{Ct}{2} + \xi\sinh\frac{Ct}{2}, \tag{2.31}$$

$$C = \sqrt{\xi^2 + \eta^2(u^2 + iu)}, \tag{2.32}$$

$$\xi = \kappa - \eta\rho iu. \tag{2.33}$$

With this result we are now able to find the prices of options under the Heston model for a given set of parameters θ. However, we need to find the optimal set of parameters such that the Heston model appropriately replicates market prices. This procedure is called called calibration and will be approached in detail in Chapter 3.

SABR Model

One other very famous model for stochastic volatility was developed by Hagan *et al.* [14] and is known as *SABR*. It stands for "*stochastic*-$\alpha\beta\rho$" and in this model it is assumed that the option prices and volatilities follow [21]

$$dS = rSdt + e^{-r(T-t)(1-\beta)}\sigma S^{\beta}dW_1, \tag{2.34}$$

$$d\sigma = \nu\sigma dW_2, \tag{2.35}$$

where we define $\alpha = \sigma(0)$ as the starting volatility and $S_0 = S(0)$ as the starting stock price. Finally, as before, the two Brownian motion processes W_1 and W_2 have a correlation of ρ.

In the original article, the authors claim that β can be fitted from historical market data, but usually investors choose this value heuristically, depending on the type of assets traded. Typical values used are $\beta = 1$ (stochastic lognormal model), used for foreign exchange options, $\beta = 0$ (stochastic normal model), typical for interest rate options and $\beta = 0.5$ (stochastic CIR model), also common for interest rate options. We will leave this parameter free upon implementation, assuming no heuristics.

One of the main problems with the SABR model is the fact that, unlike the Heston model, the stochastic volatility process is not mean-reverting. This shortcoming enables the volatility to evolve unrestrictedly which is problematic - it may become negative, which is clearly absurd, or it may become extremely large, which is troublesome. Labordère [22] proposed a mean-reverting correction to SABR, but we will study the original model by Hagan *et al.*, as it is more commonly used.

One of the main reasons why SABR is so popular is due to its quasi-closed-form solutions that enable us to quickly find the implied volatilities of options priced under this model. With the corrections done by

Oblój on Hagan's original formula [23], it can be shown that these implied volatilities are given by

$$\begin{aligned}\sigma_{SABR}(K,f,T) \approx & \frac{1}{\left[1+\frac{(1-\beta)^2}{24}\log^2\left(\frac{f}{K}\right)+\frac{(1-\beta)^4}{1920}\log^4\left(\frac{f}{K}\right)\right]}\cdot\left(\frac{\nu\log\left(f/K\right)}{x(z)}\right) \\ & \cdot\left\{1+T\left[\frac{(1-\beta)^2}{24}\frac{\alpha^2}{(Kf)^{1-\beta}}+\frac{1}{4}\frac{\rho\beta\nu\alpha}{(Kf)^{(1-\beta)/2}}+\frac{2-3\rho^2}{24}\nu^2\right]\right\},\end{aligned} \tag{2.36}$$

with z and $x(z)$ given by

$$z = \frac{\nu\left(f^{1-\beta}-K^{1-\beta}\right)}{\alpha(1-\beta)}, \tag{2.37}$$

$$x(z) = \log\left\{\frac{\sqrt{1-2\rho z+z^2}+z-\rho}{1-\rho}\right\}, \tag{2.38}$$

where we have used $f = S_0 e^{rT}$.

Similarly to the Heston model, we again need to find the set of parameters that minimize the difference between the implied volatilities obtained from the model and those obtained from market data. This calibration procedure will also be studied in Chapter 3.

Dynamic SABR Model

One of the main setbacks of the SABR model is the fact that it only works for a set of options with the same maturity. The model behaves badly when we try to fit options with different maturities [14].

To solve this problem, Hagan *et al.* suggested a similar model known as *Dynamic SABR* [14]. It follows the same processes presented in eqs.(2.34) and (2.35) but with time-dependent parameters $\rho(t)$ and $\nu(t)$,

$$dS = rSdt + e^{-r(T-t)(1-\beta)}\sigma S^\beta dW_1, \tag{2.39}$$

$$d\sigma = \nu(t)\sigma dW_2, \tag{2.40}$$

with the correlation between W_1 and W_2 now given by $\rho(t)$.

Hagan *et al.* derived again a quasi-closed-form solution for the implied volatilities of options priced under this model. Osajima later simplified this expression using asymptotic expansion [24]. The resulting formula is given by

$$\sigma_{DynSABR}(K,f,T) = \frac{1}{\omega}\left(1+A_1(T)\log\left(\frac{K}{f}\right)+A_2(T)\log^2\left(\frac{K}{f}\right)+B(T)T\right), \tag{2.41}$$

where $f = S_0 e^{rT}$, $\omega = f^{1-\beta}/\alpha$ and where $A_1(T)$, $A_2(T)$ and $B(T)$ are given by

$$A_1(T) = \frac{\beta-1}{2}+\frac{\eta_1(T)\omega}{2}, \tag{2.42}$$

$$A_2(T) = \frac{(1-\beta)^2}{12}+\frac{1-\beta-\eta_1(T)\omega}{4}+\frac{4\nu_1^2(T)+3(\eta_2^2(T)-3\eta_1^2(T))}{24}\omega^2, \tag{2.43}$$

$$B(T) = \frac{1}{\omega^2}\left(\frac{(1-\beta)^2}{24}+\frac{\omega\beta\eta_1(T)}{4}+\frac{2\nu_2^2(T)-3\eta_2^2(T)}{24}\omega^2\right), \tag{2.44}$$

with $\nu_1^2(T)$, $\nu_2^2(T)$, $\eta_1(T)$ and $\eta_2^2(T)$ defined as

$$\nu_1^2(T) = \frac{3}{T^3}\int_0^T (T-t)^2\nu^2(t)dt, \tag{2.45}$$

$$\nu_2^2(T) = \frac{6}{T^3}\int_0^T (T-t)t\nu^2(t)dt, \tag{2.46}$$

$$\eta_1(T) = \frac{2}{T^2}\int_0^T (T-t)\nu(t)\rho(t)dt, \tag{2.47}$$

$$\eta_2^2(T) = \frac{12}{T^4}\int_0^T \int_0^t \left(\int_0^s \nu(u)\rho(u)du\right)^2 dsdt, \tag{2.48}$$

where $\rho(t)$ and $\nu(t)$ are the functions chosen to model the time dependent parameters.

We now need to empirically choose some appropriate functions for $\rho(t)$ and $\nu(t)$. We can choose these functions such that the integrals in eqs.(2.45)-(2.48) are analytically solvable, greatly simplifying the calibration of this model. One classical example [25] corresponds to

$$\rho(t) = \rho_0 e^{-at}, \tag{2.49}$$

$$\nu(t) = \nu_0 e^{-bt}, \tag{2.50}$$

with $\rho_0 \in [-1, 1]$, $\nu_0 > 0$, $a > 0$ and $b > 0$. In this particular case, $\nu_1^2(T)$, $\nu_2^2(T)$, $\eta_1(T)$ and $\eta_2^2(T)$ can be exactly derived as

$$\nu_1^2(T) = \frac{6\nu_0^2}{(2bT)^3}\left[\left(\frac{(2bT)^2}{2} - 2bT + 1\right) - e^{-2bT}\right], \tag{2.51}$$

$$\nu_2^2(T) = \frac{12\nu_0^2}{(2bT)^3}\left[e^{-2bT}(1+bT) + bT - 1\right], \tag{2.52}$$

$$\eta_1(T) = \frac{2\nu_0\rho_0}{T^2(a+b)^2}\left[(a+b)T + e^{-(a+b)T} - 1\right], \tag{2.53}$$

$$\eta_2^2(T) = \frac{3\nu_0^2\rho_0^2}{T^4(a+b)^4}\left[e^{-2(a+b)T} - 8e^{-(a+b)T} + (7 + 2T(a+b)(-3+(a+b)T))\right]. \tag{2.54}$$

Despite their analytically solvable integrals, these formulas may not be robust enough to appropriately fit large data sets. Other more robust examples are

$$\rho(t) = (\rho_0 + q_\rho t)e^{-at} + d_\rho, \tag{2.55}$$

$$\nu(t) = (\nu_0 + q_\nu t)e^{-bt} + d_\nu, \tag{2.56}$$

where we have now introduced new parameters q_ρ, d_ρ, q_ν and d_ν. The main problem with these more robust formulas is the increased number of parameters, which makes the calibration procedure much more complex. Furthermore, the integral in eq.(2.48) is not analytically solvable, and must be calculated numerically, which further complicates the calibration.

Both examples shown before will be analyzed in later chapters.

Chapter 3

Implementation

3.1 Option Pricing

The theoretical models presented in Chapter 2 attempt to replicate the movements of real-world stock prices. With these predictions, we should be able to better reproduce real option prices than if we assumed a simple constant volatility, as did Black *et al.*

Currently, the two most used methods to computationally price options are known as *finite differences* [3] and *Monte Carlo* [26].

The finite differences method is an extremely fast procedure when used to price either European or American-type options, making it very appealing in these circumstances. However, when applied to other option types whose value depends on the stock prices until maturity (e.g. Asian options), the algorithm becomes very slow, rendering it almost useless. The implementation of both Heston and SABR models (presented before) using finite differences can nonetheless be found in deGraaf [27].

With the Monte Carlo algorithm, we begin by simulating a very large number of stock price paths (e.g. 100,000 simulations). The option's payoff is then calculated for each of these simulated paths and averaged, providing a fairly good estimate of the option's value. This algorithm can be easily adapted to price exotic options, making it very attractive in such cases. In the past, simulating all the stock price paths took prohibitively long computation times and this method was often discarded for this reason. However, with the recent advancements in computer hardware and new algorithmic developments, such as GPU implementation, this shortcoming has been, to some extent, solved, making the Monte Carlo algorithm quite popular in the present. For these reasons, the Monte Carlo method will be used for the analysis of the models introduced in Chapter 2.

3.1.1 Simulating stock prices

As stated, to implement the Monte Carlo algorithm, one needs to simulate stock price paths. However, by analyzing eq.(2.10), we can see that the stock prices depend on a Brownian motion process which, due to its self-similarity, is not differentiable [28]. It follows that stock price paths can never be exactly simulated. Despite this, we can approximate the movement of stock price paths by discretizing the

Brownian motion process in time, thus avoiding its self-similarity problem. We now introduce two of the most common discretization procedures.

Euler–Maruyama discretization

One of the simplest and most used discretization methods is known as *Euler–Maruyama discretization*, and can be applied to stochastic differential equations of the type

$$dX(t) = a(X(t))dt + b(X(t))dW(t), \tag{3.1}$$

where $a(X(t))$ and $b(X(t))$ are some given functions of $X(t)$ and $\{W(t),\ t > 0\}$ defines a one-dimensional Brownian motion process. To apply this discretization, we begin by partitioning the time interval $[0, T]$ into N subintervals of width $\Delta t = T/N$ and then iteratively define

$$X_{n+1} = X_n + a(X_n)\Delta t + b(X_n)\Delta W_n, \quad n = 1, \ldots, N, \tag{3.2}$$

where $\Delta W_n = W_{t+\Delta t} - W_t$. Using the known properties of Brownian motion processes, it can be shown that $\Delta W_n \sim \sqrt{\Delta t} Z$, where $Z \sim N(0, 1)$ defines a standard normal distribution.

Applying this discretization to the Geometric Brownian motion followed by stock price paths, as seen in eq.(2.10), we arrive at

$$S(t + \Delta t) = S(t) + rS(t)\Delta t + \sigma(S(t), t)S(t)\sqrt{\Delta t} Z. \tag{3.3}$$

Due to its simplicity, the Euler–Maruyama discretization method is the most common in the simulation of stock price paths whenever we have constant or deterministic volatilities.

Milstein Discretization

For stochastic volatility models, such as Heston and SABR, where the volatility itself follows a stochastic process, the Euler–Maruyama discretization may not be sufficiently accurate. In these cases, we can apply the more precise Milstein method [29], defined as

$$X_{n+1} = X_n + a(X_n)\Delta t + b(X_n)\Delta W_n + \frac{1}{2} b(X_n)b'(X_n)((\Delta W_n)^2 - \Delta t), \tag{3.4}$$

where $b'(X_n)$ denotes the derivative of $b(X_n)$ w.r.t. X_n. Note that when $b'(X_n) = 0$, the Milstein method degenerates to the simpler Euler–Maruyama discretization. This discretization should be applied not only to the stock price process but also to the stochastic volatility.

Applying this discretization to the Heston model produces

$$S(t + \Delta t) = S(t) + rS(t)\Delta t + S(t)\sqrt{\nu(t)}\sqrt{\Delta t} Z_1 + \frac{1}{2}\nu(t)S(t)\Delta t(Z_1^2 - 1), \tag{3.5}$$

$$\nu(t + \Delta t) = \nu(t) + \kappa(\overline{\nu} - \nu(t))\Delta t + \eta\sqrt{\nu(t)\Delta t} Z_2 + \frac{\eta^2}{4}\Delta t(Z_2^2 - 1), \tag{3.6}$$

where Z_1 and Z_2 are two normal random variables with a correlation of ρ.

Applying the Milstein discretization to the SABR model results in

$$\begin{aligned} S(t+\Delta t) =& S(t) + rS(t)\Delta t + e^{-r(T-t)(1-\beta)}\sigma(t)S^{\beta}(t)\sqrt{\Delta t}Z_1 + \\ &+ \frac{\beta}{2}e^{-2r(T-t)(1-\beta)}\sigma^2(t)S^{2\beta-1}(t)\Delta t(Z_1^2-1), \end{aligned} \tag{3.7}$$

$$\sigma(t+\Delta t) = \sigma(t) + \nu\sigma(t)\sqrt{\Delta t}Z_2 + \frac{\nu^2}{2}\sigma(t)\Delta t(Z_2^2-1), \tag{3.8}$$

where again Z_1 and Z_2 are two normal random variables with a correlation of ρ.

In both models we need to generate the two correlated normal variables, Z_1 and Z_2, which we can easily produce from

$$\begin{aligned} &Z_1 \sim N(0,1); \\ &Z_2 = \rho Z_1 + \sqrt{1-\rho^2}Y, \end{aligned} \tag{3.9}$$

where $Y \sim N(0,1)$ is uncorrelated with Z_1.

Because it is more precise, the Milstein method will be used in the implementation of both Heston and SABR stochastic volatility models. The simpler Euler–Maruyama discretization will be assumed for both constant and Dupire's local volatility.

It is important to note that, the smaller our subintervals Δt are, the better is the approximation done when discretizing the Brownian motion process. However, by decreasing Δt we increase the number of intervals and with it the number of calculations required to obtain a stock price path. This compromise between computation time and precision must be handled appropriately. In Figure 3.1 we can see how smaller subintervals better resemble a true GBM process, whereas if they become too large this resemblance practically vanishes.

3.1.2 Pricing options from simulations

To price options using the Monte Carlo algorithm, we generate M paths by recursively calculating $\{S_i(t),\ \ i = 1,\dots,M\}$, using either of the discretization methods described before.

When the stock price at the maturity, $S_i(T)$, is obtained for all paths, the option's payoff for each path is calculated from eqs.(2.1)-2.6. We then average all these results and discount them back to the present, obtaining the option's value

$$\begin{aligned} C(K,T) &= e^{-rT}\frac{1}{M}\sum_{i=1}^{M}\text{Payoff}_{call}(K,T); \\ P(K,T) &= e^{-rT}\frac{1}{M}\sum_{i=1}^{M}\text{Payoff}_{put}(K,T), \end{aligned} \tag{3.10}$$

where Payoff$_{\cdot}(K,T)$ denotes the payoff function of the chosen option type (e.g. European, Asian).

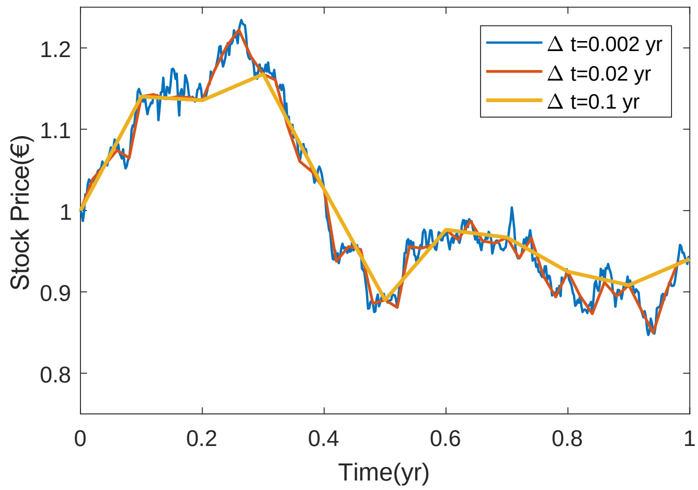


Figure 3.1: Effect of the subinterval size on the GBM discretization with maturity $T = 1\,\mathrm{yr}$, interest rate $r = 0.01\,\mathrm{yr}^{-1}$, volatility $\sigma = 0.1\,\mathrm{yr}^{-1/2}$ and initial stock price $S_0 = 1$€. The subinterval size selected was $\Delta t = 0.002\,\mathrm{yr}$, $\Delta t = 0.02\,\mathrm{yr}$ and $\Delta t = 0.1\,\mathrm{yr}$ for the blue, red and orange plot lines, respectively. To emphasize this effect, the underlying Brownian Motion $W(t)$ used to generate all three paths was the same.

3.2 Model Calibration

Both SABR and Heston stochastic volatility models contain variables that need to be calibrated in order to appropriately replicate market option prices.

Calibrating the models' parameters means finding the optimal values for these parameters such that the difference between the prices of real market options and options priced under the models' assumptions is minimized. This difference should be measured with a cost function such as

$$\mathrm{Cost}(\theta) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} \left(C_{\mathrm{market}}(T_i, K_j) - C_{\mathrm{model}}(\theta; T_i, K_j) \right)^2, \tag{3.11}$$

where we denote θ as the model's parameter set, w_i corresponds to some weight function and where $C_{\mathrm{model}}(\cdot)$ and $C_{\mathrm{market}}(\cdot)$ denote the model and market option prices, respectively, for maturities $T_i, (i = 1, \ldots, n)$ and strikes $K_j, (j = 1, \ldots, m)$.

Because we are modeling volatilities, and because implied volatilities are more sensitive to slight changes in the parameters, it is more appropriate to use a cost function based on the implied volatility instead of the price, such as

$$\mathrm{Cost}(\theta) = \sum_{i=1}^{n} \sum_{j=1}^{m} w_{i,j} \left(\sigma_{imp,\mathrm{mkt}}(T_i, K_j) - \sigma_{imp,\mathrm{mdl}}(\theta; T_i, K_j) \right)^2, \tag{3.12}$$

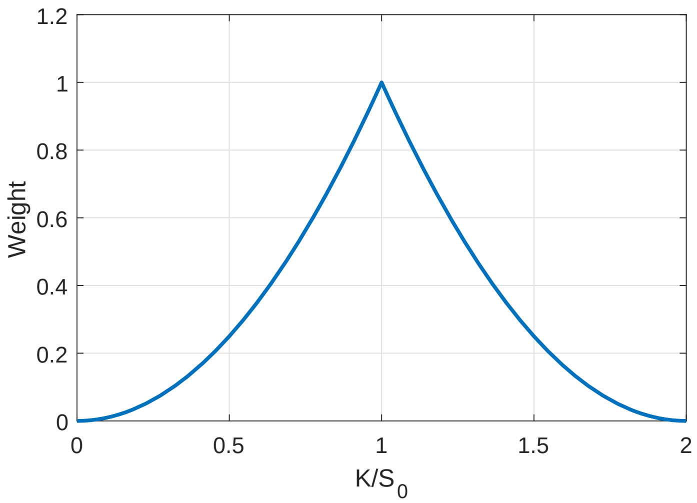
where $\sigma_{imp,\mathrm{mkt}}(\cdot)$ and $\sigma_{imp,\mathrm{mdl}}(\cdot)$ correspond to the real-market and model implied volatilities, respectively, for maturities $T_i, (i = 1, \ldots, n)$ and strikes $K_j, (j = 1, \ldots, m)$. This cost function will be used in the calibration procedure, for the reason stated before.

consider other weight functions?

The weight function $w_{i,j}$ should be chosen such that higher weights are given to options with strikes closer to the current stock price S_0, because these points have a higher influence in the shape of the volatility smile than the others. One example of such a function is

$$w_{i,j} = \left(1 - \left|1 - \frac{K_j}{S_0}\right|\right)^2, \tag{3.13}$$

where we assume strikes are restricted to $K < 2S_0$. This function is represented in Figure 3.2.



K/S_0	Weight
0.50	0.2500
0.75	0.5625
0.90	0.8100
1.00	1.0000
1.10	0.8100
1.25	0.5625
1.50	0.2500

Figure 3.2: Weight function plot and significant weight values

As we can see, the maximum weight value is given to the point where the strike equals the initial stock price ($K = S_0$) and less weight is given for points farther from this value.

To obtain the value of the cost function for a given set of parameters, we need to calculate $m \times n$ model option prices (see eq.(3.12)). We could achieve this by implementing the Monte Carlo method with the discretization procedures described before. Because we want to calibrate the model's parameters, a large number of instances of the cost function will have to be executed for our optimization algorithms to converge to an optimal solution. Thus, it can be seen that a very great number of Monte Carlo pricers will have to be executed. For this reason, even with GPU implementation and recent hardware, using Monte Carlo to calibrate the model's parameters will become prohibitively slow. We could reduce the computation time by limiting the number of simulated paths, but this would introduce a high amount of noise in the prices, making the optimization procedure nearly impossible. Thus, we can conclude that though the Monte Carlo algorithm is very useful to price a small amount of options, it is nearly useless in the calibration procedures required to use both the Heston and SABR models.

Fortunately, as mentioned, both Heston and SABR have closed-form solutions, shown in eqs. (2.24), (2.36) and (2.41), that we can use to directly price the options with each model's parameters, without the need to run the slow Monte Carlo pricer. The optimization algorithms should then converge much faster to the optimal solution for the model's parameters.

3.2.1 Optimization Algorithms

There are several possible methods to find the parameter set that minimizes the cost function shown in eq.(3.12). Our main concern when choosing the best algorithms for calibration is the nonlinearity of the cost function. This is problematic because several local minima might exist and an unsuitable algorithm may get stuck in these points, causing the globally optimal solution to not be found.

With this issue in mind, we selected two powerful algorithms known as *Multi-Start* [30] and *CMA-ES* [31] (short for Covariance Matrix Adaptation Evolution Strategy), which we will summarize below. It should be noted that we will only provide a general idea of how each optimizer works. For detailed descriptions, the original sources should be consulted.

The optimization algorithms will search the D-dimensional sample space (D corresponds to the number of parameters of each model), for the optimal solution. Each point in this space corresponds to a possible set of parameters, θ. The optimizers should also deal with parameter boundaries, since these are required for some of the models' parameters (e.g. the correlation parameter, ρ, in both Heston and SABR is obviously contained between -1 and 1).

Feller's condition inequalities

Multi-Start Optimizer

The Multi-Start optimizer is nothing more than the application of a simple optimization algorithm multiple times with multiple different starting points.

This algorithm starts by generating a set of N different starting points, $\theta_{0,i}, \;\; i = 1, \ldots, N$, distributed in the D-dimensional sample space. These can be generated at random (i.e. by sampling from a uniform distribution) or using some complex meta-heuristic such as scatter search [30]. For simplicity, and because it is usually faster to execute, we will use the former randomized sampling.

The procedure then applies a weak optimizer to each of these starting points, finding one (local) minimum for all of them. Examples of such simple optimizers are the *Active Set Method*, *Sequential Quadratic Programming*, among others. These optimizers are weak because they are only expected to converge to the (local) optimum closest to their starting point, though they are very fast to converge.

After a local minimum is found for each of the selected starting points, the minimum where the cost function is minimized is chosen as optimal solution.

This procedure is depicted in Algorithm 1.

Algorithm 1: Multi-Start Optimizer

Generate $\theta_{0,i}, \quad i = 1, \ldots, N$ `/* Multiple starting points */`

for $i = 1, \ldots, N$ **do**

  Run weak optimization algorithm with starting point $\theta_{0,i}$

  Calculate $\mathrm{Cost}(\theta'_i)$ for the minimum found, θ'_i

end

Optimal parameters: $\theta^* = \arg\min_{\theta'_i} \{\mathrm{Cost}(\theta'_i)\}$

One of the advantages of the Multi-Start is the fact that, because the weak algorithms are independent of one another (assuming no meta-heuristics are used), we can run them in parallel, further increasing calibration speed. As a disadvantage, we can point out the fact that, for highly nonlinear functions, a large number of starting points may be required, decreasing the calibration speed.

As a sidenote, we should point out that, for a large enough starting sample set, N, the global minimum will be found with probability 1, even for highly nonlinear objective functions. Though the proof is trivial, this remark is important, because we need a compromise between a large starting set that takes too long to compute and a small data set that might return a non-optimal result.

This optimizer is implemented in MATLAB with the *MultiStart* function [32]

CMA-ES Optimizer

The CMA-ES optimizer belongs to the class of evolutionary algorithms. These methods are based on the principle of biological evolution: at each iteration (generation), new candidate solutions (individuals) are generated from a given random distribution (mutation) obtained using the data (genes) of the previous solutions (parents). Of these newly generated solutions (individuals), we select the ones where the cost function is minimized (with the best fitness) to generate the candidate solutions of the next iterations (to become the parents of the next generation) and we reject the others.

As for the CMA-ES in particular, the algorithm takes λ samples from a multivariate normal distribution in the D-dimensional sample space

$$N(\mathbf{x}; \mathbf{m}, \mathbf{C}) = \frac{1}{\sqrt{(2\pi)^D |\det \mathbf{C}|}} \exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T \mathbf{C}^{-1} (\mathbf{x}-\mathbf{m})\right), \tag{3.14}$$

where $\mathbf{m}$ and $\mathbf{C}$ correspond to the distribution's mean vector and covariance matrix. These λ samples are our candidate solutions.

We classify each of these points according to their fitness (i.e. the cost function's value for a given point). We then select the μ samples with the lowest cost and discard the others. These new points will be the parents of the next generation, i.e. they will be used to generate the new mean and covariance matrix for the normal distribution.

At each iteration, the new mean is produced from a weighted average of the points, with the weights proportional to each point's fitness. The method for the covariance matrix update is rather complex and depends not only on the μ best samples but also on the values of the covariance matrices used in previous iterations. All the basic equations required for the implementation of this optimizer can be found in Appendix C. For a more detailed explanation, as well as other aspects of the algorithm, see Hansen [33].

These sampling-classification-adaptation steps are repeated until some stopping criterion is met, such as a fixed number of iterations or an minimum error threshold. When the stopping criterion is met, the mean vector of the last iteration is assumed as the optimal parameter vector.

The number of candidate solutions generated at each step, λ, and the ones that remain after classification, μ, can be chosen arbitrarily, but an adequate heuristic is to choose $\lambda = \lfloor 4 + 3\log D \rfloor + 1$ and $\mu = \lfloor \lambda/2 \rfloor + 1$.

This procedure is summarized in Algorithm 2.

Algorithm 2: CMA-ES Optimizer

Define mean vector $\mathbf{m} = \theta_0$ `/* Initial guess */`

Define covariance matrix $\mathbf{C} = I$

while *Termination criterion not met* **do**

  Sample λ points from multivariate normal distribution $N(\mathbf{x}; \mathbf{m}, \mathbf{C})$

  Calculate the cost for all generated points and keep the μ best. Discard the rest

  Update the mean vector and covariance matrix (using eqs.(C.5) and (C.9))

end

Optimal parameters: $\theta^* = \mathbf{m}$

The complexity of the covariance matrix updating process makes the CMA-ES a very robust optimization algorithm, enabling it to find the global optimum of highly nonlinear functions [34]. Furthermore, the CMA-ES is almost parameter free. It simply requires an initial guess, to generate the starting mean vector, and the algorithm is expected to converge to a global minimum. As for disadvantages, there may be cases where the convergence of this algorithm is slow, and the Multi-Start may be used as an alternative when a fast convergence is required, though the CMA-ES will outperform the Multi-Start algorithm in terms of precision.

This optimizer was implemented by Hansen in MATLAB (as well as in other computer languages) as function *purecmaes* [35].

Chapter 4

Results

4.1 Constant Volatility Model

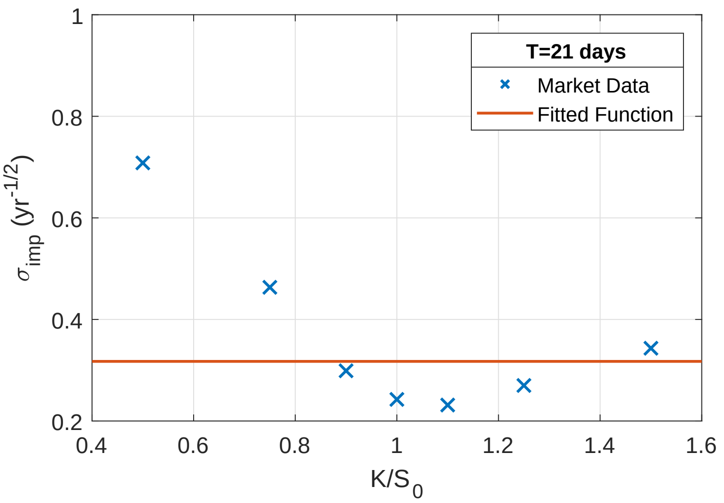


Figure 4.1: Implied volatility function fitted to the implied volatility data for maturity $T = 21$ days under constant volatility model.

$\sigma_{imp,\mathrm{mdl}}$ (yr$^{-1/2}$)	Cost
0.3174	0.0635

Table 4.1: Fit results from constant volatility model for maturity $T = 21$ days under constant volatility model.

4.2 Static SABR Model

show data and mention where it came from. confidentiality, etc

K (€)	$\sigma_{imp,\mathrm{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\mathrm{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\mathrm{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\mathrm{mkt}}(\%)$
0.50	0.7082		55.2	0.50001	0.50000	0.0026
0.75	0.4632		31.5	0.25065	0.25002	0.3
0.90	0.2989		6.2	0.10439	0.10540	1.0
1.00	0.2425	0.3174	30.9	0.02792	0.03654	30.9
1.10	0.2314		37.1	2.42×10^{-3}	7.41×10^{-3}	205.9
1.25	0.2699		17.6	5.34×10^{-5}	25.01×10^{-5}	367.9
1.50	0.3433		7.5	5.75×10^{-7}	1.12×10^{-7}	80.5

Table 4.2: Comparison between fitted function and original data for maturity $T = 21$ days under constant volatility model.

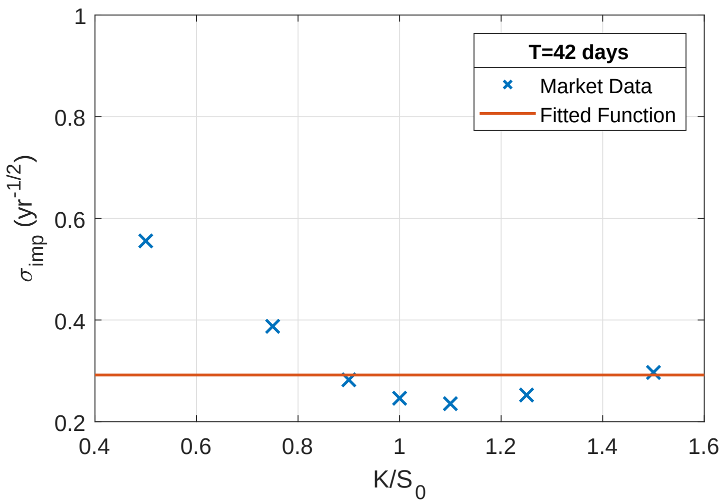


Figure 4.2: Implied volatility function fitted to the implied volatility data for maturity $T = 42$ days under constant volatility model.

$\sigma_{imp,\mathrm{mdl}}$ (yr$^{-1/2}$)	Cost
0.2918	0.0282

Table 4.3: Fit results from constant volatility model for maturity $T = 42$ days under constant volatility model.

in annex, should plot y-axes with different limits?

should we include the market values in all comparison tables?

plot surface function

put units in fitted parameters

K (€)	$\sigma_{imp,\text{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\text{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\text{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\text{mkt}}(\%)$
0.50	0.5556		47.5	0.50005	0.50000	0.01
0.75	0.3876		24.7	0.25186	0.25027	0.6
0.90	0.2824		3.3	0.11069	0.11166	0.9
1.00	0.2461	0.2918	18.6	0.04006	0.04749	18.5
1.10	0.2354		23.9	8.52×10^{-3}	15.00×10^{-3}	75.9
1.25	0.2525		15.6	6.21×10^{-4}	15.75×10^{-4}	153.8
1.50	0.2968		1.7	1.58×10^{-5}	1.24×10^{-5}	21.4

Table 4.4: Comparison between fitted function and original data for maturity $T = 42$ days under constant volatility model.

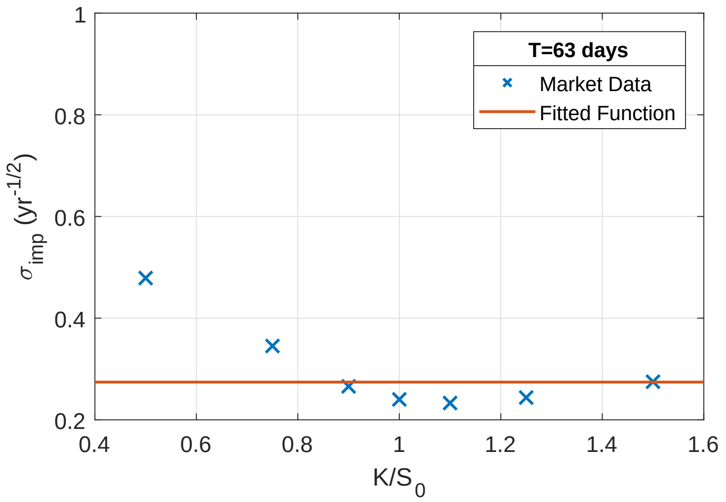


Figure 4.3: Implied volatility function fitted to the implied volatility data for maturity $T = 63$ days under constant volatility model.

$\sigma_{imp,\text{mdl}}$ (yr$^{-1/2}$)	Cost
0.2742	0.0164

Table 4.5: Fit results from constant volatility model for maturity $T = 63$ days under constant volatility model.

K (€)	$\sigma_{imp,\text{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\text{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\text{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\text{mkt}}(\%)$
0.50	0.4789		42.7	0.50009	0.50000	0.02
0.75	0.3452		20.6	0.25296	0.25077	0.9
0.90	0.2658		3.2	0.11533	0.11650	1.0
1.00	0.2401	0.2742	14.2	0.04787	0.05465	14.2
1.10	0.2330		17.7	0.01421	0.02069	45.5
1.25	0.2438		12.5	1.80×10^{-3}	3.33×10^{-3}	85.1
1.50	0.2749		0.3	7.66×10^{-5}	7.44×10^{-5}	2.9

Table 4.6: Comparison between fitted function and original data for maturity $T = 63$ days under constant volatility model.

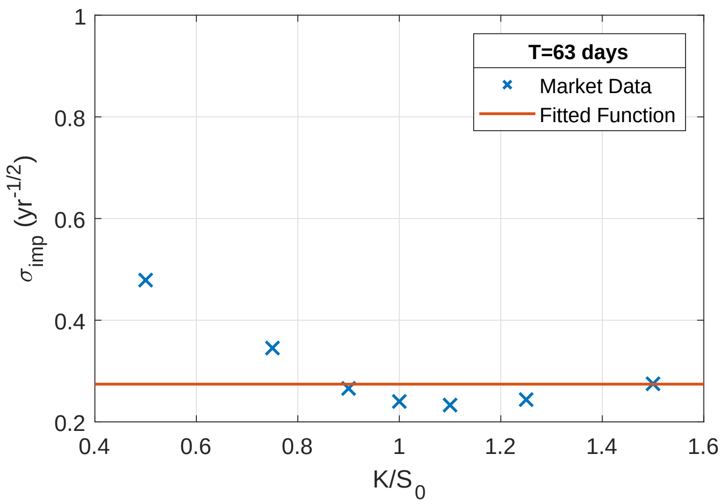


Figure 4.4: Implied volatility function fitted to the implied volatility data for maturity $T = 126$ days under constant volatility model.

$\sigma_{imp,\text{mdl}}$ (yr$^{-1/2}$)	Cost
0.2518	0.0069

Table 4.7: Fit results from constant volatility model for maturity $T = 126$ days under constant volatility model.

K (€)	$\sigma_{imp,\text{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\text{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\text{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\text{mkt}}(\%)$
0.50	0.3878		35.1	0.50035	0.50000	0.07
0.75	0.2954		14.7	0.25694	0.25344	1.4
0.90	0.2444		3.1	0.12716	0.12882	1.3
1.00	0.2295	0.2518	9.7	0.06467	0.07094	9.7
1.10	0.2269		11.0	0.02862	0.03488	21.9
1.25	0.2340		7.6	7.57×10^{-3}	9.98×10^{-3}	31.8
1.50	0.2521		0.1	8.58×10^{-4}	8.51×10^{-4}	0.8

Table 4.8: Comparison between fitted function and original data for maturity $T = 126$ days under constant volatility model.

α (yr$^{-1/2}$)	β	ρ	ν	Cost
0.2381	0.3766	-0.3760	2.1022	0.000415

Table 4.9: Fit results from constant volatility model for maturity $T = 21$ days under static SABR model.

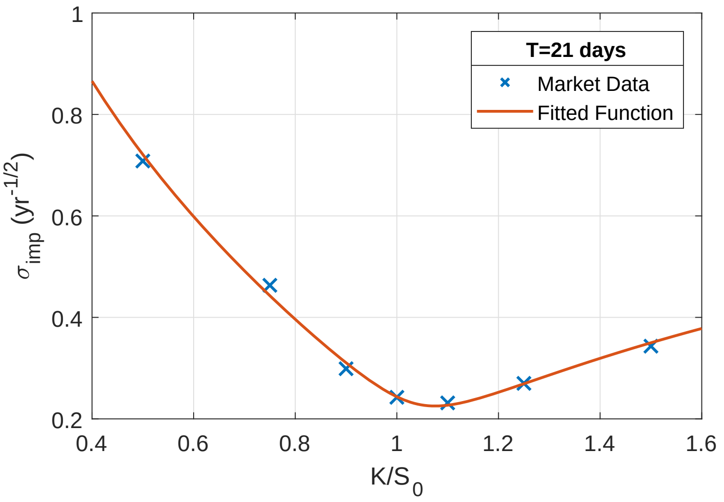


Figure 4.5: Implied volatility function fitted to the implied volatility data for maturity $T = 21$ days under static SABR model.

K (€)	$\sigma_{imp,\mathrm{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\mathrm{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\mathrm{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\mathrm{mkt}}(\%)$
0.50	0.7082	0.7209	1.8	0.50001	0.50002	0.001
0.75	0.4632	0.4428	4.4	0.25065	0.25047	0.1
0.90	0.2989	0.3105	3.9	0.10439	0.10501	0.6
1.00	0.2425	0.2435	0.4	0.02792	0.02804	0.4
1.10	0.2314	0.2269	2.0	2.42×10^{-3}	2.23×10^{-3}	8.0
1.25	0.2699	0.2692	0.3	5.34×10^{-5}	5.18×10^{-5}	3.0
1.50	0.3433	0.3500	1.9	5.75×10^{-7}	8.32×10^{-7}	44.7

Table 4.10: Comparison between fitted function and original data for maturity $T = 21$ days under static SABR model.

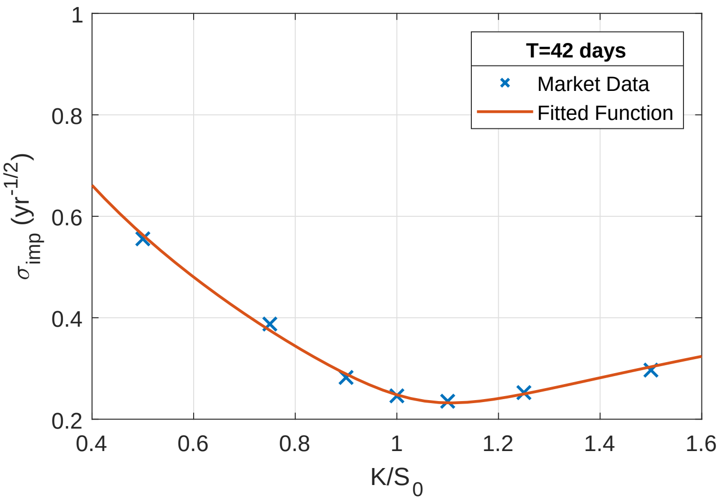


Figure 4.6: Implied volatility function fitted to the implied volatility data for maturity $T = 42$ days under static SABR model.

α (yr$^{-1/2}$)	β	ρ	ν	Cost
0.2434	0.7362	-0.3664	1.4451	0.000166

Table 4.11: Fit results from constant volatility model for maturity $T = 42$ days under static SABR model.

K (€)	$\sigma_{imp,\mathrm{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\mathrm{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\mathrm{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\mathrm{mkt}}(\%)$
0.50	0.5556	0.5631	1.4	0.50005	0.50006	0.002
0.75	0.3876	0.3751	3.2	0.25186	0.25155	0.1
0.90	0.2824	0.2891	2.4	0.11069	0.11139	0.6
1.00	0.2461	0.2481	0.8	0.04006	0.04039	0.8
1.10	0.2354	0.2322	1.4	8.52×10^{-3}	8.19×10^{-3}	3.9
1.25	0.2525	0.2497	1.1	6.21×10^{-4}	5.75×10^{-4}	7.4
1.50	0.2968	0.3033	2.2	1.58×10^{-5}	2.12×10^{-5}	33.9

Table 4.12: Comparison between fitted function and original data for maturity $T = 42$ days under static SABR model.

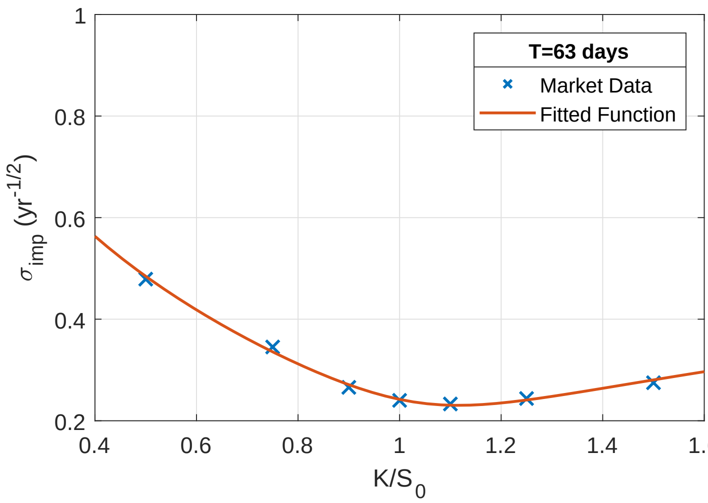


Figure 4.7: Implied volatility function fitted to the implied volatility data for maturity $T = 63$ days under static SABR model.

α (yr$^{-1/2}$)	β	ρ	ν	Cost
0.2375	0.7750	-0.3119	1.1420	0.000102

Table 4.13: Fit results from constant volatility model for maturity $T = 63$ days under static SABR model.

K (€)	$\sigma_{imp,\text{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\text{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\text{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\text{mkt}}(\%)$
0.50	0.4789	0.4845	1.2	0.50009	0.50011	0.002
0.75	0.3452	0.3357	2.8	0.25296	0.25256	0.2
0.90	0.2658	0.2710	2.0	0.11533	0.11605	0.6
1.00	0.2401	0.2421	0.8	0.04787	0.04826	0.8
1.10	0.2330	0.2305	1.1	0.01421	0.01384	2.6
1.25	0.2438	0.2409	1.2	1.80×10^{-3}	1.68×10^{-3}	6.7
1.50	0.2749	0.2804	2.0	7.66×10^{-5}	9.56×10^{-5}	24.8

Table 4.14: Comparison between fitted function and original data for maturity $T = 63$ days under static SABR model.

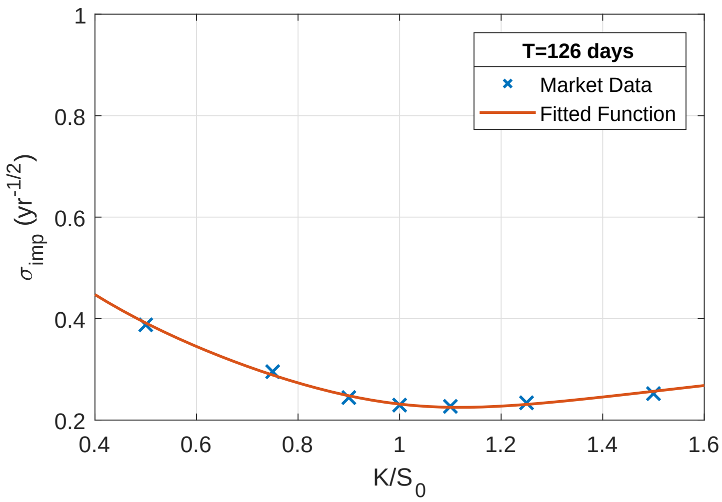


Figure 4.8: Implied volatility function fitted to the implied volatility data for maturity $T = 126$ days under static SABR model.

α (yr$^{-1/2}$)	β	ρ	ν	Cost
0.2267	0.8771	-0.2383	0.8215	0.000055

Table 4.15: Fit results from constant volatility model for maturity $T = 126$ days under static SABR model.

K (€)	$\sigma_{imp,\text{mkt}}$ (yr$^{-1/2}$)	$\sigma_{imp,\text{mdl}}$ (yr$^{-1/2}$)	$\Delta\sigma_{imp}/\sigma_{imp,\text{mkt}}(\%)$	C_{mkt} (€)	C_{mdl} (€)	$\Delta C/C_{\text{mkt}}(\%)$
0.50	0.3878	0.3914	0.9	0.50035	0.50038	0.006
0.75	0.2954	0.2887	2.2	0.25694	0.25633	0.2
0.90	0.2444	0.2479	1.5	0.12716	0.12794	0.6
1.00	0.2295	0.2314	0.8	0.06467	0.06522	0.8
1.10	0.2269	0.2251	0.8	0.02862	0.02817	1.6
1.25	0.2340	0.2309	1.3	7.57×10^{-3}	7.18×10^{-3}	5.2
1.50	0.2521	0.2567	1.8	8.58×10^{-4}	9.82×10^{-4}	14.5

Table 4.16: Comparison between fitted function and original data for maturity $T = 126$ days under static SABR model.

Chapter 5

Conclusions

Implement importance sampling

Implement antithetic paths

we tried several algorithms but CMA and multi-start were better

use mean-reverting sabr

Bibliography

[1] Bank for International Settlements. Semiannual otc derivatives statistics. `http://stats.bis.org/statx/srs/table/d5.1`. (Accessed: 2018-05-20).

[2] Financial Times. OTC derivatives shrink to lowest level since financial crisis. `https://www.ft.com/content/dbc08ae2-1247-11e6-91da-096d89bd2173`, May 2016.

[3] J. Hull. *Options, Futures, and Other Derivatives*. Boston: Prentice Hall, 2012.

[4] Investopedia. European option. `https://www.investopedia.com/terms/e/europeanoption.asp`, . (Accessed: 2018-05-29).

[5] Investopedia. American option. `https://www.investopedia.com/terms/a/americanoption.asp`, . (Accessed: 2018-05-29).

[6] F. Longstaff and E. Schwartz. Valuing american options by simulation: a simple least-squares approach. *The review of financial studies*, 14(1):113–147, 2001.

[7] Investopedia. Exotic option. `https://www.investopedia.com/terms/e/exoticoption.asp`, . (Accessed: 2018-05-29).

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

[9] P. Wilmott. *Paul Wilmott on Quantitative Finance*. The Wiley Finance Series. Wiley, 2006.

[10] D. Heath, R. Jarrow, and A. Morton. Bond pricing and the term structure of interest rates: A new methodology for contingent claims valuation. *Econometrica*, 60(1):77–105, 1992.

[11] R. Dilão, J. A. de Matos, and B. Ferreira. On the value of european options on a stock paying a discrete dividend. *Journal of Modelling in Management*, 4(3):235–248, 2009.

[12] P. Wilmott. *Paul Wilmott Introduces Quantitative Finance*. The Wiley Finance Series. Wiley, 2013.

[13] B. Dupire. Pricing with a smile. *Risk Magazine*, pages 18–20, 1994.

[14] P. Hagan et al. Managing smile risk. 1:84–108, 01 2002.

[15] S. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. 6:327–43, 02 1993.

[16] R. Crisóstomo. An analysis of the heston stochastic volatility model: Implementation and calibration using matlab. 2015.

[17] C. Kahl and P. Jäckel. Not-so-complex logarithms in the heston model.

[18] S. del Baño Rollin et al. On the density of log-spot in the heston volatility model. *Stochastic Processes and their Applications*, 120(10):2037 – 2063, 2010.

[19] W. Schoutens et al. A perfect calibration! now what?

[20] Y. Cui et al. Full and fast calibration of the heston stochastic volatility model. *European Journal of Operational Research*, 263(2):625 – 638, 2017.

[21] G. Vlaming. Pricing options with the sabr model. Master's thesis, 2011.

[22] P. H. Labordère. A general asymptotic implied volatility for stochastic volatility models. May 2005.

[23] J. Obloj. Fine-tune your smile: Correction to hagan et al. Mar 2008.

[24] Y. Osajima. The asymptotic expansion formula of implied volatility for dynamic sabr model and fx hybrid model. 2007.

[25] J. Fernández et al. Static and dynamic sabr stochastic volatility models: Calibration and option pricing using gpus. *Mathematics and Computers in Simulation*, 94:55 – 75, 2013.

[26] P. Glasserman. *Monte Carlo Methods in Financial Engineering*. Springer, 2004.

[27] C. de Graaf. Finite difference methods in derivatives pricing under stochastic volatility models.

[28] T. Mikosch. *Elementary Stochastic Calculus with Finance in View*. Advanced Series on Statistical Science and Applied Probability. 1998.

[29] G. N. Milstein. Approximate integration of stochastic differential equations. *Theory of Probability and Its Applications*, 19(3):557–562, 1975.

[30] Z. Ugray et al. Scatter search and local nlp solvers: A multistart framework for global optimization. *INFORMS Journal on Computing*, 19(3):328–340, 2007.

[31] N. Hansen. The cma evolution strategy: a comparing review. In *Towards a new evolutionary computation*, pages 75–102. Springer, 2006.

[32] Multistart Algorithm. `http://www.mathworks.com/help/gads/multistart.html`. (Accessed: 2018-05-28).

[33] N. Hansen. The cma evolution strategy: A tutorial. .

[34] R. Dilão and D. Muraro. A parallel multi-objective optimization algorithm for the calibration of mathematical models. *Swarm and Evolutionary Computation*, 8:13 – 25, 2013.

[35] N. Hansen. CMA-ES Source Code - MATLAB and Octave - Code for Reading. `http://www.lri.fr/~hansen/purecmaes.m`, . (Accessed: 2018-05-28).

Appendix A

Option Market Data

Here, we present the data kindly provided by *BNP Paribas* for European options. For confidentiality reasons, the strike prices were normalized by the initial stock price i.e. $K \to K/S_0$ so that the original strike prices are inaccessible. Suffice it to say that the underlying asset of the options here represented is a *stock index*, a weighted average of the prices of some selected stocks (e.g. PSI-20 (Portugal)).

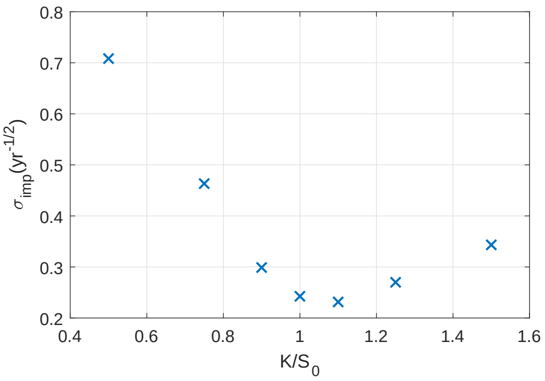
The data provided pertains to the options' implied volatilities. We can easily obtain their prices from these values using eq.(2.14). The converted prices of call European options are shown below.

The number of days here denoted correspond to trading days (i.e. days where exchanges are open and trading occurs) so that one month corresponds to 21 days and one year to 252.

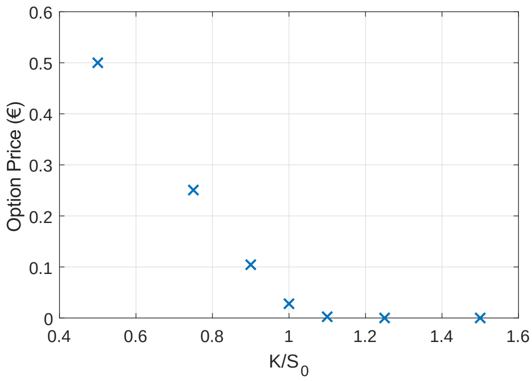
T (days)	K (€)	σ_{imp} (yr$^{-1/2}$)	C (€)
	0.50	0.7082	0.500013
	0.75	0.4632	0.250648
	0.90	0.2989	0.104393
21	1.00	0.2425	0.027923
	1.10	0.2314	0.002421
	1.25	0.2699	0.000053
	1.50	0.3433	0.000001
	0.50	0.5556	0.500050
	0.75	0.3876	0.251865
	0.90	0.2824	0.110693
42	1.00	0.2461	0.040063
	1.10	0.2354	0.008525
	1.25	0.2525	0.000621
	1.50	0.2968	0.000016
	0.50	0.4789	0.500093
	0.75	0.3452	0.252957
	0.90	0.2658	0.115328
63	1.00	0.2401	0.047872
	1.10	0.2330	0.014215
	1.25	0.2438	0.001799
	1.50	0.2749	0.000077

T (days)	K (€)	σ_{imp} (yr$^{-1/2}$)	C (€)
	0.50	0.3878	0.500353
	0.75	0.2954	0.256942
	0.90	0.2444	0.127156
126	1.00	0.2295	0.064670
	1.10	0.2269	0.028619
	1.25	0.2340	0.007569
	1.50	0.2521	0.000858
	0.50	0.3448	0.500720
	0.75	0.2729	0.261024
	0.90	0.2348	0.136975
189	1.00	0.2246	0.077469
	1.10	0.2241	0.040739
	1.25	0.2304	0.014714
	1.50	0.2438	0.002697
	0.50	0.3168	0.501122
	0.75	0.2582	0.264828
	0.90	0.2281	0.145258
252	1.00	0.2209	0.087955
	1.10	0.2220	0.051165
	1.25	0.2286	0.022195
	1.50	0.2408	0.005585

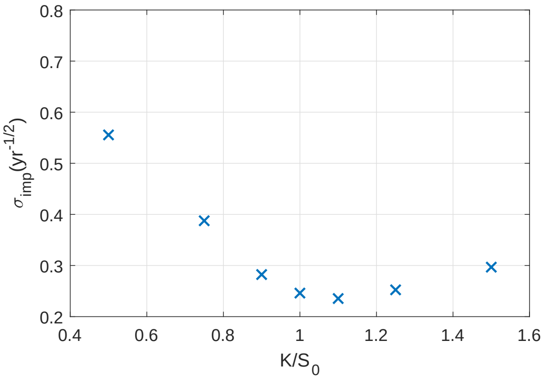
Table A.1: Data provided by *BNP Paribas* to be used in model calibration and validation.



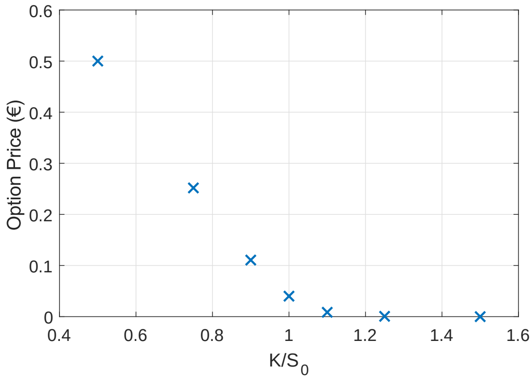
(a) Implied Volatility, T=21 days



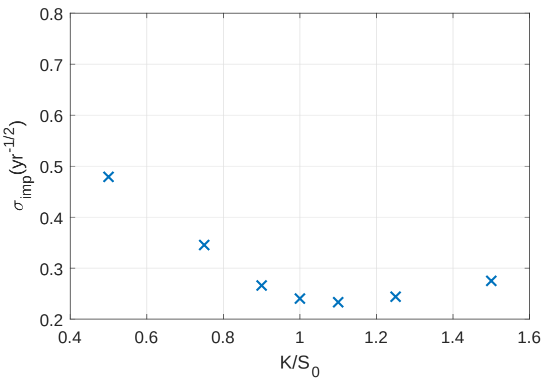
(b) European Call Price, T=21 days



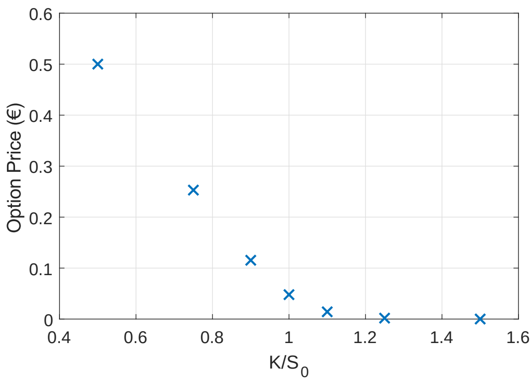
(c) Implied Volatility, T=42 days



(d) European Call Price, T=42 days

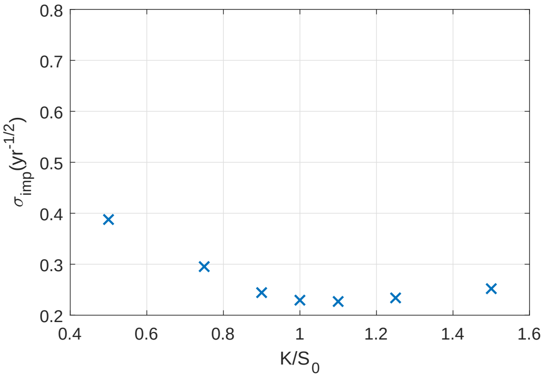


(e) Implied Volatility, T=63 days

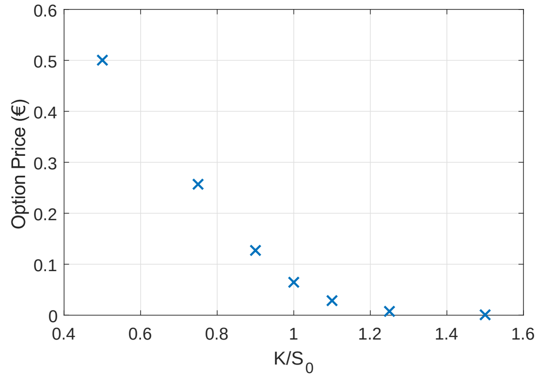


(f) European Call Price, T=63 days

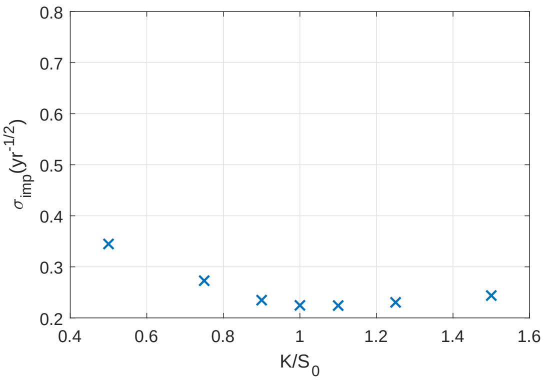
Figure A.1: Scatter plots of the implied volatilities and European call prices provided, for 21, 42 and 63 days, to be used in model calibration and validation.



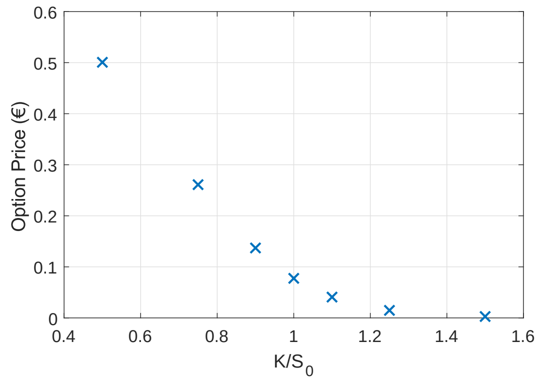
(a) Implied Volatility, T=126 days



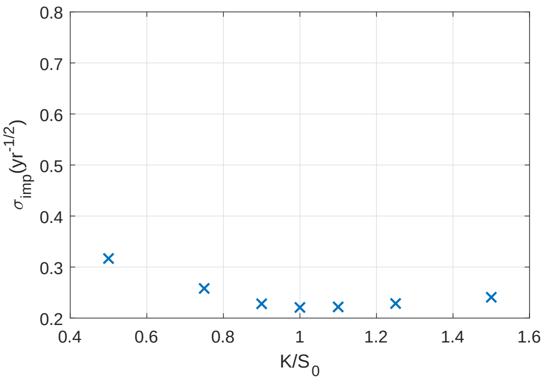
(b) European Call Price, T=126 days



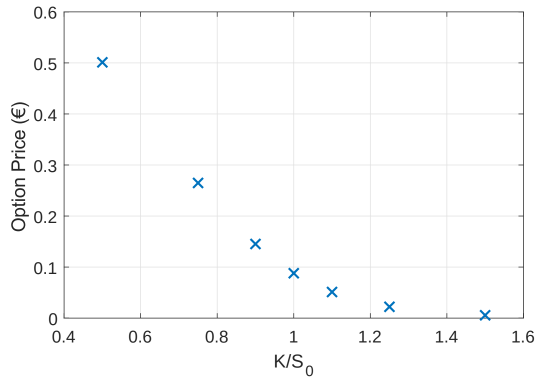
(c) Implied Volatility, T=189 days



(d) European Call Price, T=189 days



(e) Implied Volatility, T=252 days



(f) European Call Price, T=252 days

Figure A.2: Scatter plots of the implied volatilities and European call prices provided, for 126, 189 and 252 days, to be used in model calibration and validation.

Appendix B

Dupire's Formula Derivation

Here we present a brief demonstration of Dupire's formula, as shown in eq. (2.16).

In the original article, Dupire begins by assuming that the stock price S follows a dynamic transition probability density function $p(S(t), t, S'(t'), t')$. In other words, by integrating this density function we would obtain the probability of the stock price reaching a price S' at a time t' having started at S at time t.

The present value of a call option, $C(S, t, K, T)$, can be deduced as its expected future payoff, discounted backwards in time, which results in

$$\begin{aligned} C(K,T) = e^{-r(T-t)}\mathbb{E}\left[\max\left(S'-K,0\right)\right] &= e^{-r(T-t)}\int_0^\infty \max\left(S'-K,0\right)p(S,t,S',T)dS' \\ &= e^{-r(T-t)}\int_K^\infty (S'-K)p(S,t,S',T)dS'. \end{aligned} \tag{B.1}$$

Taking the first derivative of this result with respect to the strike price K, we obtain

$$\frac{\partial C}{\partial K} = -e^{-r(T-t)}\int_K^\infty p(S,t,S',T)dS'. \tag{B.2}$$

The second derivative results in

$$\frac{\partial^2 C}{\partial K^2} = e^{-r(T-t)}p(S,t,S',T). \tag{B.3}$$

Due to its stochastic nature, the transition probability density function follows the Fokker-Planck equation, given by

$$\frac{\partial p}{\partial T} = \frac{1}{2}\sigma^2\frac{\partial^2(S^2p)}{\partial S^2} - r\frac{\partial(Sp)}{\partial S}. \tag{B.4}$$

with σ denoting our (still unknown) function of S and t, evaluated at $t = T$.

From eq. B.1 we can easily derive

$$\frac{\partial C}{\partial T} = -rC + e^{-r(T-t)}\int_K^\infty (S'-K)\frac{\partial p}{\partial T}dS'. \tag{B.5}$$

Using eq. B.4, we can transform this relation into

$$\frac{\partial C}{\partial T} = -rC + e^{-r(T-t)} \int_K^\infty (S' - K) \left(\frac{1}{2}\sigma^2 \frac{\partial^2 (S'^2 p)}{\partial S'^2} - r\frac{\partial (S' p)}{\partial S'} \right) dS'. \tag{B.6}$$

Integrating twice by parts and collecting all terms, we get

$$\frac{\partial C}{\partial T} = \frac{1}{2}\sigma^2 K^2 \frac{\partial^2 C}{\partial K^2} - rK\frac{\partial C}{\partial K}. \tag{B.7}$$

Rearranging all terms, we are left with

$$\sigma(K,T) = \sqrt{\frac{\dfrac{\partial C}{\partial T} + rK\dfrac{\partial C}{\partial K}}{\dfrac{1}{2}K^2\dfrac{\partial^2 C}{\partial K^2}}}, \tag{B.8}$$

from which we can easily derive Dupire's formula, as shown in eq.(2.16), using a simple variable change, i.e. $\sigma(K,T) \implies \sigma(S,t)$, which gives

$$\sigma(S,t) = \sqrt{\frac{\dfrac{\partial C}{\partial T} + rS\dfrac{\partial C}{\partial K}}{\dfrac{1}{2}S^2\dfrac{\partial^2 C}{\partial K^2}}}. \tag{B.9}$$

Appendix C

CMA-ES Algorithm Formulas

Here we present the formulas required for the calculation of the mean vector, $\mathbf{m}$, and the covariance matrix, $\mathbf{C}$, to be used, at each iteration of the CMA-ES optimization algorithm, on the multivariate normal distribution

$$N(\mathbf{x};\mathbf{m},\mathbf{C}) = \frac{1}{\sqrt{(2\pi)^D|\mathrm{det}\mathbf{C}|}}\exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})\right). \tag{C.1}$$

C.1 The Optimization Algorithm

C.1.1 Initialization

We initialize the algorithm by setting the first mean vector, $\mathbf{m}^{(0)}$, to some initial guess, θ_0, and the covariance matrix to the unit matrix, $\mathbf{C}^{(0)} = \mathbf{I}$.

C.1.2 Sampling

We sample λ points, $\mathbf{y}_i^{(1)},\ i = 1,\ldots,\lambda$, from a multivariate normal distribution $N(\mathbf{x};\mathbf{0},\mathbf{C}^{(0)})$, generating the first candidate solutions

$$\mathbf{x}_i^{(1)} = \mathbf{m}^{(0)} + \sigma^{(0)}\mathbf{y}_i^{(1)}, \quad i = 1,\ldots,\lambda, \tag{C.2}$$

where $\sigma^{(0)} = 1$.

C.1.3 Classification

The candidate solutions are ordered based on their cost function, such that we denote $\mathbf{x}_{i:\lambda}^{(1)}$ as the i-th best classified point from the set $\mathbf{x}_1^{(1)},\ldots,\mathbf{x}_\lambda^{(1)}$. In other words, $\mathrm{Cost}(\mathbf{x}_{1:\lambda}^{(1)}) \leq \mathrm{Cost}(\mathbf{x}_{2:\lambda}^{(1)}) \leq \ldots \leq \mathrm{Cost}(\mathbf{x}_{\lambda:\lambda}^{(1)})$.

C.1.4 Selection

From the ordered set $\mathbf{x}_{i:\lambda}^{(1)}$ we choose the first μ data points (with the lowest cost) and discard the others. We then define the weights ω_i as

$$\omega_i = \frac{(\log(\mu + 1/2) - \log(i))}{\sum_{i=1}^{\mu} (\log(\mu + 1/2) - \log(i))}, \qquad i = 1, \ldots, \mu. \tag{C.3}$$

As an alternative we could also use $\omega_i = 1/\mu$.

C.1.5 Adaptation

We are finally able to calculate the new mean vector and covariance matrix using

$$\left\langle \mathbf{y}^{(k)} \right\rangle_w = \sum_{i=1}^{\mu} \omega_i \mathbf{y}_{i:\lambda}^{(k)}, \tag{C.4}$$

$$\mathbf{m}^{(k)} = \mathbf{m}^{(k-1)} + \sigma^{(k-1)} \left\langle \mathbf{y}^{(k)} \right\rangle_w = \sum_{i=1}^{\mu} \omega_i \mathbf{x}_{i:\lambda}^{(k)}, \tag{C.5}$$

$$\mathbf{p}_\sigma^{(k)} = (1 - c_\sigma)\mathbf{p}_\sigma^{(k-1)} + \sqrt{c_\sigma(2 - c_\sigma)\mu_{\text{eff}}} \left(\mathbf{C}^{(k-1)}\right)^{-1/2} \left\langle \mathbf{y}^{(k)} \right\rangle_w, \tag{C.6}$$

$$\sigma^{(k)} = \sigma^{(k-1)} \exp\left(\frac{c_\sigma}{d_\sigma}\left(\frac{\|\mathbf{p}_\sigma^{(k)}\|}{E^*} - 1\right)\right), \tag{C.7}$$

$$\mathbf{p}_c^{(k)} = (1 - c_c)\mathbf{p}_c^{(k-1)} + h_\sigma^{(k)} \sqrt{c_c(2 - c_c)\mu_{\text{eff}}} \left\langle \mathbf{y}^{(k)} \right\rangle_w, \tag{C.8}$$

$$\mathbf{C}^{(k)} = (1 - c_1 - c_\mu)\, \mathbf{C}^{(k-1)} + c_1 \left(\mathbf{p}_c^{(k)} \left(\mathbf{p}_c^{(k)}\right)^T + \delta\left(h_\sigma^{(k)}\right) \mathbf{C}^{(k-1)}\right) + c_\mu \sum_{i=1}^{\mu} \omega_i \mathbf{y}_{i:\lambda}^{(k)} \left(\mathbf{y}_{i:\lambda}^{(k)}\right)^T, \tag{C.9}$$

where we define

$$\mu_{\text{eff}} = \left(\sum_{i=1}^{\mu} \omega_i^2\right)^{-1}, \tag{C.10}$$

$$c_c = \frac{4 + \mu_{\text{eff}}/D}{D + 4 + 2\mu_{\text{eff}}/D}, \tag{C.11}$$

$$c_\sigma = \frac{\mu_{\text{eff}} + 2}{D + \mu_{\text{eff}} + 5}, \tag{C.12}$$

$$d_\sigma = 1 + 2\max\left(0,\ \sqrt{\frac{\mu_{\text{eff}} - 1}{D + 1}} - 1\right) + c_\sigma, \tag{C.13}$$

$$c_1 = \frac{2}{(D + 1.3)^2 + \mu_{\text{eff}}}, \tag{C.14}$$

$$c_\mu = \min\left(1 - c_1,\ 2\frac{\mu_{\text{eff}} - 2 + 1/\mu_{\text{eff}}}{(D + 2)^2 + \mu_{\text{eff}}}\right), \tag{C.15}$$

$$E^* = \frac{\sqrt{2}\,\Gamma\left(\frac{D+1}{2}\right)}{\Gamma\left(\frac{D}{2}\right)}, \tag{C.16}$$

$$h_{\sigma}^{(k)} = \begin{cases} 1, & \text{if} \frac{\|\mathbf{p}_{\sigma}^{(k)}\|}{\sqrt{1-(1-c_{\sigma})^{2(k+1)}}} < \left(1.4 + \frac{2}{D+1}\right) E^{*} \\ 0, & \text{otherwise} \end{cases}, \tag{C.17}$$

$$\delta\left(h_{\sigma}^{(k)}\right) = \left(1 - h_{\sigma}^{(k)}\right) c_c \left(2 - c_c\right), \tag{C.18}$$

$$\left(\mathbf{C}^{(k)}\right)^{-1/2} = \mathbf{B}\left(\mathbf{D}^{(k)}\right)^{-1} \mathbf{B}^{T}, \tag{C.19}$$

with D corresponding to the number of parameters of the model (i.e. the dimensions of the sample space) and we define $\mathbf{p}_{\sigma}^{(0)} = \mathbf{p}_{c}^{(0)} = 0$.

This steps are iterated until the termination criterion is met.

Appendix D

Placeholder

D.1 Topic Overview

Provide an overview of the topic to be studied...

D.2 Objectives

Explicitly state the objectives set to be achieved with this thesis...

D.3 Thesis Outline

Briefly explain the contents of the different chapters...

D.4 Theoretical Overview

Some overview of the underlying theory about the topic...

D.5 Theoretical Model 1

Multiple citations are compressed when using the `sort&compress` option when loading the `natbib` package as `\usepackage[numbers,sort&compress]{natbib}` in file `Thesis_Preamble.tex`, resulting in citations like [3, 12].

D.6 Theoretical Model 2

Other models...

Insert your chapter material here...

D.7 Numerical Model

Description of the numerical implementation of the models explained in Chapter 2...

D.8 Verification and Validation

Basic test cases to compare the implemented model against other numerical tools (verification) and experimental data (validation)...

Insert your chapter material here...

D.9 Problem Description

Description of the baseline problem...

D.10 Baseline Solution

Analysis of the baseline solution...

D.11 Enhanced Solution

Quest for the optimal solution...

D.11.1 Figures

Insert your section material and possibly a few figures...

Make sure all figures presented are referenced in the text!

Images

Make reference to Figures.

By default, the supported file types are *.png,.pdf,.jpg,.mps,.jpeg,.PNG,.PDF,.JPG,.JPEG*.

See `http://mactex-wiki.tug.org/wiki/index.php/Graphics_inclusion` for adding support to other extensions.

Drawings

Insert your subsection material and for instance a few drawings...

The schematic illustrated in Fig.can represent some sort of algorithm.

D.11.2 Equations

Equations can be inserted in different ways.

The simplest way is in a separate line like this

$$\frac{\mathrm{d}q_{ijk}}{\mathrm{d}t} + \mathcal{R}_{ijk}(\mathbf{q}) = 0\,. \tag{D.1}$$

If the equation is to be embedded in the text. One can do it like this $\partial\mathcal{R}/\partial\mathbf{q} = 0$.

It may also be split in different lines like this

D.11.3 Tables

Insert your subsection material and for instance a few tables...

Make sure all tables presented are referenced in the text!

Follow some guidelines when making tables:

D.11.4 Mixing

If necessary, a figure and a table can be put side-by-side as in Fig.

Insert your chapter material here...

D.12 Achievements

The major achievements of the present work...

D.13 Future Work

A few ideas for future work...

In case an appendix if deemed necessary, the document cannot exceed a total of 100 pages...

Some definitions and vector identities are listed in the section below.

D.14 Vector identities

$$\nabla \times (\nabla\phi) = 0 \tag{D.2}$$

$$\nabla \cdot (\nabla \times \mathbf{u}) = 0 \tag{D.3}$$