TÉCNICO LISBOA

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.

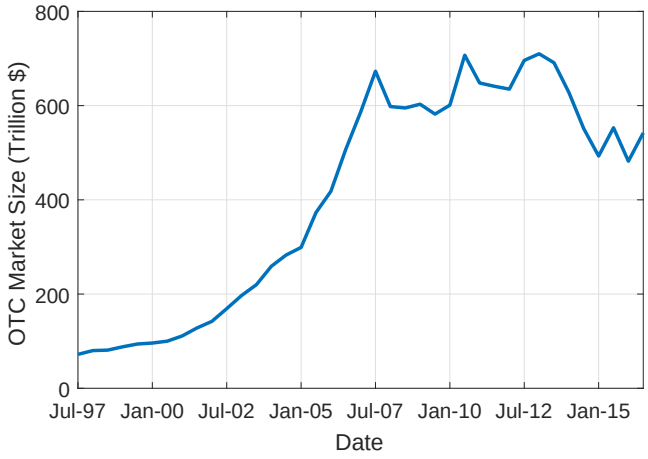


Figure 1.1: Size of OTC derivatives market since May 1996.

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. Not only will we approach the two main types of options - European and American - but we will also shortly introduce other less common types, commonly referred to as Exotic options, focusing particularly on Barrier 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.

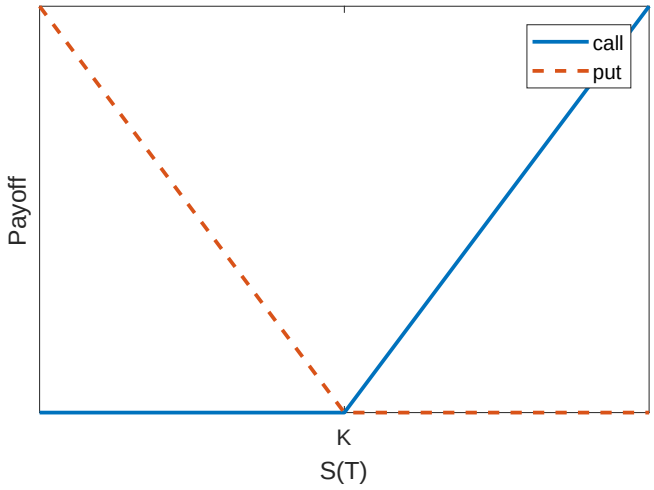


Figure 2.1: Payoff functions of European *call* and *put options*.

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 than or equal to the prices of similar European options. The reason behind this is the fact that with European contracts our exercising decision is restricted to a single day, whereas with American options we have that same day as well as several others 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 ideal 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 one of the most common types of Exotic options - *Barrier* options - though many others exist.

Barrier Options

A *Barrier option* behaves similarly to a European option with the difference that it only becomes active (or becomes worthless) 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 during this period, the option's payoff would be similar to that of its European equivalent. The payoff function of this type of option is therefore given by

$$\begin{aligned}
\text{Payoff}_{Barr,\ call}(K,T) &= \begin{cases} \max\left(S(T)-K,0\right), & \text{if } \exists\, t<T:\ S(t)>B \\ 0, & \text{otherwise} \end{cases}; \\
\text{Payoff}_{Barr,\ put}(K,T) &= \begin{cases} \max\left(K-S(T),0\right), & \text{if } \exists\, t<T:\ S(t)>B \\ 0, & \text{otherwise} \end{cases}.
\end{aligned} \tag{2.3}$$

Though Exotic options are used by banks and investors everyday, we will mainly focus on European options: not only are these the most common type of option traded, as we mentioned before, but data is also readily available for many different maturities and strike prices, making European options ideal for model calibration and validation, which will be the main goal of this thesis. Barrier options will nonetheless be implemented and studied, though no benchmark will be used to verify the models' validity in this case.

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), 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

$$\text{Price}(K, t^*) = e^{-rt^*}\mathbb{E}\left[\text{Payoff}(K, t^*)\right], \tag{2.4}$$

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. In particular, with eqs.(2.1) in mind, the price functions of European call and put options is clearly given by

$$\begin{aligned} C(K,T)_{Euro} &= e^{-rT}\mathbb{E}\left[\max\left(S(T)-K,0\right)\right] = e^{-rT}\mathbb{E}\left[\left(S(T)-K\right)\mathbb{1}_{\{S(T)>K\}}\right]; \\ P(K,T)_{Euro} &= e^{-rT}\mathbb{E}\left[\max\left(K-S(T),0\right)\right] = e^{-rT}\mathbb{E}\left[\left(K-S(T)\right)\mathbb{1}_{\{S(T)<K\}}\right], \end{aligned} \tag{2.5}$$

with $C(K,T)$ and $P(K,T)$ being the values (i.e. prices) 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)

$$\boxed{\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0,} \tag{2.6}$$

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.

Proof Itô's Lemma can be applied to our option price V, which depends on the (stochastic) stock price S and time t, so that we obtain

$$dV = \frac{\partial V}{\partial t}dt + \frac{\partial V}{\partial S}dS + \frac{1}{2}\frac{\partial^2 V}{\partial S^2}(dS)^2. \tag{2.7}$$

We now assume that the stock price S follows a geometric Brownian Motion,

$$dS = rSdt + \sigma SdW, \tag{2.8}$$

where W is a Brownian motion process.

It can be shown that $(dW)^2 = dt$ and $(dt)^2 = dt.dW = 0$. With these properties in mind, we can substitute eq.(2.8) into eq.(2.7), giving

$$\begin{aligned} dV =& \frac{\partial V}{\partial t}dt + \frac{\partial V}{\partial S}\left(rSdt + \sigma SdW\right) + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2}dt \\ =& \left(rS\frac{\partial V}{\partial S} + \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2}\right)dt + \sigma S\frac{\partial V}{\partial S}dW. \end{aligned} \tag{2.9}$$

We now construct a portfolio where we sell *one* option and buy an amount $\partial V/\partial S$ of stocks. The value of such a portfolio would be

$$\Pi = -V + \frac{\partial V}{\partial S}S. \tag{2.10}$$

From this equation we can easily derive the change of the portfolio's value, $d\Pi$, in the time interval dt as

$$\begin{aligned} d\Pi =& -dV + \frac{\partial V}{\partial S}dS \\ =& \left(\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2}\right)dt. \end{aligned} \tag{2.11}$$

Because the portfolio's value doesn't change with the Brownian motion W, it follows that it is *riskless*. To avoid any arbitrage possibilities (i.e. making profit without risk, which is usually impossible), the value of this portfolio must be the same as a risk-free asset. The value of a portfolio with such an asset would change with time as

$$d\Pi = r\Pi dt. \tag{2.12}$$

Substituting eqs. (2.10) and (2.11) into eq. (2.12) gives

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2\frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0. \tag{2.13}$$

□

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.(2.6)), 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 Chapter 3, so 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 behavior, 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 set dividend payment to zero in all our models.

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

$$dS(t) = rS(t)dt + \sigma S(t)dW(t), \tag{2.14}$$

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.

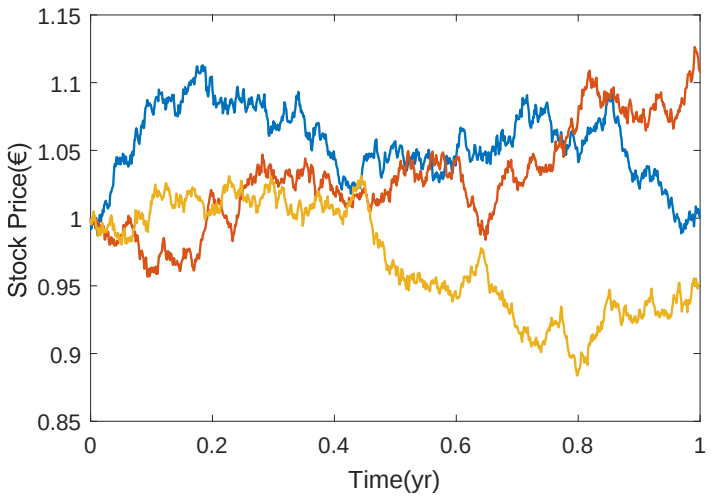


Figure 2.2: Example of three Geometric Brownian Motion processes with maturity $T = 1\,\text{yr}$, interest rate $r = 0.01\,\text{yr}^{-1}$, volatility $\sigma = 0.1\,\text{yr}^{-1/2}$ and initial stock price $S_0 = 1\,€$.

With this result, pricing options is fairly straightforward - we simply need to solve the PDE in eq.(2.6) 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

$$\begin{aligned} C &= N(d_1)S_0 - N(d_2)Ke^{-rT}; \\ P &= -N(-d_1)S_0 + N(-d_2)Ke^{-rT}, \end{aligned} \tag{2.15}$$

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

$$\begin{aligned} d_1 &= \frac{1}{\sigma\sqrt{T}}\left[\ln\left(\frac{S_0}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)T\right]; \\ d_2 &= d_1 - \sigma\sqrt{T}. \end{aligned} \tag{2.16}$$

This is a very important result that can be used to precisely price European options, so long as all the parameters are exactly known (which never occurs).

From eq.(2.15) we can also derive the relationship between C and P, known as the *put-call parity*

$$C = S_0 - Ke^{-rT} + P. \tag{2.17}$$

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.

Finally, we can obtain a lower bound for the prices of European call and put options. Using some non-arbitrage arguments it can be shown that

$$\begin{aligned} C &\geq \max\left[S_0 - Ke^{-rT}\right]; \\ P &\geq \max\left[Ke^{-rT} - S_0\right]. \end{aligned} \tag{2.18}$$

Chapter 3

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 3.1, 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).

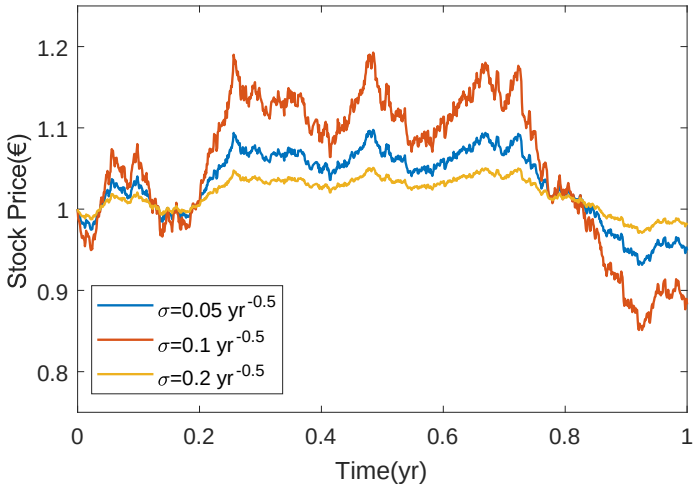


Figure 3.1: Example of three identical GBM processes with maturity $T = 1\,\mathrm{yr}$, interest rate $r = 0.01\,\mathrm{yr}^{-1}$ and initial stock price $S_0 = 1\,€$. The volatilities are $\sigma = 0.05\,\mathrm{yr}^{-1/2}$, $\sigma = 0.1\,\mathrm{yr}^{-1/2}$ and $\sigma = 0.2\,\mathrm{yr}^{-1/2}$ 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.

Of all the parameters in the BS formula (eq.(2.6)), 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.

Usually, volatility is estimated from the standard deviation of the rate of returns [3]. We begin by

measuring the stock price at fixed time intervals (e.g. daily, monthly), such that S_i corresponds to the stock price at the end of the ith interval. We define the return rate, u_i, as

$$u_i = \log\left(\frac{S_i}{S_{i-1}}\right). \tag{3.1}$$

We can then calculate the standard deviation of this rate, s, with the classical formula

$$s = \sqrt{\frac{1}{n-1}\sum_{i=1}^{n}(u_i - \overline{u})^2}, \tag{3.2}$$

assuming we have $n+1$ observations and denoting $\overline{u}$ as the average value of the return rates. The volatility (measured yearly) is given by

$$\sigma = \frac{s}{\sqrt{\tau}} \tag{3.3}$$

where τ defines the time interval length measured in years. The volatility can also be measured in other time periods: we can define a monthly or a daily volatility, but these are less common and will therefore not be used.

Our goal is to model the volatility of any given stock price. We begin by introducing the implied volatility, crucial to fully grasp the concepts used later. Afterwards, we will introduce four models to replicate market volatility: one with local volatility (Dupire's formula) and three with stochastic volatility (Heston, Static/Dynamic SABR). Other very common models exist. The GARCH model (Generalized Autoregressive Conditional Heteroskedasticity) along with all its many variations (EGARCH, NGARCH, ...) is particularly popular among econometricians. However, this model is mostly used to forecast volatility, and performs poorly when used to price derivatives [12]. Because pricing is our objective, GARCH will not be covered in this work. We will also study the constant volatility model, as used by Black *et al.*, as a benchmark for the quality of our models.

3.1 Implied Volatility

Implied volatility can be described as the value of stock price volatility that, when input into the BS pricer in eq.(2.15), 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).

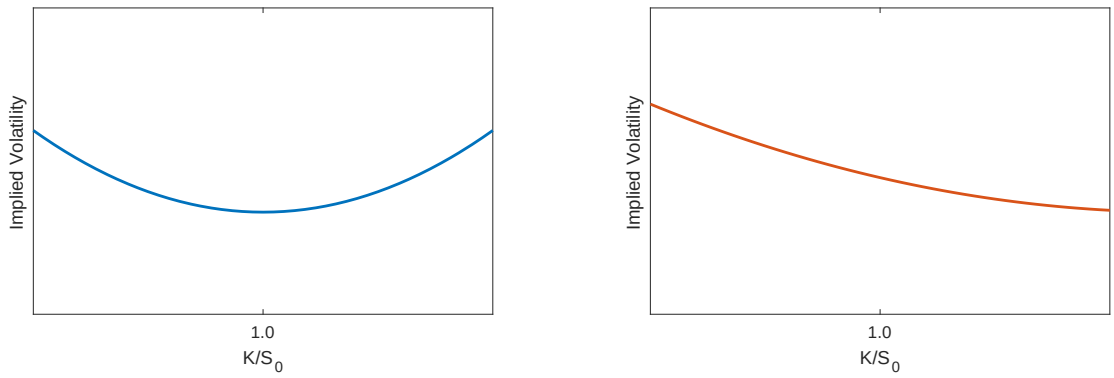
Because eq.(2.15) is not invertible, we need to use some numerical procedure (e.g. Newton's method) to find the value of implied volatility that matches the market and model prices, i.e. we must find, numerically, the solution to the equation

$$C(\sigma_{imp}) - C_{\text{mkt}} = 0, \tag{3.4}$$

where $C(\sigma_{imp})$ corresponds to the result of eq.(2.15) using σ_{imp} as (implied) volatility and C_{mkt} to the price of the option observed in the market.

Because eq.(2.15) 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 [13], as is indeed the case for the data we will use later.

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 3.2.



(a) Smile (b) Skew

Figure 3.2: Representation of the implied volatility smile (left) and skew (right) functions.

Because of their higher implied volatility, we can conclude that, if we observe a smile in the data, 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 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 in the data instead of a skew, and vice-versa, is determined by the type of product serving as underlying asset - Forex market options usually exhibit volatility smiles whereas index and commodities options usually show 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.

We saw in eq.(3.4) that the implied volatility depends on the option's market price. As we will see in later sections, it is quite important to visualize this relation. Therefore we present in Figure 3.3 the implied volatility functions w.r.t. the option's market price for three different strike prices.

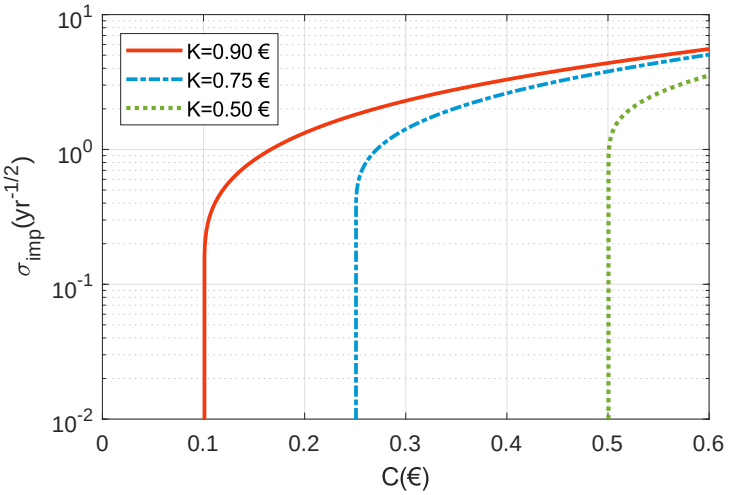


Figure 3.3: Relationship between the implied volatility and the price of a call option. The parameters used were $T = 21$ days, interest rate $r = 0.01\,\mathrm{yr}^{-1}$ and initial stock price $S_0 = 1$€. The stike prices are $K = 0.9$€, $K = 0.75$€ and $K = 0.5$€ for the red full line, the blue dot-dashed line and the green dot-dashed line, respectively.

We can see that the implied volatility goes to zero extremely fast when we approach the lower bound of the call option's price, $S_0 - Ke^{-rT}$ (from eq.(2.18)) - the lower bounds for the option prices with strikes $K = 0.9$€, $K = 0.75$€ and $K = 0.5$€ are, respectively, $C = 0.1007$€, $C = 0.2506$€ and $C = 0.5004$€). We can also observe that this property is more pronounced for lower strikes. This observation will become very important in later sections.

3.2 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 [14]. 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 due to expectancy,

and volatilities go down.

The constant volatility 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 section 3.1, 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. 2.14 is transformed into

$$dS(t) = rS(t)dt + \sigma(S,t)S(t)dW(t), \tag{3.5}$$

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

This local volatility model implies that we have a nonlinear *deterministic* volatility surface, $\sigma(S,t)$, which can be thought of as the market's expectation of future volatility at time t if the stock price is S at that time.

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.

3.2.1 Dupire's model

One of the most famous results in the modelling of the local volatility function was obtained by Dupire [15]. In his article, this author derives a theoretical formula for $\sigma(S,t)$, given by

$$\sigma(S,t) = \sqrt{\frac{\frac{\partial C}{\partial T} + rS\frac{\partial C}{\partial K}}{\frac{1}{2}S^2\frac{\partial^2 C}{\partial K^2}}}, \tag{3.6}$$

where $C = C(K,T)$ is the price of an European call option with strike price K and maturity T. All the derivatives are evaluated at $K = S$ and $T = t$.

Proof

We begin 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

$$\begin{aligned} C(K,T) = e^{-r(T-t)}\mathbb{E}\left[\max\left(S'-K,0\right)\right] &= e^{-r(T-t)}\int_0^\infty \max\left(S'-K,0\right)p(S,t,S',T)dS' \\ &= e^{-r(T-t)}\int_K^\infty (S'-K)p(S,t,S',T)dS'. \end{aligned} \tag{3.7}$$

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

$$\frac{\partial C}{\partial K} = -e^{-r(T-t)}\int_K^\infty p(S,t,S',T)dS'. \tag{3.8}$$

The second derivative results in

$$\frac{\partial^2 C}{\partial K^2} = e^{-r(T-t)} p(S, t, K, T). \tag{3.9}$$

assuming $p(S, t, \infty, T) = 0$.

We now make a brief detour to derive the Fokker-Planck equation (following the procedure shown in [13]), which we require for the next steps. We begin by assuming a trinomial model: at each time step, our stock price can only increase or decrease by an amount dS, with probabilities ϕ^+ and ϕ^-, respectively, or remain the same, with probability $(1 - \phi^+ - \phi^-)$. The expected value of the change is therefore

$$\phi^+ dS + (1 - \phi^+ - \phi^-).0 + \phi^-(-dS) = dS(\phi^+ - \phi^-), \tag{3.10}$$

and its variance is approximately given by

$$(dS)^2(\phi^+ + \phi^- - (\phi^+ - \phi^-)^2) \approx (dS)^2(\phi^+ + \phi^-) \tag{3.11}$$

where we take only the first order terms. The expected value of the change of a GBM can be thought of as its drift term (i.e. $rSdt$), and the variance is its stochastic term squared (i.e. $(\sigma S dW)^2 = \sigma^2 S^2 dt$), so that we have

$$\begin{aligned} rSdt &= (\phi^+ - \phi^-)dS; \\ \sigma^2 S^2 dt &= (\phi^+ + \phi^-)(dS)^2, \end{aligned} \tag{3.12}$$

from which we can derive the probabilities ϕ^+ and ϕ^- as

$$\begin{aligned} \phi^+ &= \frac{1}{2}\frac{dt}{dS^2}(rSdS + \sigma^2 S^2); \\ \phi^- &= \frac{1}{2}\frac{dt}{dS^2}(-rSdS + \sigma^2 S^2). \end{aligned} \tag{3.13}$$

We can now move backwards and derive the probability of reaching the price S' at time t' having started at the previous time $t' - dt$ with some (unknown) price S, which could be either $S' + dS$, $S' - dS$ or S' (assuming a trinomial movement). This probability is indeed the probability density, $p(S, t, S', T)$ and can be derived as

$$p(S, t, S', T) = \phi^- p(S' + dS, t - dt, S', T) + (1 - \phi^+ - \phi^-)(S', t - dt, S', T) + \phi^+ p(S' - dS, t - dt, S', T). \tag{3.14}$$

If we expand each term on its Taylor series around point (S', t'), we get

$$\begin{aligned} p(S, t, S', T) = \phi^- &\left(p(S', t, S', T) + dS\frac{dp}{dS'} + \frac{1}{2}(dS)^2\frac{d^2 p}{dS^2} - dt\frac{dp}{dt'} \right) \\ &+ (1 - \phi^+ - \phi^-)\left(p(S', t, S', T) - dt\frac{dp}{dt'} \right) \\ &+ \phi^+ \left(p(S', t, S', T) - dS\frac{dp}{dS'} + \frac{1}{2}(dS)^2\frac{d^2 p}{dS^2} - dt\frac{dp}{dt'} \right) \\ = &\frac{1}{2}\frac{dt}{dS'^2}\sigma^2 S'^2 (dS)^2\frac{d^2 p}{dS'^2} - \frac{dt}{dS'^2} rS'(dS)^2\frac{dp}{dS'}, \end{aligned} \tag{3.15}$$

from which we can derive the famous Fokker-Planck equation (with rather sloppy notation) as

$$\frac{\partial p}{\partial T} = \frac{1}{2}\frac{\partial^2(\sigma^2 S'^2 p)}{\partial S'^2} - \frac{\partial(rS'p)}{\partial S'}. \tag{3.16}$$

where we have used $t' = T$.

From eq. 3.7 we can easily derive

$$\frac{\partial C}{\partial T} = -rC + e^{-r(T-t)}\int_K^\infty (S'-K)\frac{\partial p}{\partial T}dS'. \tag{3.17}$$

Using eq. 3.16, we can transform this relation into

$$\frac{\partial C}{\partial T} = -rC + e^{-r(T-t)}\int_K^\infty (S'-K)\left(\frac{1}{2}\frac{\partial^2(\sigma^2 S'^2 p)}{\partial S'^2} - r\frac{\partial(S'p)}{\partial S'}\right)dS'. \tag{3.18}$$

We now split the terms in the integral to evaluate them independently. We integrate the second term by parts as

$$\begin{aligned}\int_K^\infty (S'-K)\left(-r\frac{\partial(S'p)}{\partial S'}\right)dS' &= -r(S'-K)(S'p)\big|_{S'=K}^\infty + r\int_K^\infty S'pdS' \\ &= r\int_K^\infty (S'-K)pdS' + rK\int_K^\infty pdS' \\ &= e^{r(T-t)}rV - rK\frac{\partial C}{\partial K}\end{aligned} \tag{3.19}$$

where in the first step we assumed that p and its first derivative w.r.t. S go sufficiently fast to zero and in last step we used eqs.(3.7) and (3.8).

Turning now to the first term of the integral, we integrate twice by parts

$$\begin{aligned}\int_K^\infty (S'-K)\left(\frac{1}{2}\frac{\partial^2(\sigma^2 S'^2 p)}{\partial S'^2}\right)dS' &= \frac{1}{2}(S'-K)\frac{\partial(\sigma^2 S'^2 p)}{\partial S'}\bigg|_{S'=K}^\infty - \frac{1}{2}\int_K^\infty \left(\frac{\partial(\sigma^2 S'^2 p)}{\partial S'}\right)dS' \\ &= \frac{1}{2}\sigma^2 S'^2 p\big|_{S'=K}^\infty = \frac{1}{2}\sigma^2(K,T)K^2 p(S,t,K,T) \\ &= \frac{1}{2}e^{r(T-t)}\sigma^2(K,T)K^2\frac{\partial^2 C}{\partial K^2},\end{aligned} \tag{3.20}$$

where in the last step we used eq.(3.9) and again assumed that p goes to zero sufficiently fast.

Collecting all the terms, we obtain

$$\frac{\partial C}{\partial T} = \frac{1}{2}\sigma^2(K,T)K^2\frac{\partial^2 C}{\partial K^2} - rK\frac{\partial C}{\partial K}. \tag{3.21}$$

from which, by rearranging all the terms and applying the variable change $\sigma(K,T) \implies \sigma(S,t)$, we get

$$\sigma(S,t) = \sqrt{\frac{\dfrac{\partial C}{\partial T} + rS\dfrac{\partial C}{\partial K}}{\dfrac{1}{2}S^2\dfrac{\partial^2 C}{\partial K^2}}}. \tag{3.22}$$

assuming all the derivatives are evaluated at $K = S$ and $T = t$.

□

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.(3.6) to obtain the local volatility surface. We can then sample the local volatility at each time step of our simulation.

A major problem can be pointed out in eq.(3.6). For options far in or far out of the money (i.e. with strikes much greater or much smaller that S_0), it can be shown that the option price depends almost linearly on the strike. This means that the second derivative of the price w.r.t. the strike is very small in these regions. Because this value is in the denominator of eq.(3.6), the local volatility will explode in such cases, which is unrealistic.

One possible solution to this problem is to relate our local volatility with the implied volatility surface instead of the option price's [16]. The relation obtained is

$$\boxed{\sigma(S,t) = \sqrt{\frac{\sigma_{imp}^2 + 2t\sigma_{imp}\frac{\partial \sigma_{imp}}{\partial T} + 2rSt\sigma_{imp}\frac{\partial \sigma_{imp}}{\partial K}}{\left(1 + Sd_1\sqrt{t}\frac{\partial \sigma_{imp}}{\partial K}\right)^2 + S^2 t\sigma_{imp}\left(\frac{\partial^2 \sigma_{imp}}{\partial K^2} - d_1\left(\frac{\partial \sigma_{imp}}{\partial K}\right)^2\sqrt{t}\right)}},} \tag{3.23}$$

where d_1 is given by

$$d_1 = \frac{\log(S_0/S) + \left(r + \frac{1}{2}\sigma_{imp}^2\right)t}{\sigma_{imp}\sqrt{t}}, \tag{3.24}$$

with S_0 being the stock price at $t = 0$. We define $\sigma_{imp} = \sigma_{imp}(K,T)$ as the implied volatilities of options with maturity T, and strike K. Furthermore, σ_{imp} and all its derivatives are evaluated at $K = S$ and $T = t$. This formula can be obtained from eq.(3.6) by applying the transformation from call prices to implied volatilities.

We now need to generate the implied volatility surface from market data in order to find the gradients shown in eq.(3.23). We again need to interpolate between the market's implied volatilities in order to find this surface. However, because we only have access to market data for a finite set implied volatilities, finding the implied volatility surface requires some form of interpolation and extrapolation. This problem is therefore very much ill-posed (i.e. a small change in the input generates a very different output) and the local volatility surface might look unrealistic [13]. Furthermore, it depends heavily on the interpolation/extrapolation method chosen, which is problematic.

Because we are directly using the market data, the implied volatility surface is nonparametric and therefore no fitting procedure is required. To ease the problem of ill-posedness, we could heuristically choose some functions to model this surface, fitting their parameters to better replicate the market data, and finally replacing these functions in eq.(3.23), as done by Dewynne [17]. However, this solution depends heavily on the functions chosen and will not be considered.

Three other problems can be identified in Dupire's local volatility assumption, however. First, it can

be shown that the local volatility surface changes with time [13]. This means that the whole interpolation procedure must be done regularly in order for the model to work properly. Secondly, some authors have pointed out that the volatility smile obtained from Dupire's local-volatility model doesn't follow real market dynamics [18]: 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. Finally, the gradients used in eq.(3.23) have to be generated numerically. This numerical differentiation is very unstable, especially when done on our rough interpolated surface, which might lead to errors in the local volatility obtained.

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

3.3 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.

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

$$dS(t) = rS(t)dt + \sigma(S,t)S(t)dW_1(t), \tag{3.25}$$

but we further hypothesize that the volatility follows

$$d\sigma(S,t) = p(S,\sigma,t)dt + q(S,\sigma,t)dW_2(t), \tag{3.26}$$

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.

$$dW_1 dW_2 = \rho dt. \tag{3.27}$$

This correlation factor ρ can be explained by the relationship between prices and volatilities [12]. Historically, we can see that high volatility periods usually occur when the market is under stress due to low returns (i.e. stock prices decrease). Examples of such periods occurred during the Gulf War in 1991 and during the invasion of Iraq in 2003. On the other hand, whenever the market stabilizes and returns increase, the volatility goes down. These factors seem to indicate the existence of a negative correlation between stock prices and volatilities. Thus, to fully grasp market behavior, this correlation must be taken into account.

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.

Many stochastic volatility models exist, such as Hull-White [19] and Stein-Stein [20]. However, the *Heston* model is by far the most popular of these [12]. Another model, known as *SABR*, is also widely used by practitioners, especially in the interest rate derivative markets (i.e. derivatives whose underlying asset is an interest rate). For these reasons, both these models will be studied in detail in this work.

3.3.1 Heston Model

One of the most popular stochastic volatility models is known as *Heston model*. It was developed in 1993 by Steven Heston [21] and it states that stock prices satisfy the relations

$$\boxed{dS = rSdt + \sqrt{\nu}SdW_1,} \tag{3.28}$$

$$\boxed{d\nu = \kappa(\overline{\nu} - \nu)dt + \eta\sqrt{\nu}dW_2,} \tag{3.29}$$

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. We furthermore define ν_0 as the initial variance (i.e. variance at time $t = 0$). 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 [22].

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

$$\begin{aligned} C_H(\theta; K, T) &= e^{-rT}\mathbb{E}\left[(S(T) - K)\,\mathbb{1}_{\{S(T)>K\}}\right] \\ &= e^{-rT}\left(\mathbb{E}\left[S(T)\mathbb{1}_{\{S(T)>K\}}\right] - K\mathbb{E}\left[\mathbb{1}_{\{S(T)>K\}}\right]\right) \\ &= S_0P_1 - e^{-rT}KP_0, \end{aligned} \tag{3.30}$$

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

$$P_1 = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left(\frac{e^{-iu\log K}}{iuS_0e^{rT}}\phi(\theta; u - i, T)\right)du, \tag{3.31}$$

$$P_0 = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left(\frac{e^{-iu\log K}}{iu}\phi(\theta; u, T)\right)du, \tag{3.32}$$

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.(3.31) and (3.32) and with them find the option price with eq.(3.30). In his original article, Heston proposed a solution to this very characteristic function [21]. However, some posterior authors demonstrated that, for large maturities, some discontinuities appeared for the proposed equation [23]. One possible alternative, proposed by Schoutens [24], avoids this shortcoming and is given by

$$\phi(\theta; u, t) = \exp\left\{iu\left(\log S_0 + rt\right) + \frac{\kappa\overline{\nu}}{\eta^2}\left[(\xi - \alpha)\, t - 2\log\left(\frac{1 - ge^{-\alpha t}}{1 - g}\right)\right] + \frac{\nu_0}{\eta^2}\left(\xi - \alpha\right)\frac{1 - e^{-\alpha t}}{1 - ge^{-\alpha t}}\right\}, \tag{3.33}$$

where we define

$$\xi = \kappa - \eta\rho i u, \tag{3.34}$$

$$\alpha = \sqrt{\xi^2 + \eta^2(u^2 + iu)}, \tag{3.35}$$

$$g = \frac{\xi - \alpha}{\xi + \alpha}. \tag{3.36}$$

Proof We will follow the derivation presented in Gatheral [25] with slight changes of variables, for consistency.

We begin by defining the (call) option's value equation (from Itô's Lemma) as

$$\frac{\partial C}{\partial t} + \frac{1}{2}\nu S^2\frac{\partial^2 C}{\partial S^2} + \rho\eta\nu S\frac{\partial^2 C}{\partial\nu\partial S} + \frac{1}{2}\eta^2\nu\frac{\partial^2 C}{\partial\nu^2} + rS\frac{\partial C}{\partial S} - rC + \kappa(\overline{\nu} - \nu)\frac{\partial C}{\partial\nu} = 0. \tag{3.37}$$

We now define the forward price as $F(t) = S(t)e^{r(T-t)}$, and we introduce the variables $\tau = T - t$ and $x(t) = \log\left(F(t)/K\right)$. We furthermore denote C^* as the future value to expiration of the option price (i.e. $C^* = Ce^{rT}$). The relation in eq. (3.37) reduces to

$$-\frac{\partial C^*}{\partial\tau} + \frac{1}{2}\nu\frac{\partial^2 C^*}{\partial x^2} - \frac{1}{2}\nu\frac{\partial C^*}{\partial x} + \frac{1}{2}\eta^2\nu\frac{\partial^2 C^*}{\partial\nu^2} + \rho\eta\nu\frac{\partial^2 C^*}{\partial x\partial\nu} + \kappa(\overline{\nu} - \nu)\frac{\partial C^*}{\partial\nu} = 0, \tag{3.38}$$

which has a solution of the form [26]

$$C^*(x, \nu, \tau) = Ce^{rT} = \left(S_0 P_1 - e^{-rT}KP_0\right)e^{rT} = K\left[e^x P_1(x, \nu, \tau) - P_0(x, \nu, \tau)\right]. \tag{3.39}$$

Substituting the solution (3.39) in eq. (3.38) implies that P_0 and P_1 must satisfy

$$-\frac{\partial P_j}{\partial\tau} + \frac{1}{2}\nu\frac{\partial^2 P_j}{\partial x^2} - \left(\frac{1}{2} - j\right)\nu\frac{\partial P_j}{\partial x} + \frac{1}{2}\eta^2\nu\frac{\partial^2 P_j}{\partial\nu^2} + \rho\eta\nu\frac{\partial^2 P_j}{\partial x\partial\nu} + \left(\kappa(\overline{\nu} - \nu) + j\nu\rho\eta\right)\frac{\partial P_j}{\partial\nu} = 0, \tag{3.40}$$

for $j = 0, 1$ and subject to the terminal condition $P_j(x, \nu, \tau = 0) = \mathbb{1}_{\{x>0\}}$.

To solve this problem we use a Fourier transform technique. Defining $\tilde{P}_j$ as the Fourier transform of P_j,

$$\tilde{P}_j(u, \nu, \tau) = \int_{-\infty}^{\infty} e^{-iux} P_j(x, \nu, \tau)dx, \tag{3.41}$$

and substituting this result in eq. (3.40) produces

$$\nu \left\{ \omega_j \tilde{P}_j - \beta_j \frac{\partial \tilde{P}_j}{\partial \nu} + \frac{\eta^2}{2} \frac{\partial^2 \tilde{P}_j}{\partial \nu^2} \right\} + \kappa \overline{\nu} \frac{\partial \tilde{P}_j}{\partial \nu} - \frac{\partial \tilde{P}_j}{\partial \tau} = 0, \tag{3.42}$$

with

$$\omega_j = -\frac{u^2}{2} - \frac{iu}{2} + iju, \tag{3.43}$$

$$\beta_j = \kappa - \rho\eta iu - \rho\eta j. \tag{3.44}$$

Heston proposed that $\tilde{P}_j$ is a function of the form

$$\tilde{P}_j(u, \nu, \tau) = \frac{1}{iu} \exp\left\{ \zeta_j(u, \tau)\overline{\nu} + \psi_j(u, \tau)\nu \right\}. \tag{3.45}$$

We can thus derive

$$\frac{\partial \tilde{P}_j}{\partial \tau} = \left\{ \overline{\nu} \frac{\partial \zeta_j}{\partial \tau} + \nu \frac{\partial \psi_j}{\partial \tau} \right\} \tilde{P}_j, \tag{3.46}$$

$$\frac{\partial \tilde{P}_j}{\partial \nu} = \psi_j \tilde{P}_j, \tag{3.47}$$

$$\frac{\partial^2 \tilde{P}_j}{\partial \nu^2} = \psi_j^2 \tilde{P}_j. \tag{3.48}$$

Thus, eq. (3.42) is only satisfied if

$$\frac{\partial \zeta_j}{\partial \tau} = \kappa \psi_j, \tag{3.49}$$

$$\frac{\partial \psi_j}{\partial \tau} = \omega_j - \beta_j \psi_j + \frac{\eta^2}{2} \psi_j^2. \tag{3.50}$$

Integrating these solutions with terminal conditions $\zeta_j(u, 0) = \psi_j(u, 0) = 0$, we obtain

$$\zeta_j(u, \tau) = \frac{\kappa}{\eta^2} \left[(\beta_j - \gamma_j)\tau - 2\log\left(\frac{1 - g_j e^{-\gamma_j \tau}}{1 - g_j} \right) \right], \tag{3.51}$$

$$\psi_j(u, \tau) = \frac{1}{\eta^2} (\beta_j - \gamma_j) \frac{1 - e^{-\gamma_j \tau}}{1 - g_j e^{-\gamma_j \tau}} \tag{3.52}$$

where we define

$$\gamma_j = \sqrt{\beta_j - 2\omega_j \eta^2}, \tag{3.53}$$

$$g_j = \frac{\beta_j - \gamma_j}{\beta_j + \gamma_j}. \tag{3.54}$$

We finally arrive at the solution for P_j given by

$$P_j(x, \nu, \tau) = \frac{1}{2} + \frac{1}{\pi} \int_0^\infty \mathrm{Re}\left(\frac{\exp\left\{ \zeta_j(u, \tau)\overline{\nu} + \psi_j(u, \tau)\nu + iux \right\}}{iu} \right) du \tag{3.55}$$

A change of variables is possible, as shown by Crisostomo [27], such that the two different charac-

teristic functions for P_1 and P_0 can be transformed into a single one by defining

$$P_1 = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left(\frac{e^{-iu\log K}}{iuS_0e^{rT}}\phi(\theta; u-i, T)\right)du, \tag{3.56}$$

$$P_0 = \frac{1}{2} + \frac{1}{\pi}\int_0^\infty \mathrm{Re}\left(\frac{e^{-iu\log K}}{iu}\phi(\theta; u, T)\right)du, \tag{3.57}$$

with the characteristic function $\phi(\theta; u, T)$ given by

$$\phi(\theta; u, t) = \exp\left\{iu\left(\log S_0 + rt\right) + \frac{\kappa\overline{\nu}}{\eta^2}\left[\left(\xi-\alpha\right)t - 2\log\left(\frac{1-ge^{-\alpha t}}{1-g}\right)\right] + \frac{\nu_0}{\eta^2}\left(\xi-\alpha\right)\frac{1-e^{-\alpha t}}{1-ge^{-\alpha t}}\right\}. \tag{3.58}$$

□

The main problem with the characteristic function presented in eq.(3.33) is the fact that it is highly nonlinear. Because we will apply some optimization procedure to minimize the difference between the model and the market option prices (i.e. calibration), the optimizer is very likely become stuck in some local minimum and not find the globally optimal solution. This shortcoming led some authors to propose several modified versions of this function, such as Rollin *et al.* [28]. Most recently, Cui *et al.* [29] presented a characteristic function that not only doesn't have the previously mentioned discontinuities but also solves the nonlinearity problem, given by

$$\phi(\theta; u, t) = \exp\left\{iu\left(\log S_0 + rt\right) - \frac{t\kappa\overline{\nu}\rho iu}{\eta} - \nu_0 A + \frac{2\kappa\overline{\nu}}{\eta^2}D\right\}, \tag{3.59}$$

with A and D given by

$$A = \frac{A_1}{A_2}, \tag{3.60}$$

$$D = \log\alpha + \frac{(\kappa-\alpha)t}{2} - \log\left(\frac{\alpha+\xi}{2} + \frac{\alpha-\xi}{2}e^{-\alpha t}\right), \tag{3.61}$$

where ξ and α are given by eqs.(3.34) and (3.35), respectivelly, and where we introduce the variables A_1, A_2, given by

$$A_1 = (u^2 + iu)\sinh\frac{\alpha t}{2}, \tag{3.62}$$

$$A_2 = \alpha\cosh\frac{\alpha t}{2} + \xi\sinh\frac{\alpha t}{2}. \tag{3.63}$$

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 4.

One last consideration is required for the Heston model. By analyzing eqs.(3.28) and (3.29) we can see that the square root of the variance is used. This shouldn't pose a problem, since the variance of any process is always positive. However, because in our case this variable is itself stochastic, we must guarantee that it doesn't become negative. To ensure this, we can apply Feller's condition [30] and force

the parameters to obey

$$2\kappa\overline{\nu} > \eta^2. \tag{3.64}$$

However, this condition restricts the sample space of our model. One famous alternative is to set the variance to zero every time it becomes negative:

$$\nu = \max\left[\nu^*, 0\right], \tag{3.65}$$

where ν^* corresponds to the unrestricted variance and ν is the new (always positive) variance, ensuring that the square root outputs a real number. Because this alternative allows any value for the parameters, it will be adopted in the implementation of the Heston model.

3.3.2 Static SABR Model

One other very famous model for stochastic volatility was developed by Hagan *et al.* [18] and is known as *SABR* - we will henceforth refer to it by *Static SABR*, to distinguish it from the *Dynamic SABR* model that we will see next. SABR stands for "*stochastic-*$\alpha\beta\rho$" and in this model it is assumed that the option prices and volatilities follow [31]

$$\boxed{dS = rSdt + e^{-r(T-t)(1-\beta)}\sigma S^{\beta} dW_1,} \tag{3.66}$$

$$\boxed{d\sigma = \nu\sigma dW_2,} \tag{3.67}$$

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.

The parameters β and ν correspond, respectively to the *skewness* (i.e. how the volatility smile moves when the stock price changes) and the *volatility of volatility* (i.e. how erratic is the volatility process).

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 [18]. We will leave this parameter free when fitting the model to the market data, assuming no heuristics.

One of the main problems with the static 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 [32] proposed a mean-reverting correction to static SABR, but we will study the original model by Hagan *et al.*, as it is more commonly used. We should still note that while negative volatilities make no sense in the real world, by examining eq.(3.66) we can see that this should pose no problem upon simulations, since this effect would be equivalent to inverting the Brownian motion process (i.e. $dW \implies -dW$), which is obviously allowed.

One of the main reasons why static 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 [33], it can be shown that these implied volatilities are given by

$$\begin{aligned}\sigma_{SABR}(K,f,T) \approx & \frac{1}{\left[1+\frac{(1-\beta)^2}{24}\log^2\left(\frac{f}{K}\right)+\frac{(1-\beta)^4}{1920}\log^4\left(\frac{f}{K}\right)\right]}\cdot\left(\frac{\nu\log\left(f/K\right)}{x(z)}\right) \\ & \cdot\left\{1+T\left[\frac{(1-\beta)^2}{24}\frac{\alpha^2}{(Kf)^{1-\beta}}+\frac{1}{4}\frac{\rho\beta\nu\alpha}{(Kf)^{(1-\beta)/2}}+\frac{2-3\rho^2}{24}\nu^2\right]\right\},\end{aligned} \tag{3.68}$$

with z and $x(z)$ given by

$$z=\frac{\nu\left(f^{1-\beta}-K^{1-\beta}\right)}{\alpha(1-\beta)}, \tag{3.69}$$

$$x(z)=\log\left\{\frac{\sqrt{1-2\rho z+z^2}+z-\rho}{1-\rho}\right\}, \tag{3.70}$$

where we have used $f = S_0e^{rT}$. The proof of this solution is quite lengthy and will not be replicated here, but it can be found in Hagan [18].

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 4.

3.3.3 Dynamic SABR Model

One of the main setbacks of the static 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 [18].

To solve this problem, Hagan *et al.* suggested a similar model known as *Dynamic SABR* [18]. It follows the same processes presented in eqs.(3.66) and (3.67) but with time-dependent parameters $\rho(t)$ and $\nu(t)$,

$$\boxed{dS = rSdt + e^{-r(T-t)(1-\beta)}\sigma S^{\beta}dW_1,} \tag{3.71}$$

$$\boxed{d\sigma = \nu(t)\sigma dW_2,} \tag{3.72}$$

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 [34]. The resulting formula is given by

$$\sigma_{DynSABR}(K,f,T)=\frac{1}{\omega}\left(1+A_1(T)\log\left(\frac{K}{f}\right)+A_2(T)\log^2\left(\frac{K}{f}\right)+B(T)T\right), \tag{3.73}$$

where $f = S_0e^{rT}$, $\omega = f^{1-\beta}/\alpha$ and where $A_1(T)$, $A_2(T)$ and $B(T)$ are given by

$$A_1(T)=\frac{\beta-1}{2}+\frac{\eta_1(T)\omega}{2}, \tag{3.74}$$

$$A_2(T) = \frac{(1-\beta)^2}{12} + \frac{1-\beta-\eta_1(T)\omega}{4} + \frac{4\nu_1^2(T) + 3(\eta_2^2(T) - 3\eta_1^2(T))}{24}\omega^2, \tag{3.75}$$

$$B(T) = \frac{1}{\omega^2}\left(\frac{(1-\beta)^2}{24} + \frac{\omega\beta\eta_1(T)}{4} + \frac{2\nu_2^2(T) - 3\eta_2^2(T)}{24}\omega^2\right), \tag{3.76}$$

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

$$\nu_1^2(T) = \frac{3}{T^3}\int_0^T (T-t)^2\nu^2(t)dt, \tag{3.77}$$

$$\nu_2^2(T) = \frac{6}{T^3}\int_0^T (T-t)t\nu^2(t)dt, \tag{3.78}$$

$$\eta_1(T) = \frac{2}{T^2}\int_0^T (T-t)\nu(t)\rho(t)dt, \tag{3.79}$$

$$\eta_2^2(T) = \frac{12}{T^4}\int_0^T\int_0^t\left(\int_0^s \nu(u)\rho(u)du\right)^2 dsdt, \tag{3.80}$$

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.(3.77)-(3.80) are analytically solvable, greatly simplifying the calibration of this model. One classical example [35] corresponds to

$$\rho(t) = \rho_0 e^{-at}, \tag{3.81}$$

$$\nu(t) = \nu_0 e^{-bt}, \tag{3.82}$$

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

$$\nu_1^2(T) = \frac{6\nu_0^2}{(2bT)^3}\left[\left(\frac{(2bT)^2}{2} - 2bT + 1\right) - e^{-2bT}\right], \tag{3.83}$$

$$\nu_2^2(T) = \frac{12\nu_0^2}{(2bT)^3}\left[e^{-2bT}(1+bT) + bT - 1\right], \tag{3.84}$$

$$\eta_1(T) = \frac{2\nu_0\rho_0}{T^2(a+b)^2}\left[(a+b)T + e^{-(a+b)T} - 1\right], \tag{3.85}$$

$$\eta_2^2(T) = \frac{3\nu_0^2\rho_0^2}{T^4(a+b)^4}\left[e^{-2(a+b)T} - 8e^{-(a+b)T} + (7 + 2T(a+b)(-3 + (a+b)T))\right]. \tag{3.86}$$

There are other more robust solutions for $\rho(t)$ and $\nu(t)$, such as presented by Fernandez *et al.* [35], but these usually don't have analytically solvable integrals and need to be computed numerically. For this reason, these will not be considered in the present thesis.

Chapter 4

Implementation

4.1 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* [36].

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 [37].

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.

4.1.1 Simulating stock prices

As stated, to implement the Monte Carlo algorithm, one needs to simulate stock price paths. However, by analyzing eq.(2.14), we can see that the stock prices depend on a Brownian motion process which, due to its self-similarity, is not differentiable [38]. 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

$$dX(t) = a(X(t))dt + b(X(t))dW(t), \tag{4.1}$$

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

$$X_{n+1} = X_n + a(X_n)\Delta t + b(X_n)\Delta W_n, \quad n = 1, \ldots, N, \tag{4.2}$$

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.(2.14), we arrive at

$$S(t + \Delta t) = S(t) + rS(t)\Delta t + \sigma(S(t), t)S(t)\sqrt{\Delta t} Z. \tag{4.3}$$

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 [39], defined as

$$X_{n+1} = X_n + a(X_n)\Delta t + b(X_n)\Delta W_n + \frac{1}{2} b(X_n) b'(X_n)((\Delta W_n)^2 - \Delta t), \tag{4.4}$$

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

$$S(t + \Delta t) = S(t) + rS(t)\Delta t + S(t)\sqrt{\nu(t)}\sqrt{\Delta t} Z_1 + \frac{1}{2}\nu(t)S(t)\Delta t(Z_1^2 - 1), \tag{4.5}$$

$$\nu(t + \Delta t) = \nu(t) + \kappa(\overline{\nu} - \nu(t))\Delta t + \eta\sqrt{\nu(t)\Delta t} Z_2 + \frac{\eta^2}{4}\Delta t(Z_2^2 - 1), \tag{4.6}$$

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

$$\begin{aligned} S(t+\Delta t) =& S(t) + rS(t)\Delta t + e^{-r(T-t)(1-\beta)}\sigma(t)S^{\beta}(t)\sqrt{\Delta t}Z_1 + \\ &+ \frac{\beta}{2}e^{-2r(T-t)(1-\beta)}\sigma^2(t)S^{2\beta-1}(t)\Delta t(Z_1^2-1), \end{aligned} \tag{4.7}$$

$$\sigma(t+\Delta t) = \sigma(t) + \nu\sigma(t)\sqrt{\Delta t}Z_2 + \frac{\nu^2}{2}\sigma(t)\Delta t(Z_2^2-1), \tag{4.8}$$

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

$$\begin{aligned} &Z_1 \sim N(0,1); \\ &Z_2 = \rho Z_1 + \sqrt{1-\rho^2}Y, \end{aligned} \tag{4.9}$$

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 4.1 we can see how smaller subintervals better resemble a true GBM process, whereas if they become too large this resemblance practically vanishes.

4.1.2 Pricing options from simulations

To price options using the Monte Carlo algorithm, we generate M paths by recursively calculating $\{S_i(t),\ \ i = 1, \ldots, M\}$, 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 using eqs.(2.1)-(2.3). We then average all these results and discount them back to the present, obtaining the option's value

$$\begin{aligned} C(K,T) &= e^{-rT}\frac{1}{M}\sum_{i=1}^{M}\text{Payoff}_{call}(K,T); \\ P(K,T) &= e^{-rT}\frac{1}{M}\sum_{i=1}^{M}\text{Payoff}_{put}(K,T), \end{aligned} \tag{4.10}$$

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

Despite its versatility, the Monte Carlo method can be a very noisy estimator for the price of options,

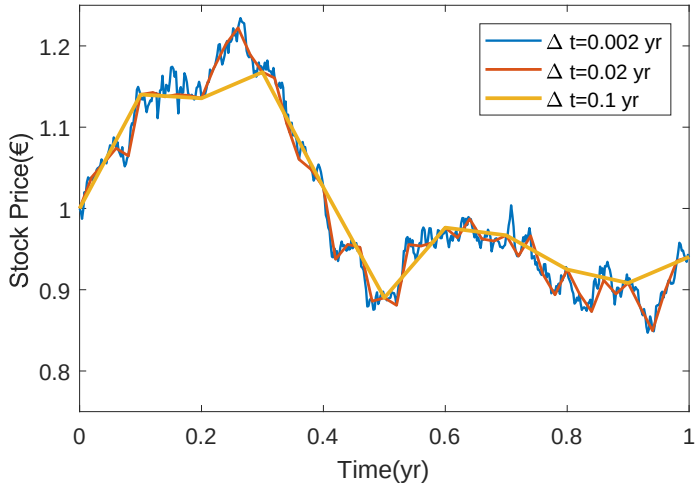


Figure 4.1: Effect of the subinterval size on the GBM discretization with maturity $T = 1\,\text{yr}$, interest rate $r = 0.01\,\text{yr}^{-1}$, volatility $\sigma = 0.1\,\text{yr}^{-1/2}$ and initial stock price $S_0 = 1\,€$. The subinterval size selected was $\Delta t = 0.002\,\text{yr}$, $\Delta t = 0.02\,\text{yr}$ and $\Delta t = 0.1\,\text{yr}$ 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.

in particular for options with very high strike prices. To understand this problem, notice that the payoff function of European options is zero for stocks below the strike. If we have a very large strike, very few of the simulated paths will actually cross the strike at maturity and contribute to the calculation of the option's price (all the other paths will contribute with a payoff of zero). The averaging procedure shown in eq.(4.10) will therefore be performed on an extremely small subset of paths, which will be noisy. We could counter this effect by simulating a larger and larger number of paths, so that we always have a significant number of paths above the strike, contributing to the option's price, though this comes at the cost of increased computation times. Furthermore, on the limit, there will always be a strike high enough that none of the simulated paths reach it, meaning that the option's value for that simulation would be zero. This does not hold in real life, since there is always a positive price for any option, even for extremely high strikes.

4.2 Surface Interpolation (Dupire)

To apply Dupire's local volatility model, as shown in eq.(3.23), we need to generate the implied volatility surface from the market data. Because we only have data for a finite set of maturities and strikes, the implied volatility surface must be obtained by some form of interpolation and extrapolation. From this interpolated surface we can also calculate the gradients needed for the local volatility formula.

Since the data we have is usually not evenly distributed in the sample space (e.g. we have strikes $K = 0.5, 0.75, 0.9$ and 1, which have different intervals between them), we must choose an interpolation algorithm that works in this case.

One possible method is known as *Delaunay triangulation* [40]. In short, this interpolation algorithm

generates a triangulation between points P_1, P_2 and P_3 if the circumscribed circle of these points contains no other points P_j inside. A scheme of this

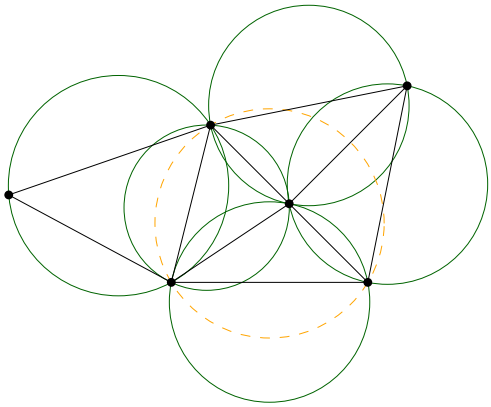


Figure 4.2: Example of a Delaunay triangulation, where we connect the points for which the circumscribed circles (green) don't contain any other points inside. A circle for which this property does not hold (orange dashed) is also represented, though no triangulation is shown.
Source: `https://www.ti.inf.ethz.ch/ew/Lehre/CG13/lecture/Chapter%206.pdf`

The function *scatteredInterpolant*, implemented in MATLAB, applies this very same algorithm, and will therefore be used in the implied volatility surface generation.

4.3 Model Calibration (Heston and SABR)

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

$$\text{Cost}(\theta) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,j}\left(C_{\text{market}}(T_i,K_j) - C_{\text{model}}(\theta;T_i,K_j)\right)^2, \tag{4.11}$$

where we denote θ as the model's parameter set, w_i corresponds to some weight function and where $C_{\text{model}}(\cdot)$ and $C_{\text{market}}(\cdot)$ 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

$$\boxed{\text{Cost}(\theta) = \sum_{i=1}^{n}\sum_{j=1}^{m} w_{i,j}\left(\sigma_{imp,\text{mkt}}(T_i,K_j) - \sigma_{imp,\text{mdl}}(\theta;T_i,K_j)\right)^2,} \tag{4.12}$$

where $\sigma_{imp,\text{mkt}}(\cdot)$ and $\sigma_{imp,\text{mdl}}(\cdot)$ 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

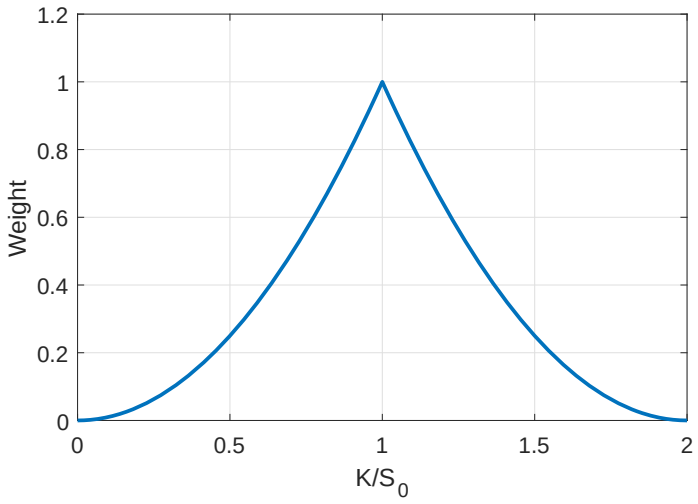
calibration procedure, for the reason stated before.

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

$$w_{i,j} = \left(1 - \left|1 - \frac{K_j}{S_0}\right|\right)^2, \tag{4.13}$$

where we assume strikes are restricted to $K < 2S_0$. This function is represented in Figure 4.3.



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 4.3: 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.(4.12)). 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.

We must also mention the fact that, while volatility models are commonplace in finance, investors use different models with different parameters to price their options. This means that even if our calibrated

model perfectly fits market data at a given time, after a while the prices will change and the calibration will become outdated. Therefore, a frequent recalibration of the parameters is required which demands fast calibration methods to be employed.

Fortunately, as mentioned, both Heston and SABR have closed-form solutions, shown in eqs. (3.30), (3.68) and (3.73), 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.

4.3.1 Optimization Algorithms

There are several possible methods to find the parameter set that minimizes the cost function shown in eq.(4.12). 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 a powerful algorithm known as *CMA-ES* [41] (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 this optimizer works. For detailed descriptions, the original source should be consulted.

The optimization algorithm 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.

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

$$N(\mathbf{x};\mathbf{m},\mathbf{C}) = \frac{1}{\sqrt{(2\pi)^D|\det\mathbf{C}|}}\exp\left(-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T\mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})\right), \tag{4.14}$$

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 B. For a more detailed explanation, as well as other aspects of the algorithm, see Hansen [42].

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 1.

Algorithm 1: 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.(B.5) and (B.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 [43]. Furthermore, unlike many other optimizers, the CMA-ES is almost non-parametric. It simply requires an starting guess, to generate the starting mean vector, and the algorithm is expected to converge to a global minimum. As for disadvantages, because we have to generate a set of samples at each iteration, if this generation process is slow, the convergence may stall significantly, particularly when many parameters are used in the model. Other algorithms may perform faster than CMA-ES, but CMA-ES is expected to outperform them in terms of precision.

This optimizer was implemented by Hansen in MATLAB (as well as in other computer languages) as a function named *purecmaes* [44] and will be used with only very slight changes, to account for the volatility models used used.

Chapter 5

Results

In this chapter we will implement all the models described before: constant volatility, Dupire's local volatility, Heston's stochastic volatility and both static and dynamic SABR models.

With the exception of Dupire's local volatility, will perform the calibration described in section 4.3 for all models: we will minimize the cost function in eq. (4.12) with the weight function shown in eq. (4.13). The model values for the implied volatilities, $\sigma_{imp,\mathrm{mdl}}$, will be obtained through the closed form solutions for each of the models. We will apply the CMA-ES optimization algorithm to find the optimal solution.

To validate the models' closed-form solutions, after finding the optimal parameters we will input them into a Monte Carlo pricer (adapted to each model), and calculate the simulated price with each model's calibrated parameters. To better grasp the behavior of the simulations we will repeat them a large number of times, N_{reps}, averaging the results to produce a function - henceforth named *simulated function* - and extracting from them the 90% confidence intervals of the simulations - these confidence intervals are obtained by sorting the simulated implied volatilities for each strike and extracting both the 90% highest and the 10% lowest.

We will fit all our functions to a dataset of implied volatilities for European options with different maturities and strike prices. This data was kindly provided by *BNP Paribas* and is shown in Appendix A.

For the case of Dupire's local volatility, because it is nonparametric, no calibration is required. Instead we must perform an interpolation in order to find the implied volatility surface from the data mentioned before and, with it, find the local volatility function. From this, we can easily obtain the prices of European options from the Monte Carlo pricer described in subsections 4.1.1 and 4.1.2. These prices (and their implied volatilities) can then be compared to their true market values. As for the stochastic volatility models above, we will also run the Monte Carlo pricer a large number of times, N_{reps}, to produce the simulated function and its 90% confidence intervals.

In order for the models to be comparable, we need to make sure that the same global parameters are used in all the adjustments and simulations, namely the initial stock price, S_0, the risk-free interest rate, r, the time step size (used in the Monte Carlo simulations), Δt, the number of pricer repetitions (to be averaged, producing the expected simulated function and the confidence intervals), N_{reps}, and the number of paths simulated, N_{paths}. Their values are shown in Table 5.1.

S_0(€)	r(yr^{-1})	Δt(days)	N_{paths}	N_{reps}
1	0	0.5	100 000	100

Table 5.1: Parameters used throughout all simulations.

The Monte Carlo pricers of each model will then be modified to price Barrier options instead of the European options used in the calibration. These results will be compared with one another, though no validation is possible due to a lack of data.

5.1 Constant Volatility Model

To find how well our models perform, we need some reference against which to compare them. One clear possibility is to assume a constant volatility throughout the options' duration, since this is the simplest possible case.

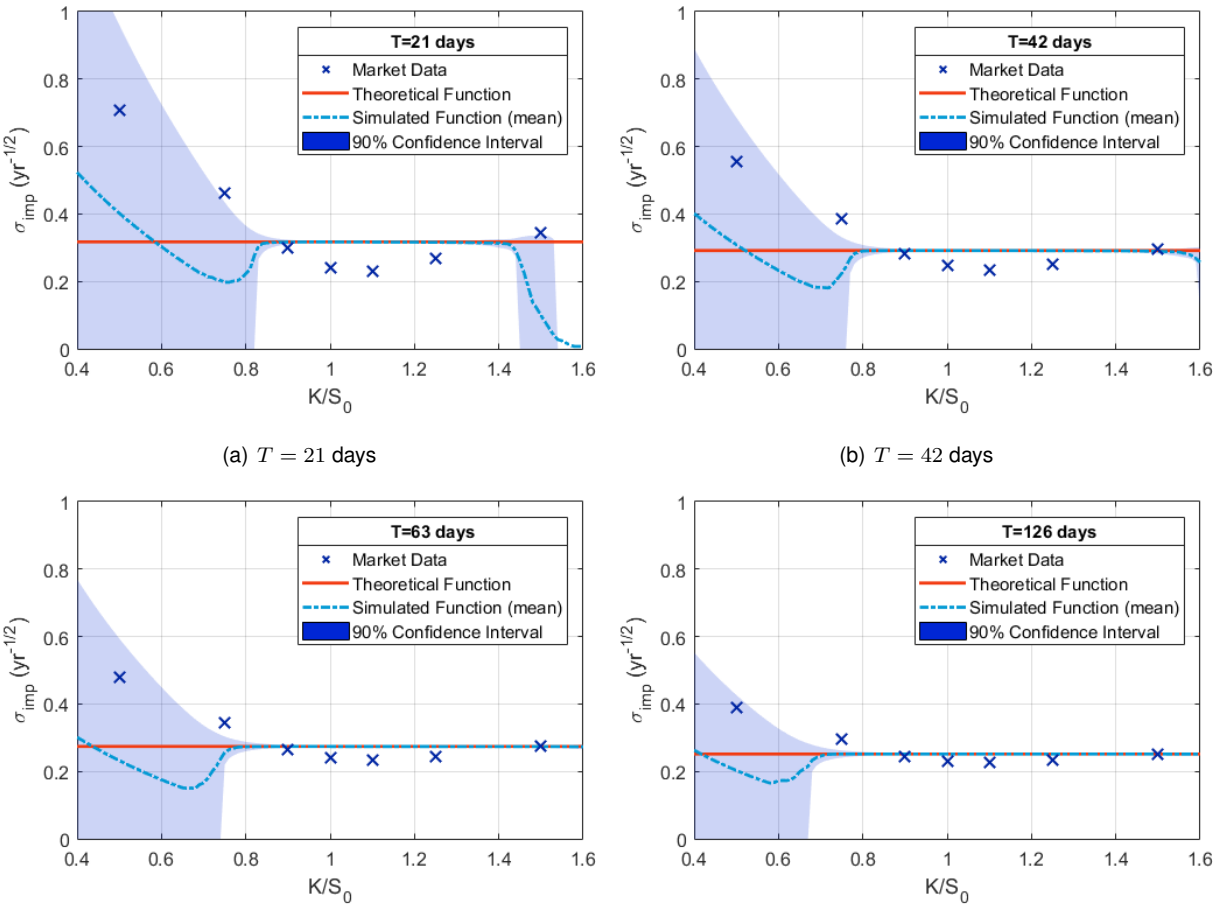
Because we are fitting the volatility function - which, in this case, is a constant value - to a set of volatility data - which is not constant - these adjustments are expected to badly represent the data. Any other minimally decent model is expected to outperform this constant volatility benchmark.

With this constant volatility model, we can choose to fit the datasets with different maturities independently of one another or together in a single coupled dataset. In other words, we can do several fits, one for each maturity, or a single one, for all maturities together. The former will be useful when benchmarking the Static SABR model, since for that model the adjustments will also be performed independently (Static SABR performs badly for multiple maturities). The latter will be more appropriate when studying the remaining models - Dupire, Heston and Dynamic SABR - since these fit the whole implied volatility surface (i.e. multiple maturities together). Thus, both versions will be implemented and briefly studied.

5.1.1 Independent Fits

We begin by presenting the results of fitting a constant volatility function to the sets of data with different maturities *independently*.

In Figure 5.1 we show the plots of each fit along with the provided market data. We also present the simulated functions and the simulations' confidence intervals.



(a) $T = 21$ days

(b) $T = 42$ days

(c) $T = 63$ days

(d) $T = 126$ days

Figure 5.1: Implied volatility functions (red lines) fitted *independently* to the implied volatility data (crosses) for different maturities under constant volatility model, plotted with their respective Monte Carlo simulated functions (light-blue dot-dashed lines) along with their 90% confidence intervals (blue region).

The fitted parameter, which, in this case, is only the constant implied volatility, is presented in Table 5.2 for each of the maturities, along with the cost function's value (eq.(4.12)) under this (fitted) model.

T (days)	$\sigma_{imp,\mathrm{mdl}}$ (yr$^{-1/2}$)	Cost
21	0.3174	0.0635
42	0.2918	0.0282
63	0.2742	0.0164
126	0.2518	0.0069

Table 5.2: Fitted parameters for each maturity (fitted independently) under constant volatility model.

T(days)	K(€)	$\sigma_{i,\mathrm{mkt}}$(yr$^{-0.5}$)	$\sigma_{i,\mathrm{mdl}}$(yr$^{-0.5}$)	$\mathrm{Error}_\sigma(\%)$	C_{mkt}(€)	C_{mdl}(€)	$\mathrm{Error}_C(\%)$
	0.50	0.7082		55.2	0.50001	0.50000	0.003
	0.75	0.4632		31.5	0.25065	0.25002	0.3
	0.90	0.2989		6.2	0.10439	0.10540	1.0
21	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
	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
42	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
	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
63	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
	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
126	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 5.3: Comparison between fitted functions (fitted independently) and original data under constant volatility model.

By observing the fit results in Table 5.2 we see that the cost function decreases with the maturity. This is indeed what is expected, since the implied volatility surface becomes flatter as maturity increases, as can be seen from the market data (and also as we can see in Figure 5.3), making the constant volatility function a better approximation in such cases, and decreasing the cost function's value.

One other property that we can observe is that the constant implied volatility also decreases with maturity. The reason behind this is the simple fact that earlier maturities contain higher implied volatilities, which pull the constant volatility function upwards.

We now focus on the simulated results in Figure 5.1.

First, we can see that the theoretical functions (full red lines) clearly don't represent the market data. This is indeed what is expected. After all, if a constant volatility model was enough to price options, the more complex models would not be needed.

Secondly, by comparing the theoretical and simulated functions, we must note that the simulation performs extremely well for strikes near S_0. Notice that the simulated function (dot-dashed blue line) perfectly follows the constant volatility theoretical result (full red line) in this region, with the confidence intervals collapsed into the simulated function indicating that all simulations produced the same result. This suggests that the simulation is working as expected.

Thirdly, we notice that on the earliest maturity (i.e. 21 days), for strikes much larger than S_0 (e.g. $K = 1.5S_0$), the simulated implied volatilities go to zero, even though they should remain constant. The reason behind this has already been discussed in subsection 4.1.2 and relates to the very, very small number of paths that reach such high strikes and end up contributing to the option price (which is then converted to an implied volatility). For the case of strike $K = 1.5S_0$, and maturity $T = 21$ days, the number of paths that reach the strike is indeed approximately 0.25 - sometimes a single path is able to breach this strike, but usually none do. This problem is not observed for the remaining maturities for the simple reason that, because they are given more time to evolve, more paths are able to reach these high strikes.

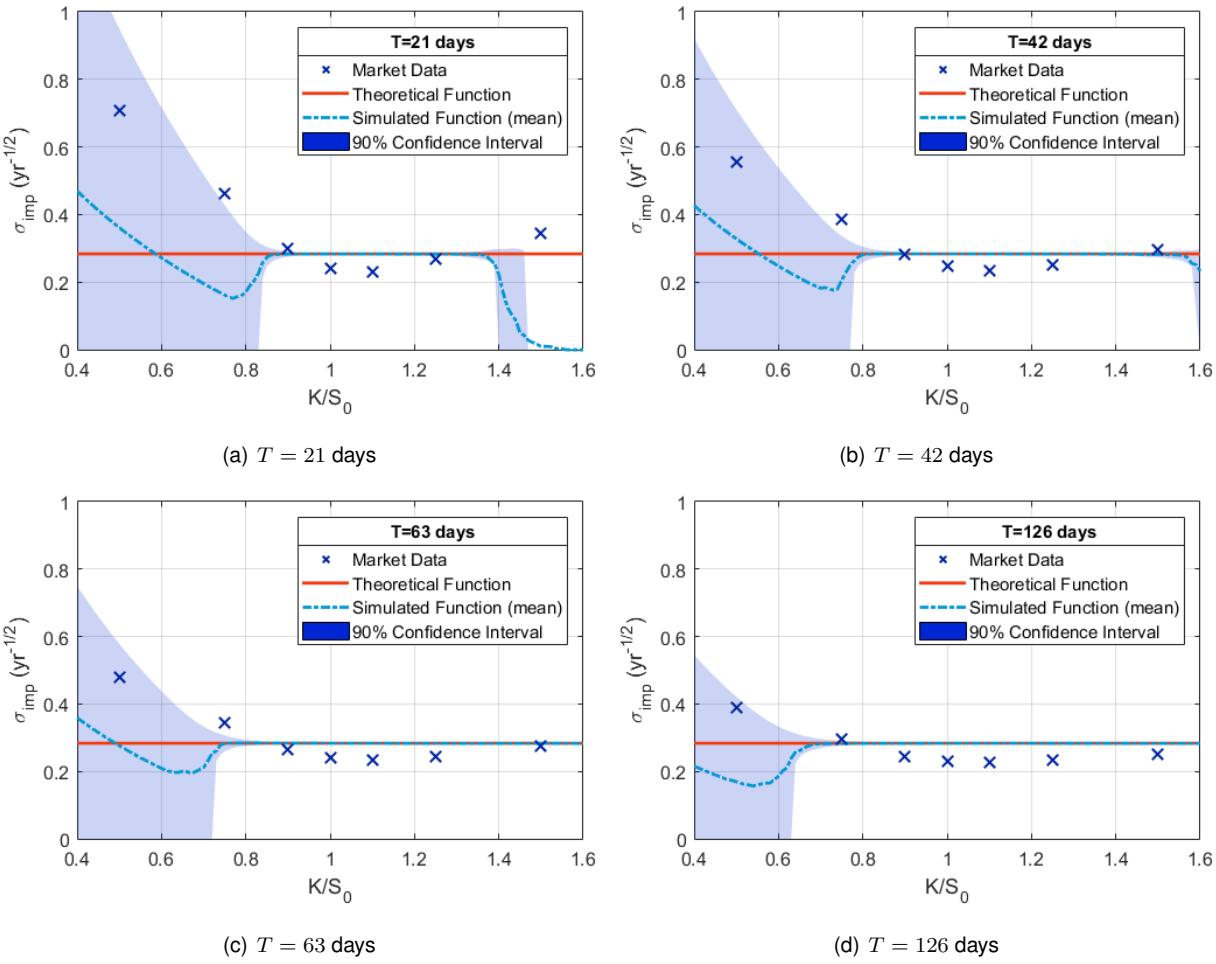
Finally we must discuss the large deviation for strikes lower than S_0, occurring over all maturities. To explain this we require the information presented in Figure 3.3, relating the implied volatility with the option's market price. We saw that the implied volatility was extremely sensitive to option prices very close to their lower bounds, $S_0 - Ke^{-rT}$. If we now look, for example, at the market price of the option with strike $K = 0.5$€ and maturity $T = 21$ days, which is $C = 0.500\,013$€, and we compare it with its lower bound, $S_0 - Ke^{-rT} = 0.5$€, we can clearly see that they are extremely close, with a difference of only 0.0026%. We can then conclude that, in this case, we are in the extremely high sensitivity region of the implied volatility functions shown in Figure 3.3. The Monte Carlo pricer behaves very well for the lower strikes since a very large number of paths contribute to the option price (the opposite of what we described before for high strikes). Despite this, because there will always be some noise associated with the simulations, the generated price is sure to present some very slight variation when executed multiple times, which is enough to cause some of the simulated implied volatilities to go to approximately zero (or to increase), which explains the large confidence intervals.

These two last problems are not inherent to the constant volatility model. Indeed they will be observed, to some extent, in all the models we will study next.

5.1.2 Dependent Fits

We now present the results obtained from the adjustment of a constant volatility function to all the implied volatility data, regardless of maturity.

In Figure 5.2 are shown the theoretical and simulated functions as well as the provided market data.



(a) $T = 21$ days

(b) $T = 42$ days

(c) $T = 63$ days

(d) $T = 126$ days

Figure 5.2: Implied volatility functions (red lines) fitted *simultaneously* to the implied volatility data (crosses) for different maturities under constant volatility model, plotted with their respective Monte Carlo simulated functions (light-blue dot-dashed lines) along with their 90% confidence intervals (blue region).

The fitted implied volatility is represented in Table 5.4 as well as the cost function's value.

$\sigma_{imp,\mathrm{mdl}}$ (yr$^{-1/2}$)	Cost
0.2838	0.1248

Table 5.4: Fitted parameters for all maturities (fitted simultaneously) under constant volatility model.

T(days)	K(€)	$\sigma_{i,\text{mkt}}(\text{yr}^{-0.5})$	$\sigma_{i,\text{mdl}}(\text{yr}^{-0.5})$	$\text{Error}_\sigma(\%)$	C_{mkt}(€)	C_{mdl}(€)	$\text{Error}_C(\%)$
21	0.50	0.7082	0.2838	59.9	0.50001	0.50000	0.003
	0.75	0.4632		38.7	0.25065	0.25000	0.3
	0.90	0.2989		5.0	0.10439	0.10364	0.7
	1.00	0.2425		17.0	0.02792	0.03267	17.0
	1.10	0.2314		22.6	2.42×10^{-3}	5.19×10^{-3}	114.4
	1.25	0.2699		5.1	5.34×10^{-5}	8.98×10^{-5}	67.9
	1.50	0.3433		17.3	5.75×10^{-7}	0.07×10^{-7}	98.8
42	0.50	0.5556	0.2838	48.9	0.50005	0.50000	0.01
	0.75	0.3876		26.8	0.25186	0.25021	0.7
	0.90	0.2824		0.5	0.11069	0.11084	0.1
	1.00	0.2461		15.3	0.04006	0.04620	15.3
	1.10	0.2354		20.6	8.52×10^{-3}	14.02×10^{-3}	64.5
	1.25	0.2525		12.4	6.21×10^{-4}	13.36×10^{-4}	115.3
	1.50	0.2968		4.4	1.58×10^{-5}	0.83×10^{-5}	47.6
63	0.50	0.4789	0.2838	40.7	0.50009	0.50000	0.02
	0.75	0.3452		17.8	0.25296	0.25097	0.8
	0.90	0.2658		6.8	0.11533	0.11786	2.2
	1.00	0.2401		18.2	0.04787	0.05656	18.2
	1.10	0.2330		21.8	0.01421	0.02227	56.7
	1.25	0.2438		16.4	1.80×10^{-3}	3.93×10^{-3}	118.2
	1.50	0.2749		3.2	7.66×10^{-5}	10.87×10^{-5}	42.0
126	0.50	0.3878	0.2838	26.8	0.50035	0.50001	0.1
	0.75	0.2954		3.9	0.25694	0.25589	0.4
	0.90	0.2444		16.1	0.12716	0.13611	7.0
	1.00	0.2295		23.7	0.06467	0.07992	23.6
	1.10	0.2269		25.1	0.02862	0.04317	50.8
	1.25	0.2340		21.3	7.57×10^{-3}	14.99×10^{-3}	98.0
	1.50	0.2521		12.6	8.58×10^{-4}	19.67×10^{-4}	129.3

Table 5.5: Comparison between fitted functions (fitted simultaneously) and original data under constant volatility model.

First, when we compare the implied volatility fitted to all the maturities with the results of the independent fits (in Table 5.2) we see that the former is actually the average of the latter. why?

If we now compare the cost function values, we see that the cost of the dependent fit (0.1248), is actually larger than the sum of cost of the independent fits $(0.0635 + 0.0282 + 0.0164 + 0.0069 = 0.1150)$. This is indeed expected: when fitting all maturities at once, the optimizer will find the implied volatility that minimizes the cost for all maturities, which will not be the one that minimizes the cost of each single maturity, so that the sum of the errors is higher.

As for the plots in Figure 5.2 they look very similar to those in Figure 5.1, so not much more can be added regarding their analysis.

5.2 Dupire Model

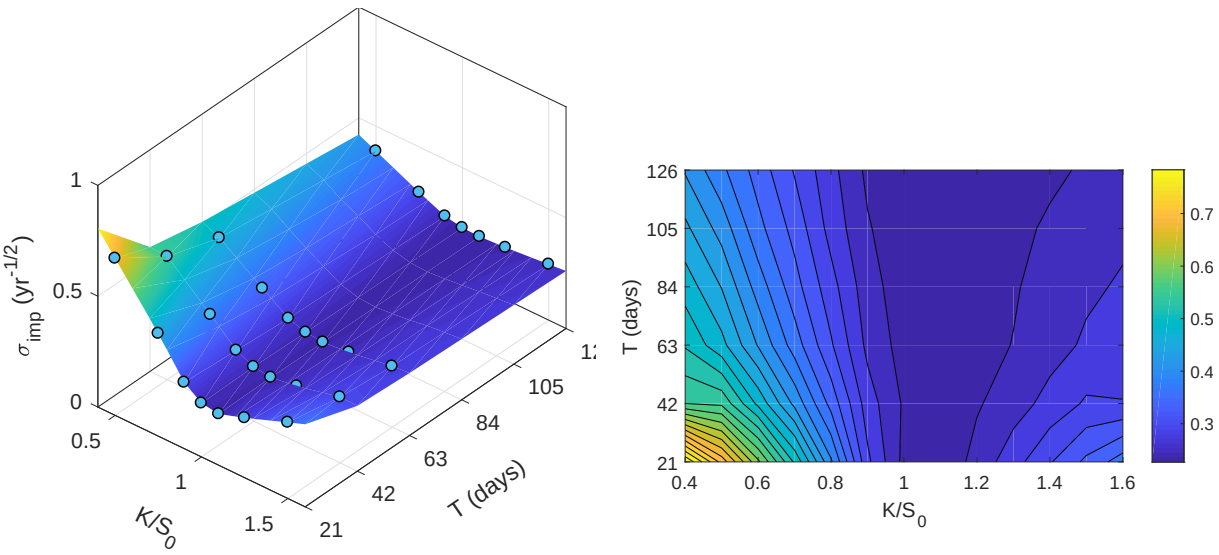
Dupire's model was defined in eq.(3.23) as is given by

$$\sigma(S,t) = \sqrt{\frac{\sigma_{imp}^2 + 2t\sigma_{imp}\frac{\partial\sigma_{imp}}{\partial T} + 2rSt\sigma_{imp}\frac{\partial\sigma_{imp}}{\partial K}}{\left(1 + Sd_1\sqrt{t}\frac{\partial\sigma_{imp}}{\partial K}\right)^2 + S^2 t\sigma_{imp}\left(\frac{\partial^2\sigma_{imp}}{\partial K^2} - d_1\left(\frac{\partial\sigma_{imp}}{\partial K}\right)^2\sqrt{t}\right)}}, \tag{5.1}$$

where d_1 is defined as

$$d_1 = \frac{\log(S_0/S) + \left(r + \frac{1}{2}\sigma_{imp}^2\right)t}{\sigma_{imp}\sqrt{t}}. \tag{5.2}$$

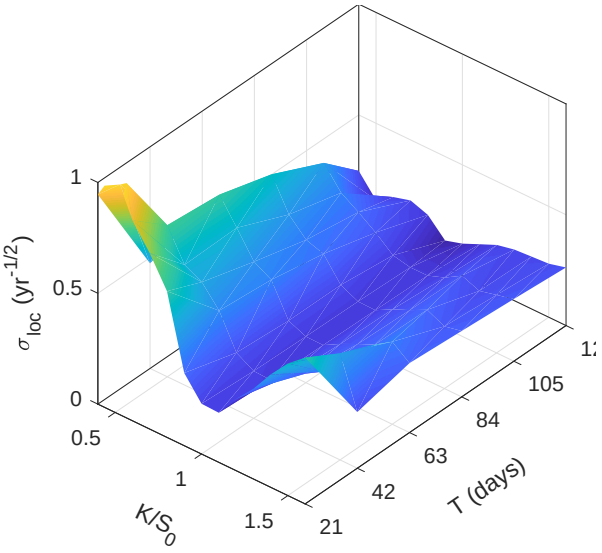
As we can see, we must produce the implied volatility surface from the market data. Because we only have data for a finite set of maturities and strikes, the implied volatility surface must be obtained by some form of interpolation and extrapolation. continue here



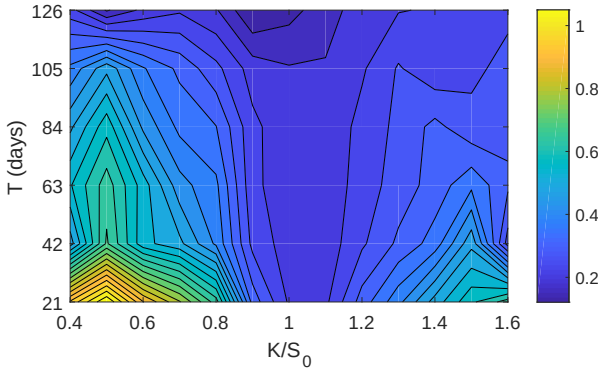
(a) σ_{imp} surface

(b) σ_{imp} contour plot

Figure 5.3: Implied volatility surface (left) and corresponding contour plot (right) of the function interpolated linearly between the original data points (blue circles).

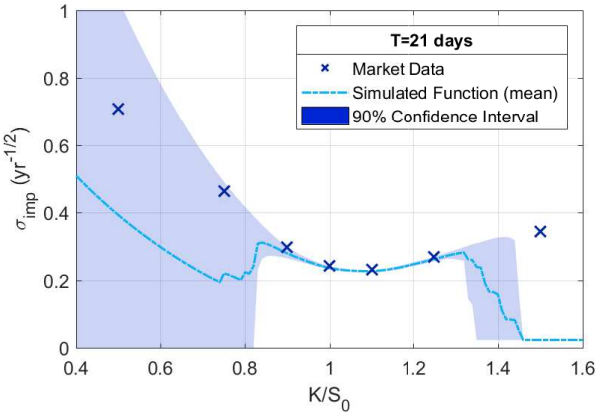


(a) σ_{loc} surface

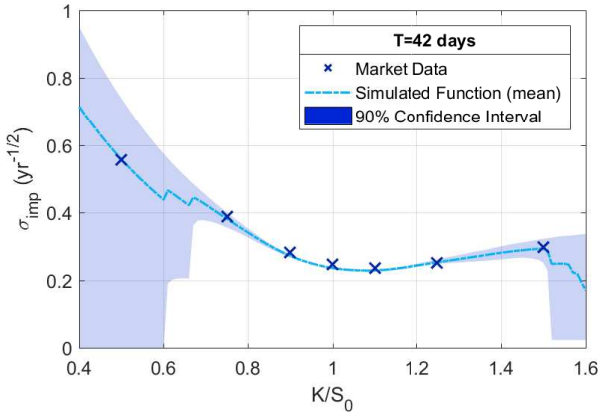


(b) σ_{loc} contour plot

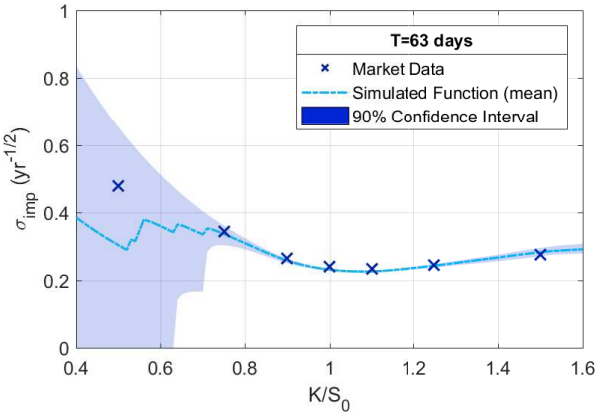
Figure 5.4: Local volatility surface (left) and corresponding contour plot (right) of the function obtained with Dupire's formula (eq.(3.23)) from the interpolated implied volatility surface in Figure 5.3.



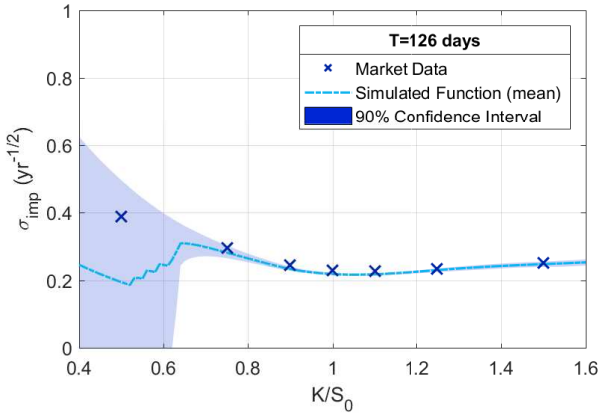
(a) $T = 21$ days



(b) $T = 42$ days

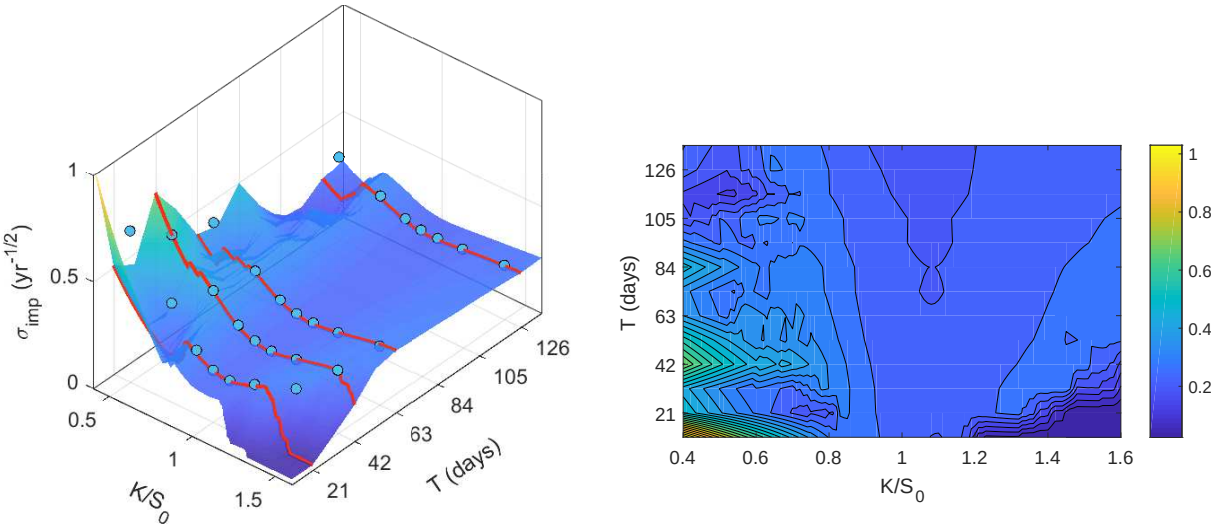


(c) $T = 63$ days



(d) $T = 126$ days

Figure 5.5: Implied volatility functions (light-blue dot-dashed) simulated with Monte Carlo under Dupire's local volatility model with their corresponding 90% confidence interval, plotted against the original market data (crosses).



(a) σ_{imp} surface

(b) σ_{imp} contour plot

Figure 5.6: Implied volatility surface (left) and corresponding contour plot (right) simulated with Monte Carlo under Dupire's local volatility model plotted against the original market data (blue circles) and the generated functions shown in Figure 5.5 (red dot-dashed lines).

Interpolation						Simulation		
T_{min}(days)	T_{max}(days)	ΔT(days)	K_{min}(€)	K_{max}(€)	ΔK(€)	N_{paths}	N_{reps}	σ_{max}(yr$^{-1/2}$)
21	126	10.5	0.4	1.6	0.05	50000	30	2

Table 5.6: Parameters used in the interpolation and simulation sections of Dupire's model.

T(days)	K(€)	$\sigma_{i,\mathrm{mkt}}(\mathrm{yr}^{-0.5})$	$\sigma_{i,\mathrm{mdl}}(\mathrm{yr}^{-0.5})$	$\mathrm{Error}_\sigma(\%)$	C_{mkt}(€)	C_{mdl}(€)	$\mathrm{Error}_C(\%)$
21	0.50	0.7082	0.2319	67.3	0.50001	0.50000	0.003
	0.75	0.4632	0.3740	19.3	0.25065	0.25011	0.2
	0.90	0.2989	0.2845	4.8	0.10439	0.10368	0.7
	1.00	0.2425	0.2374	2.1	0.02792	0.02734	2.1
	1.10	0.2314	0.2276	1.7	2.42×10^{-3}	2.26×10^{-3}	6.8
	1.25	0.2699	0.2633	2.5	5.34×10^{-5}	4.06×10^{-5}	24.0
	1.50	0.3433	0.1421	58.6	5.75×10^{-7}	0	100.0
42	0.50	0.5556	0.3846	30.8	0.50005	0.50000	0.01
	0.75	0.3876	0.3683	5.0	0.25186	0.25139	0.2
	0.90	0.2824	0.2700	4.4	0.11069	0.10946	1.1
	1.00	0.2461	0.2358	4.2	0.04006	0.03838	4.2
	1.10	0.2354	0.2286	2.9	8.52×10^{-3}	7.83×10^{-3}	8.2
	1.25	0.2525	0.2539	0.6	6.21×10^{-4}	6.46×10^{-4}	4.1
	1.50	0.2968	0.3089	4.1	1.58×10^{-5}	2.70×10^{-5}	70.3
63	0.50	0.4789	0.3176	33.7	0.50009	0.50000	0.02
	0.75	0.3452	0.3355	2.8	0.25296	0.25256	0.2
	0.90	0.2658	0.2578	3.0	0.11533	0.11424	0.9
	1.00	0.2401	0.2310	3.8	0.04787	0.04606	3.8
	1.10	0.2330	0.2253	3.3	0.01421	0.01307	8.0
	1.25	0.2438	0.2440	0.1	1.80×10^{-3}	1.81×10^{-3}	0.5
	1.50	0.2749	0.2845	3.5	7.66×10^{-5}	11.15×10^{-5}	45.7
126	0.50	0.3878	0.3870	0.2	0.50035	0.50035	0.001
	0.75	0.2954	0.2876	2.6	0.25694	0.25623	0.3
	0.90	0.2444	0.2358	3.5	0.12716	0.12528	1.5
	1.00	0.2295	0.2203	4.0	0.06467	0.06207	4.0
	1.10	0.2269	0.2190	3.5	0.02862	0.02667	6.8
	1.25	0.2340	0.2319	0.9	7.57×10^{-3}	7.31×10^{-3}	3.4
	1.50	0.2521	0.2528	0.3	8.58×10^{-4}	8.77×10^{-4}	2.2

Table 5.7: Comparison between the data obtained by generating N_{paths} paths under Dupire's local volatility model using the Monte Carlo pricing method and the original data.

5.3 Static SABR Model

As we saw before, the Static SABR model is defined as

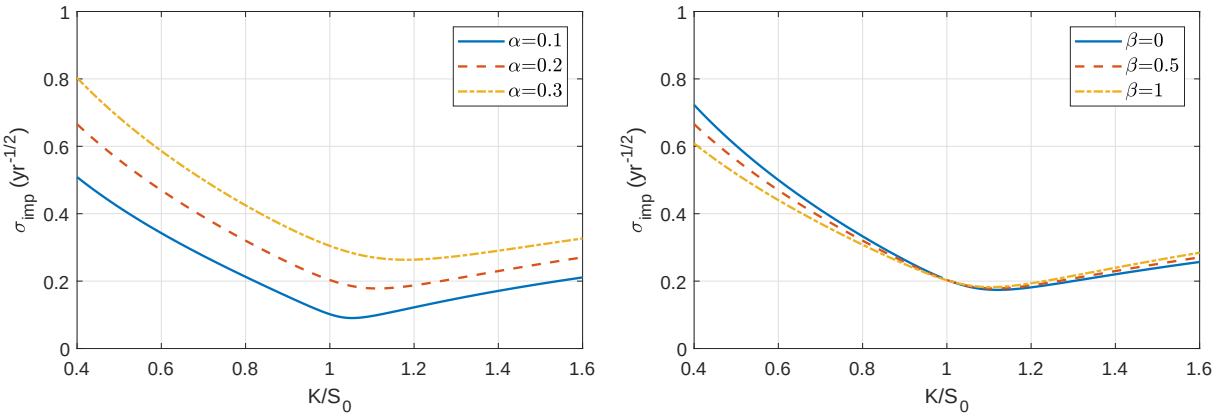
$$dS = rSdt + e^{-r(T-t)(1-\beta)}\sigma S^{\beta} dW_1, \tag{5.3}$$

$$d\sigma = \nu\sigma dW_2, \tag{5.4}$$

with $\alpha = \sigma(0)$ and with W_1 and W_2 having a correlation of ρ.

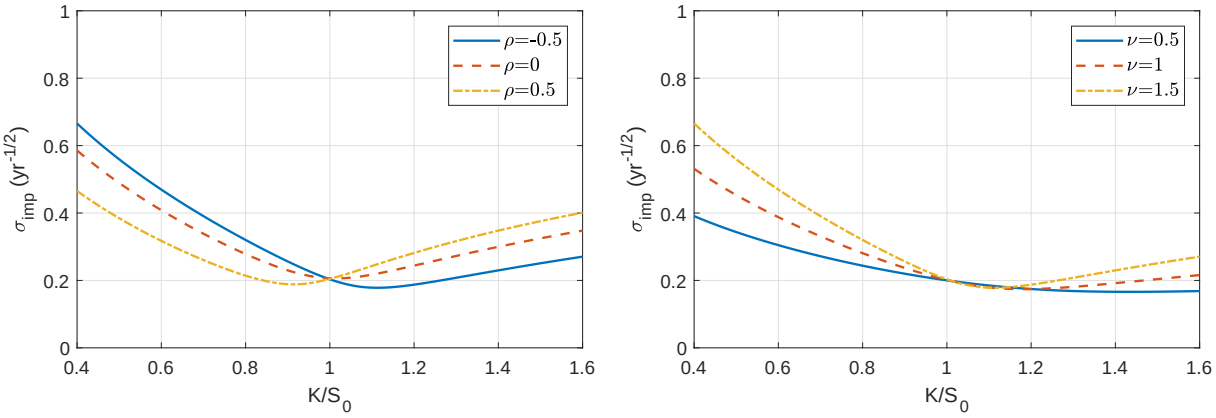
The closed form solution, shown in eq.(3.68), enables us to obtain the theoretical implied volatilities of options priced under this model and will be used in the calibration process.

The influence of each parameter of the Static SABR model on the shape of the implied volatility curve is shown in Figure 5.7.



(a) Dependence on α

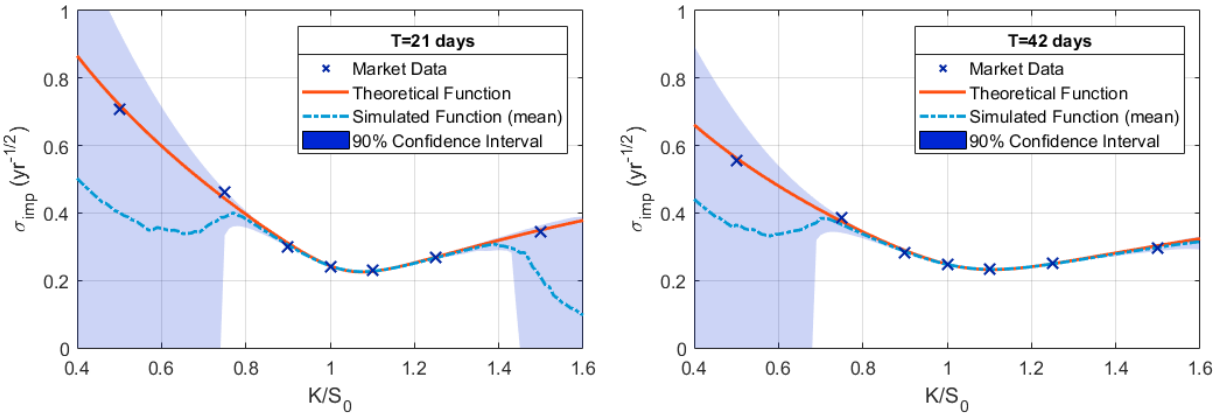
(b) Dependence on β



(c) Dependence on ρ

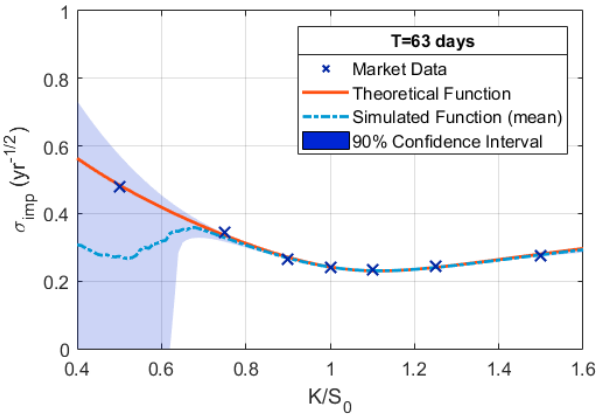
(d) Dependence on ν

Figure 5.7: Dependence of the implied volatility curve on each of the Static model SABR parameters. The default parameters used were $S_0 = 1€$, $T = 42$ days and $r = 0$. Furthermore, on all plots, except when the dependence on a parameter is represented, the parameters used were $\alpha = 0.2$, $\beta = 1$, $\rho = -0.5$ and $\nu = 1.5$.

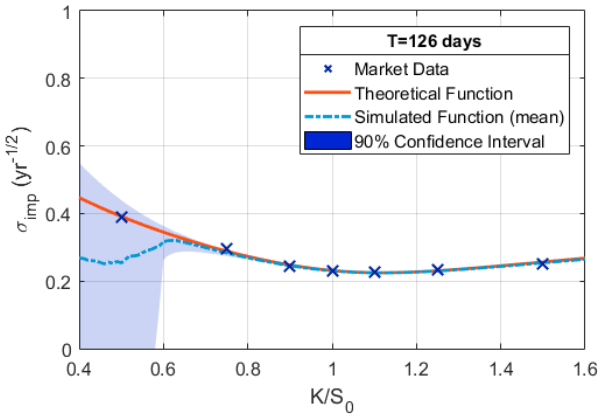


(a) $T = 21$ days

(b) $T = 42$ days



(c) $T = 63$ days



(d) $T = 126$ days

Figure 5.8: Implied volatility functions (red lines) fitted independently to the implied volatility data (crosses) for different maturities under the static SABR model, plotted with their respective Monte Carlo simulated functions (light-blue dot-dashed lines) along with their 90% confidence intervals (blue region).

T(days)	α (yr$^{-1/2}$)	β	ρ	ν	Cost
21	0.2381	0.3766	-0.3760	2.1022	0.000415
42	0.2434	0.7362	-0.3664	1.4451	0.000166
63	0.2375	0.7750	-0.3119	1.1420	0.000102
126	0.2267	0.8771	-0.2383	0.8215	0.000055

Table 5.8: Fitted parameters for each maturity (fitted independently) under static SABR model.

T(days)	K(€)	$\sigma_{i,\mathrm{mkt}}$(yr$^{-0.5}$)	$\sigma_{i,\mathrm{mdl}}$(yr$^{-0.5}$)	$\mathrm{Error}_\sigma(\%)$	C_{mkt}(€)	C_{mdl}(€)	$\mathrm{Error}_C(\%)$
	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
21	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
	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
42	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
	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
63	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
	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
126	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 5.9: Comparison between fitted results and original data under static SABR model.

5.4 Heston Model

The Heston model is defined as

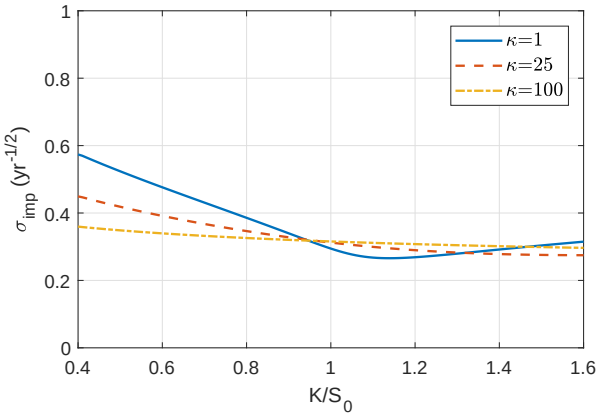
$$dS = rSdt + \sqrt{\nu}SdW_1, \tag{5.5}$$

$$d\nu = \kappa(\overline{\nu} - \nu)dt + \eta\sqrt{\nu}dW_2, \tag{5.6}$$

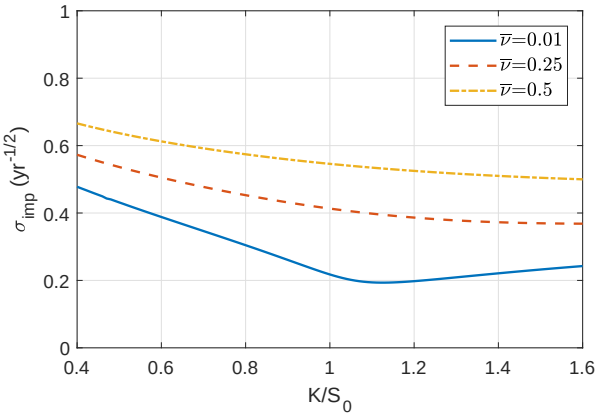
where $\nu_0 = \nu(0)$ and with W_1 and W_2 having a correlation of ρ.

The closed form solution, shown in eq.(3.30), we are able to find the theoretical prices of options priced under this model, which we can easily convert to implied volatilities. These last will be used in the calibration process.

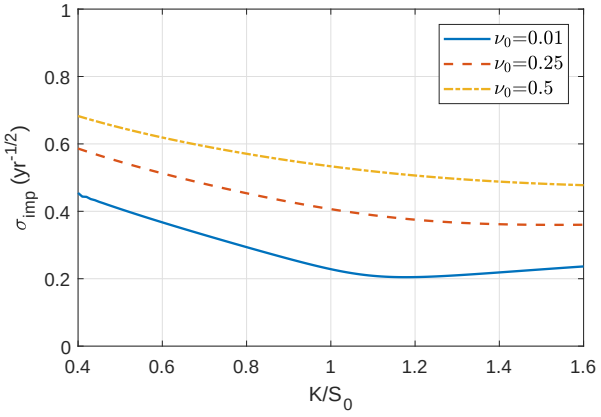
The influence of each parameter of the Heston model on the shape of the implied volatility curve is shown in Figure 5.9.



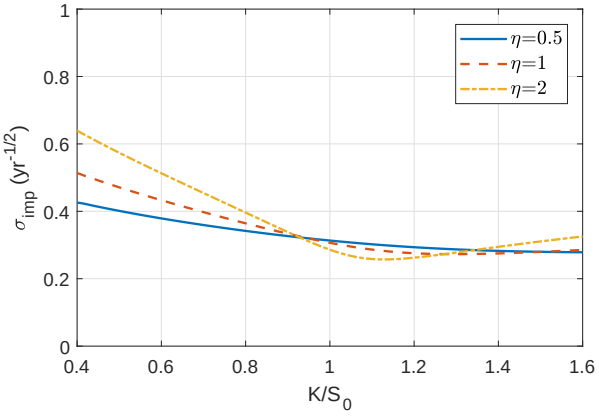
(a) Dependence on κ



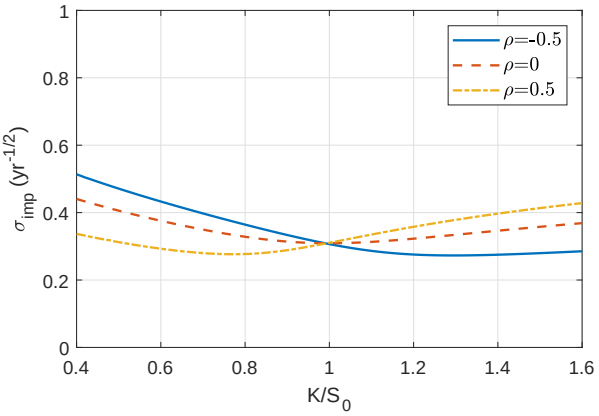
(b) Dependence on $\overline{\nu}$



(c) Dependence on ν_0

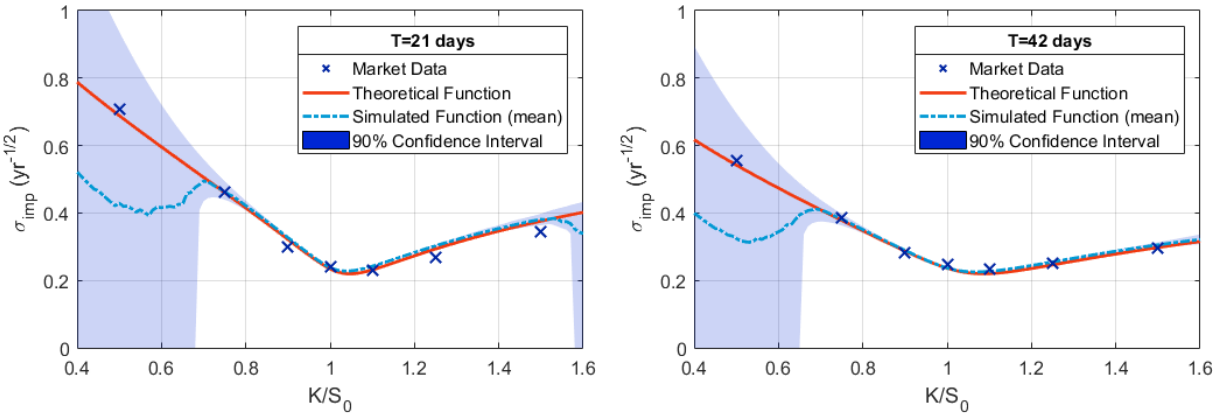


(d) Dependence on η



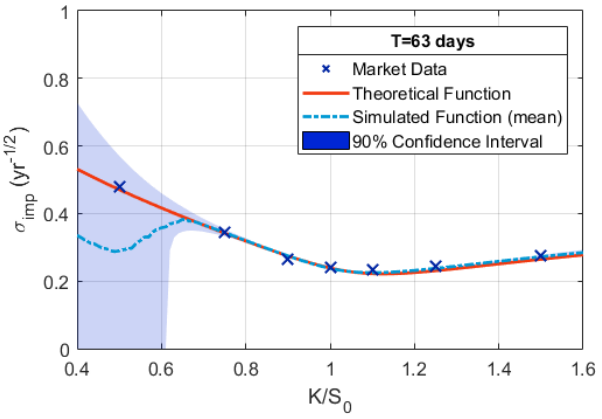
(e) Dependence on ρ

Figure 5.9: Dependence of the implied volatility curve on each of the Heston model parameters. The default parameters used were $S_0 = 1$€, $T = 42$ days and $r = 0$. Furthermore, on all plots, except when the dependence on a parameter is represented, the parameters used were $\kappa = 10$, $\overline{\nu} = 0.1$, $\nu_0 = 0.1$, $\eta = 1$ and $\rho = -0.5$.

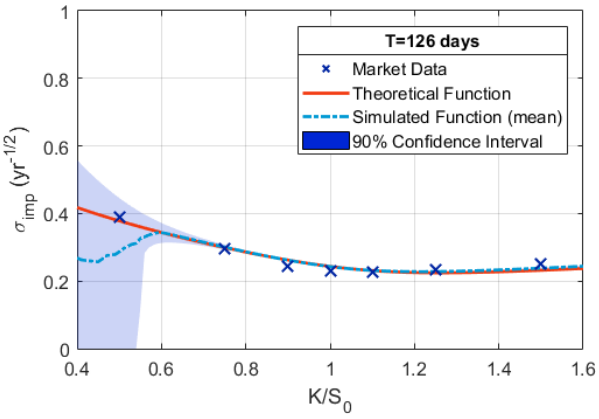


(a) $T = 21$ days

(b) $T = 42$ days

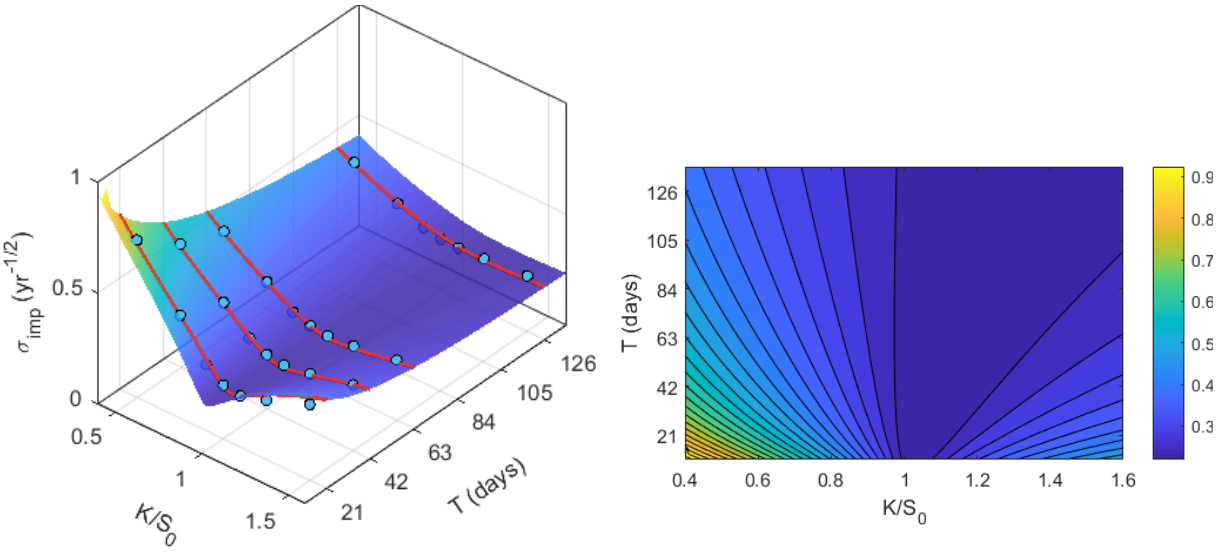


(c) $T = 63$ days



(d) $T = 126$ days

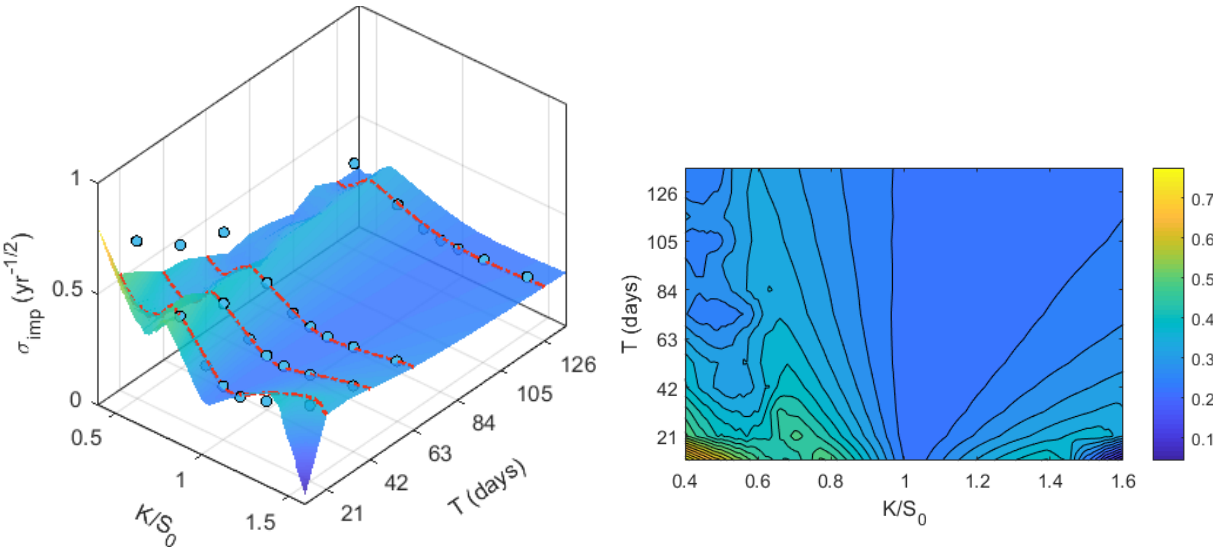
Figure 5.10: Implied volatility functions (red lines) fitted simultaneously to the implied volatility data (crosses) for different maturities under the Heston model, plotted with their respective Monte Carlo simulated functions (light-blue dot-dashed lines) along with their 90% confidence intervals (blue region).



(a) σ_{imp} surface

(b) σ_{imp} contour plot

Figure 5.11: Implied volatility surface (left) and corresponding contour plot (right) of the function fitted simmultaneously to the implied volatility data for different maturities under the Heston model, plotted against the original market data (blue circles) and the fitted functions shown in Figure 5.10 (red lines).



(a) σ_{imp} surface

(b) σ_{imp} contour plot

Figure 5.12: Implied volatility surface (left) and corresponding contour plot (right) of the function simulated using the Monte Carlo procedure with the fitted parameters shown in Table 5.10, under the dynamic SABR model, plotted against the original market data (blue circles) and the simulated functions shown in Figure 5.10 (red dot-dashed lines).

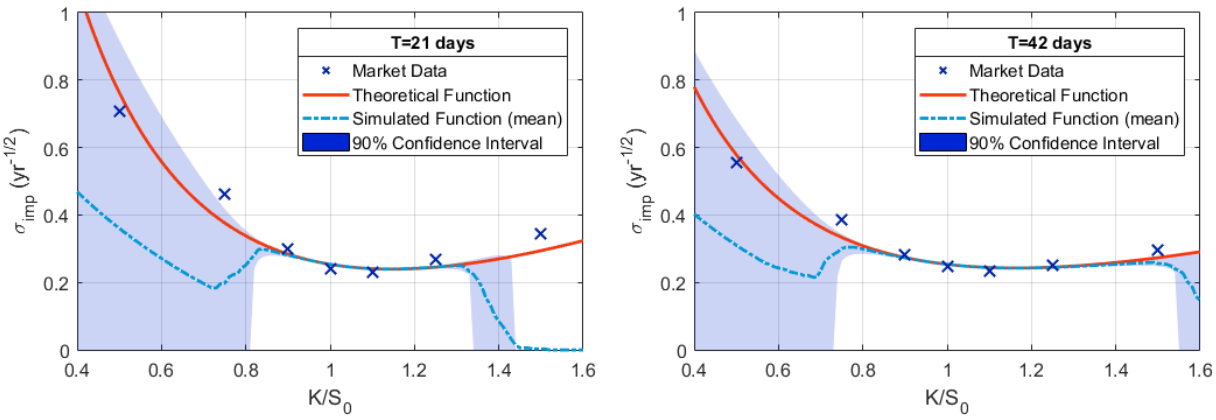
κ	$\overline{\nu}$	ν_0	ρ	η	Cost
53.4355	0.0653	0.1046	-0.4086	6.2554	0.002529

Table 5.10: Fitted parameters for all maturities (fitted simultaneously) under the Heston model.

T(days)	K(€)	$\sigma_{i,\mathrm{mkt}}(\mathrm{yr}^{-0.5})$	$\sigma_{i,\mathrm{mdl}}(\mathrm{yr}^{-0.5})$	$\mathrm{Error}_\sigma(\%)$	C_{mkt}(€)	C_{mdl}(€)	$\mathrm{Error}_C(\%)$
21	0.50	0.7082	0.6886	2.8	0.50001	0.50001	0.001
	0.75	0.4632	0.4604	0.6	0.25065	0.25062	0.01
	0.90	0.2989	0.3216	7.6	0.10439	0.10563	1.2
	1.00	0.2425	0.2346	3.2	0.02792	0.02702	3.2
	1.10	0.2314	0.2316	0.1	2.42×10^{-3}	2.43×10^{-3}	0.3
	1.25	0.2699	0.2935	8.7	5.34×10^{-5}	12.43×10^{-5}	132.5
	1.50	0.3433	0.3759	9.5	5.75×10^{-7}	29.58×10^{-7}	414.7
42	0.50	0.5556	0.5422	2.4	0.50005	0.50004	0.003
	0.75	0.3876	0.3781	2.5	0.25186	0.25162	0.1
	0.90	0.2824	0.2873	1.7	0.11069	0.11120	0.5
	1.00	0.2461	0.2366	3.9	0.04006	0.03852	3.9
	1.10	0.2354	0.2205	6.3	8.52×10^{-3}	7.02×10^{-3}	17.6
	1.25	0.2525	0.2463	2.4	6.21×10^{-4}	5.20×10^{-4}	16.2
	1.50	0.2968	0.2979	0.4	1.58×10^{-5}	1.66×10^{-5}	4.9
63	0.50	0.4789	0.4709	1.7	0.50009	0.50008	0.003
	0.75	0.3452	0.3420	0.9	0.25296	0.25282	0.1
	0.90	0.2658	0.2748	3.4	0.11533	0.11659	1.1
	1.00	0.2401	0.2392	0.4	0.04787	0.04768	0.4
	1.10	0.2330	0.2224	4.6	0.01421	0.01265	11.0
	1.25	0.2438	0.2310	5.2	1.80×10^{-3}	1.31×10^{-3}	27.1
	1.50	0.2749	0.2649	3.7	7.66×10^{-5}	4.97×10^{-5}	35.1
126	0.50	0.3878	0.3784	2.4	0.50035	0.50028	0.01
	0.75	0.2954	0.2993	1.3	0.25694	0.25732	0.1
	0.90	0.2444	0.2626	7.5	0.12716	0.13124	3.2
	1.00	0.2295	0.2439	6.3	0.06467	0.06870	6.2
	1.10	0.2269	0.2314	2.0	0.02862	0.02973	3.9
	1.25	0.2340	0.2248	3.9	7.57×10^{-3}	6.45×10^{-3}	14.8
	1.50	0.2521	0.2324	7.8	8.58×10^{-4}	4.45×10^{-4}	48.2

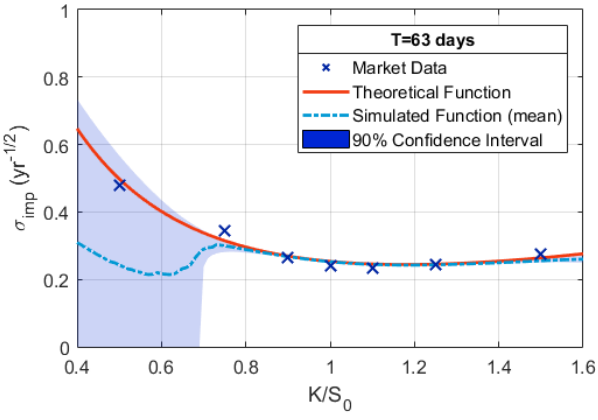
Table 5.11: Comparison between fitted results and original data under the Heston model.

5.5 Dynamic SABR Model

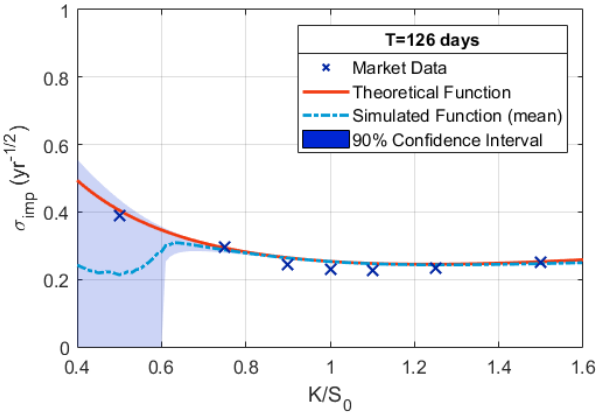


(a) $T = 21$ days

(b) $T = 42$ days

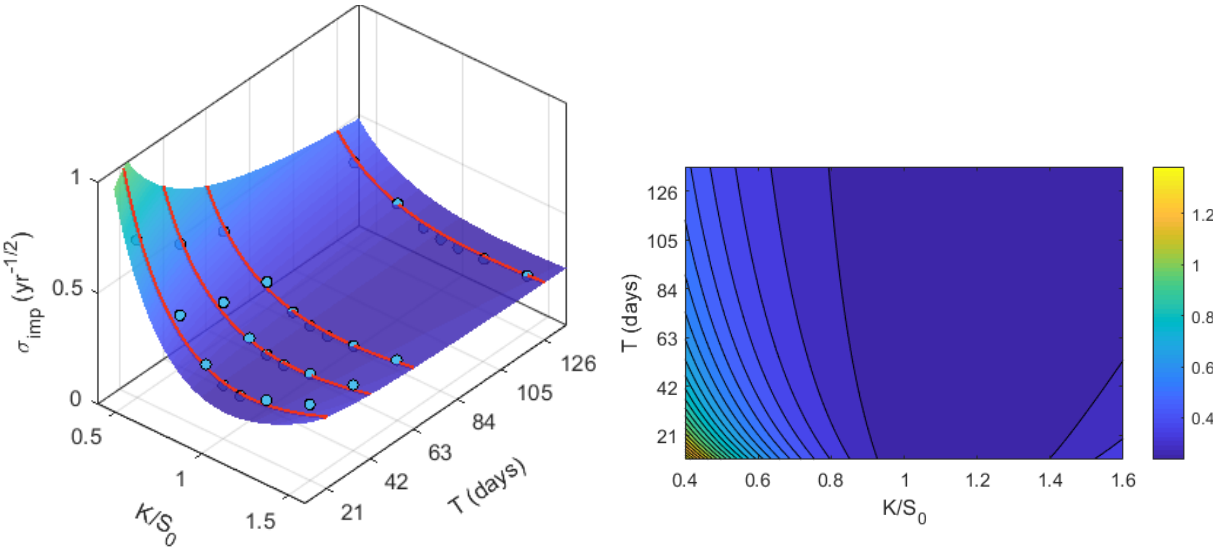


(c) $T = 63$ days



(d) $T = 126$ days

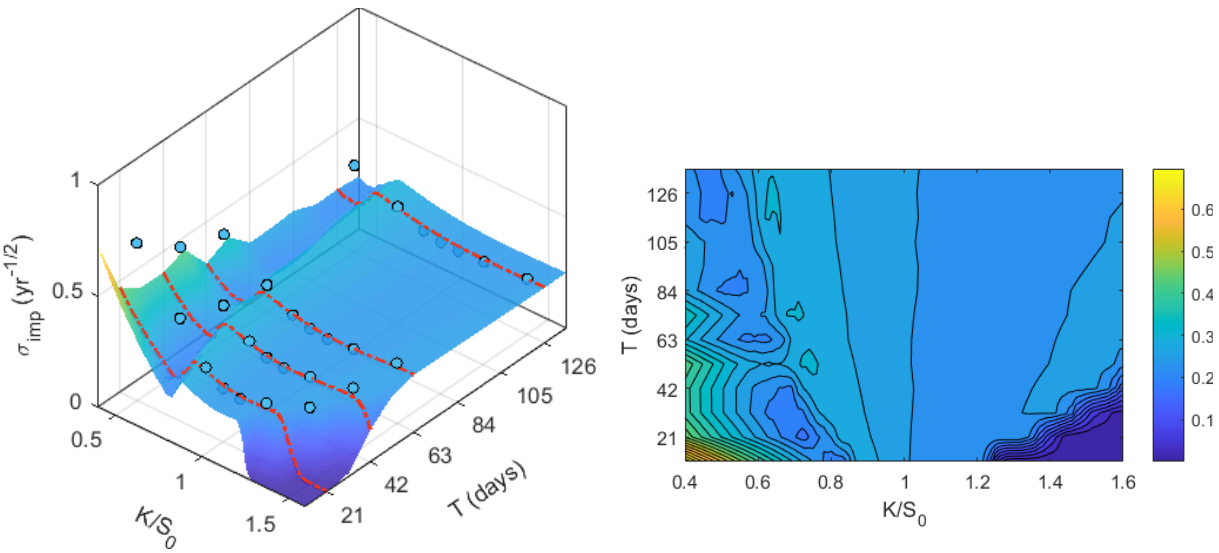
Figure 5.13: Implied volatility functions (red lines) fitted simultaneously to the implied volatility data (crosses) for different maturities under the dynamic SABR model, plotted with their respective Monte Carlo simulated functions (light-blue dot-dashed lines) along with their 90% confidence intervals (blue region).



(a) σ_{imp} surface

(b) σ_{imp} contour plot

Figure 5.14: Implied volatility surface (left) and corresponding contour plot (right) of the function fitted simmultaneously to the implied volatility data for different maturities under the dynamic SABR model, plotted against the original market data (blue circles) and the fitted functions shown in Figure 5.13 (red lines).



(a) σ_{imp} surface

(b) σ_{imp} contour plot

Figure 5.15: Implied volatility surface (left) and corresponding contour plot (right) of the function simulated using the Monte Carlo procedure with the fitted parameters shown in Table 5.12, under the dynamic SABR model, plotted against the original market data (blue circles) and the simulated functions shown in Figure 5.13 (red dot-dashed lines).

α	β	ρ_0	a	ν_0	b	Cost
0.2540	0.6348	-0.4166	0	1.8673	41.6943	0.0108

Table 5.12: Fitted parameters for all maturities (fitted simultaneously) under the dynamic SABR model.

T(days)	K(€)	$\sigma_{i,\mathrm{mkt}}(\mathrm{yr}^{-0.5})$	$\sigma_{i,\mathrm{mdl}}(\mathrm{yr}^{-0.5})$	$\mathrm{Error}_\sigma(\%)$	C_{mkt}(€)	C_{mdl}(€)	$\mathrm{Error}_C(\%)$
	0.50	0.7082	0.7625	7.7	0.50001	0.50003	0.004
	0.75	0.4632	0.3765	18.7	0.25065	0.25012	0.2
	0.90	0.2989	0.2843	4.9	0.10439	0.10367	0.7
21	1.00	0.2425	0.2540	4.7	0.02792	0.02924	4.7
	1.10	0.2314	0.2410	4.2	2.42×10^{-3}	2.86×10^{-3}	18.0
	1.25	0.2699	0.2454	9.1	5.34×10^{-5}	1.76×10^{-5}	67.1
	1.50	0.3433	0.2944	14.2	57.48×10^{-8}	1.86×10^{-8}	96.8
	0.50	0.5556	0.5788	4.2	0.50005	0.50008	0.01
	0.75	0.3876	0.3337	13.9	0.25186	0.25074	0.4
	0.90	0.2824	0.2740	3.0	0.11069	0.10985	0.8
42	1.00	0.2461	0.2538	3.1	0.04006	0.04131	3.1
	1.10	0.2354	0.2445	3.9	8.52×10^{-3}	9.49×10^{-3}	11.3
	1.25	0.2525	0.2455	2.8	6.21×10^{-4}	5.07×10^{-4}	18.3
	1.50	0.2968	0.2736	7.8	15.83×10^{-6}	4.72×10^{-6}	70.2
	0.50	0.4789	0.4980	4.0	0.50009	0.50014	0.01
	0.75	0.3452	0.3150	8.8	0.25296	0.25182	0.5
	0.90	0.2658	0.2695	1.4	0.11533	0.11585	0.4
63	1.00	0.2401	0.2537	5.7	0.04787	0.05057	5.6
	1.10	0.2330	0.2460	5.6	0.01421	0.01619	13.9
	1.25	0.2438	0.2455	0.7	1.80×10^{-3}	1.87×10^{-3}	3.8
	1.50	0.2749	0.2642	3.9	7.66×10^{-5}	4.81×10^{-5}	37.2
	0.50	0.3878	0.4050	4.4	0.50035	0.50051	0.03
	0.75	0.2954	0.2935	0.6	0.25694	0.25677	0.1
	0.90	0.2444	0.2645	8.2	0.12716	0.13168	3.6
126	1.00	0.2295	0.2537	10.5	0.06467	0.07147	10.5
	1.10	0.2269	0.2477	9.1	0.02862	0.03382	18.2
	1.25	0.2340	0.2452	4.8	7.57×10^{-3}	9.05×10^{-3}	19.6
	1.50	0.2521	0.2528	0.3	8.58×10^{-4}	8.77×10^{-4}	2.3

Table 5.13: Comparison between fitted results and original data under the dynamic SABR model.

check notation:

$\sigma_{i,\text{mkt}}$ vs $\sigma_{imp,\text{mkt}}$;

put units in fitted parameters

is implied vol measured in yr-1/2

the heston and dynamic sabr single plot figures are slices of the implied volatility surface

talk about the problem of volatility with sabr (switch volatility and price in code)

plot the decaying functions of rho and nu on the dynamic sabr model

mention the parameters used to simulate each monte carlo pricer

mention sabr negative vol is not problematic

in the background plots (and implementation), use dashed and dot-dashed lines

in the results section, the dashed lines look not-dashed

mention the oscillating errors in heston due to integration

mention noise in Monte Carlo far in/far out of the money - because very few stocks reach the strike, we are averaging over a very limited number of paths, which is very noisy

save files as .png

change labels from ”simulated function” to something else

sabr negative prices: S can become negative, so we must input the restriction S=max(S,0)

update coupled const vol table

mention how we convert prices in the Monte Carlo pricer into implied vols

mention the sensitivity of implied vol to the prices

monte carlo is noisy

explain the effect of barrier level on the price of barrier options

mention why noise in mc pricer will be problematic when option price is near the lower bound and why it only happens for K lower than S0 (because for K above S0, the bound is 0)

mention the small percentage of points that reach the very high strike price of 1.5 (see MATLAB folder)

!!!mention the implied vol sensitivity to option price but changing maturity.

barriers might be broken in between time steps in the simulations. because we are discretizing the time and not using continuous time, this introduces an error

check with Claude if BNPP data assumes a 0 interest rate or not

why is the coupled constant implied volatility the average of the independent fits of the constant implied vols?

explain why stock prices follow a GBM

Chapter 6

Conclusions

Implement importance sampling - this has been done for Heston (and can easily be adapted to SABR) in [45]

Implement antithetic paths

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

use mean-reverting sabr

we could study the greeks

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] K. Chourdakis. Financial engineering: A brief introduction using the matlab system, 2008.

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

[14] S&P Dow Jones Indices. Dow Jones Industrial Average Index (DJIA). https://us.spindices.com/indices/equity/dow-jones-industrial-average. (Accessed: 2018-06-13).

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

[16] P. Wilmott, S. Howison, and J. Dewynne. *Option Pricing: Mathematical Models and Computation*. Oxford Financial Press, 1993.

[17] J. Dewynne, S. Ehrlichman, and P. Wilmott. A simple volatility surface parametrization, 1998.

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

[19] J. Hull and W. Alan. The pricing of options on assets with stochastic volatilities. *The Journal of Finance*, 42(2):281–300.

[20] E. M. Stein and J. C. Stein. Stock price distributions with stochastic volatility: An analytic approach. *Review of Financial Studies*, 4:727–752, 1991.

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

[22] F. Rouah. *The Heston Model and its Extensions in Matlab and C#*. Wiley Finance. Wiley, 2013.

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

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

[25] J. Gatheral. *The Volatility Surface: A Practitioner's Guide*. Wiley Finance. Wiley, 2006.

[26] D. Duffie, J. Pan, and K. Singleton. Transform analysis and asset pricing for affine jump-diffusions. *Econometrica*, 68(6):1343–1376, 2000.

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

[28] 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.

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

[30] H. Albrecher, P. Mayer, W. Schoutens, and J. Tistaert. The little heston trap, 2006.

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

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

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

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

[35] 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.

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

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

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

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

[40] I. Amidror. Scattered data interpolation methods for electronic imaging systems: A survey. 11: 157–176, 04 2002.

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

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

[43] 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.

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

[45] P. Stilger. *Numerical and Empirical Studies of Option Pricing*. PhD thesis, Alliance Manchester Business School, 2015.

Appendix A

Option Market Data

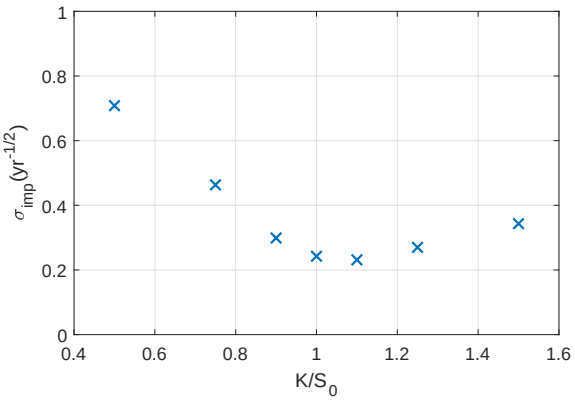
Here, we present the data kindly provided by *BNP Paribas* for European options. For confidentiality reasons, the strike prices were normalized by the initial stock price i.e. $K \to K/S_0$ so that the original strike prices are inaccessible. Suffice it to say that the underlying asset of the options here represented is a *stock index*, i.e. 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.(3.4). The converted prices of call European options are shown below.

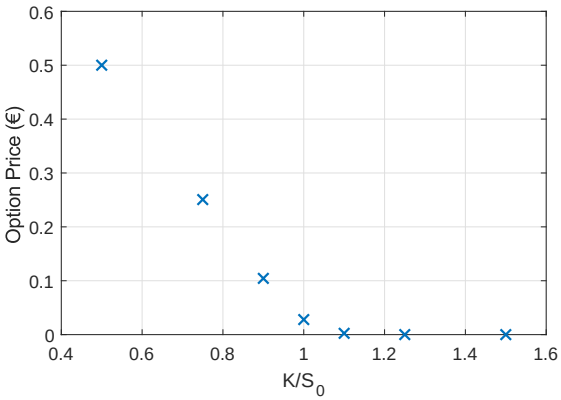
The number of days here denoted correspond to trading days (i.e. days where exchanges are open and trading occurs) so that one month corresponds to 21 days and one year to 252.

T (days)	K (€)	σ_{imp} (yr$^{-1/2}$)	C (€)	T (days)	K (€)	σ_{imp} (yr$^{-1/2}$)	C (€)
	0.50	0.7082	0.500013		0.50	0.4789	0.500093
	0.75	0.4632	0.250648		0.75	0.3452	0.252957
	0.90	0.2989	0.104393		0.90	0.2658	0.115328
21	1.00	0.2425	0.027923	63	1.00	0.2401	0.047872
	1.10	0.2314	0.002421		1.10	0.2330	0.014215
	1.25	0.2699	0.000053		1.25	0.2438	0.001799
	1.50	0.3433	0.000001		1.50	0.2749	0.000077
	0.50	0.5556	0.500050		0.50	0.3878	0.500353
	0.75	0.3876	0.251865		0.75	0.2954	0.256942
	0.90	0.2824	0.110693		0.90	0.2444	0.127156
42	1.00	0.2461	0.040063	126	1.00	0.2295	0.064670
	1.10	0.2354	0.008525		1.10	0.2269	0.028619
	1.25	0.2525	0.000621		1.25	0.2340	0.007569
	1.50	0.2968	0.000016		1.50	0.2521	0.000858

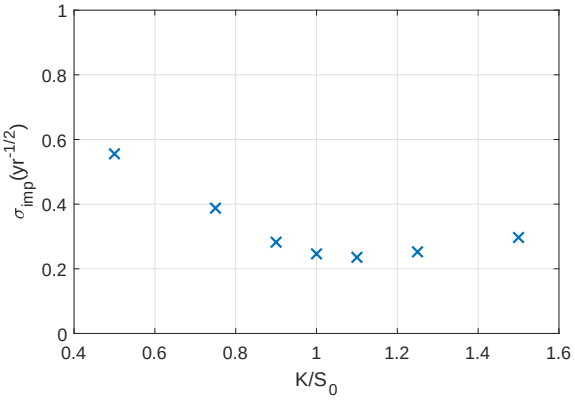
Table A.1: Data provided by *BNP Paribas* to be used in model calibration and validation.



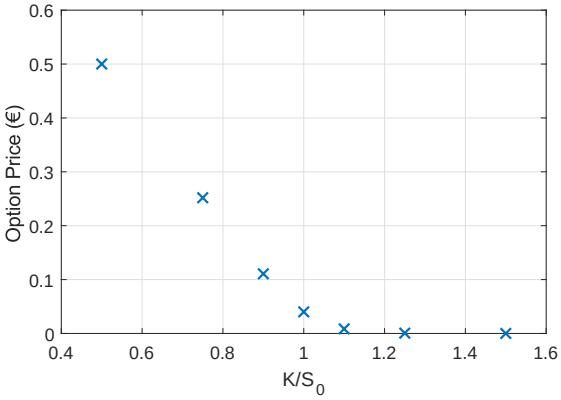
(a) Implied Volatility, T=21 days



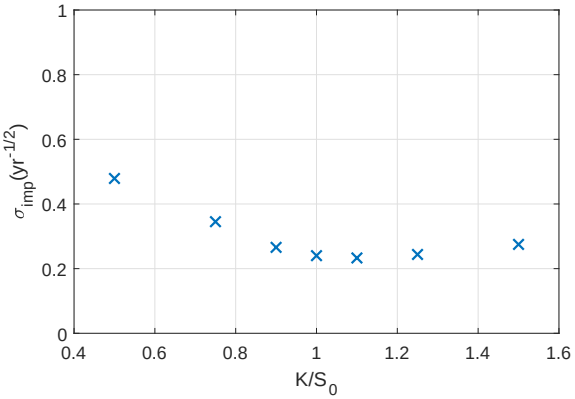
(b) European Call Price, T=21 days



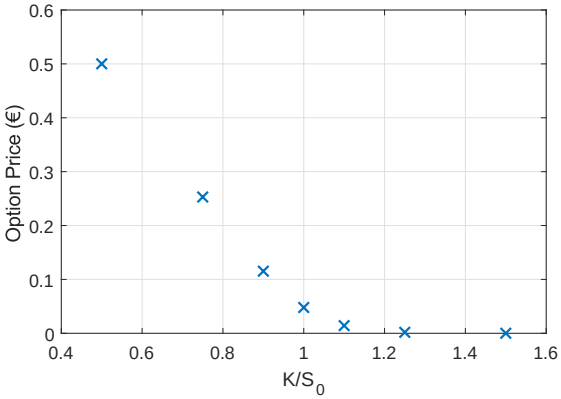
(c) Implied Volatility, T=42 days



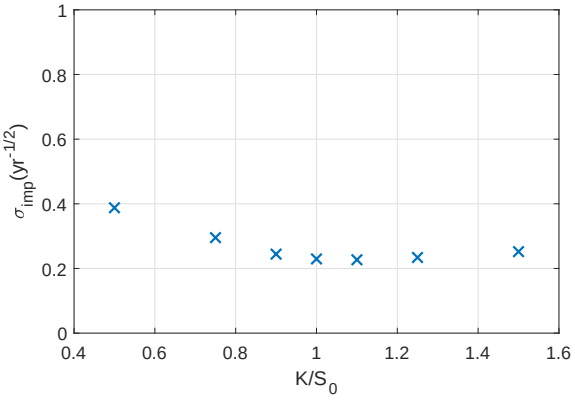
(d) European Call Price, T=42 days



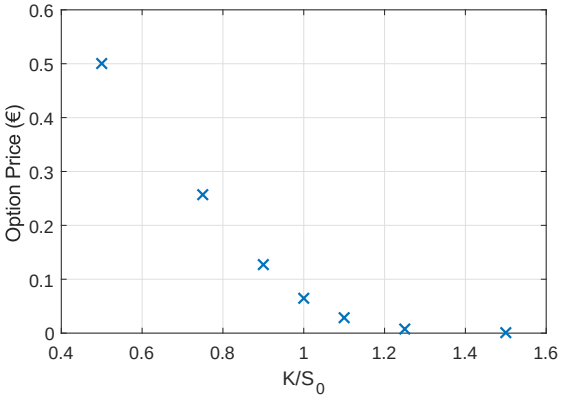
(e) Implied Volatility, T=63 days



(f) European Call Price, T=63 days



(g) Implied Volatility, T=126 days



(h) European Call Price, T=126 days

Figure A.1: Scatter plots of the implied volatilities and European call prices provided

Appendix B

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

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

B.1 The Optimization Algorithm

B.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, $\mathbf{C}^{(0)} = \mathbf{I}$.

B.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

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

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

B.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, $\mathrm{Cost}(\mathbf{x}_{1:\lambda}^{(1)}) \leq \mathrm{Cost}(\mathbf{x}_{2:\lambda}^{(1)}) \leq \ldots \leq \mathrm{Cost}(\mathbf{x}_{\lambda:\lambda}^{(1)})$.

B.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

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

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

B.1.5 Adaptation

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

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

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

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

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

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

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

where we define

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

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

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

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

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

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

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

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

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

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

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.