DEGREE PROJECT IN MATHEMATICS,
SECOND CYCLE, 30 CREDITS
STOCKHOLM, SWEDEN 2019

The Calibrated SSVI Method - Implied Volatility Surface Construction

ADAM ÖHMAN

KTH ROYAL INSTITUTE OF TECHNOLOGY
SCHOOL OF ENGINEERING SCIENCES

The Calibrated SSVI Method - Implied Volatility Surface Construction

ADAM ÖHMAN

Degree Projects in Financial Mathematics (30 ECTS credits)
Degree Programme in Applied and Computational Mathematics (120 credits)
KTH Royal Institute of Technology year 2019
Supervisor at Cinnober Financial Technology AB: Dennis Sundström
Supervisor at KTH: Boualem Djehiche
Examiner at KTH: Boualem Djehiche

TRITA-SCI-GRU 2019:316
MAT-E 2019:73

Royal Institute of Technology
School of Engineering Sciences
KTH SCI
SE-100 44 Stockholm, Sweden
URL: www.kth.se/sci

Abstract

In this thesis will the question of how to construct implied volatility surfaces in a robust and arbitrage free way be investigated.

To be able to know if the solutions are arbitrage free was an initial investigation about arbitrage in volatility surfaces made. From this investigation where two comprehensive theorems found. These theorems came from Roper in [14]. Based on these where then two applicable arbitrage tests created. These tests came to be very important tools in the remaining thesis.

The most reasonable classes of models for modeling the implied volatility surface where then investigated. It was concluded that the classes that seemed to have the best potential where the stochastic volatility models and the parametric representation models. The choice between these two classes where concluded to be based on a trade-off between simplicity and quality of the result. If it where possible to make the parametric representation models improve its result the best applicable choice would be that class. For the remaining thesis was it therefore decided to investigate this class.

The parametric representation model that was chosen to be investigated where the SVI parametrization family since it seemed to have the most potential outside of its already strong foundation.

The SVI parametrization family is diveded into 3 parametrizations, the raw SVI parametrization, the SSVI parametrization and the eSSVI parametrization.

It was concluded that the raw SVI parametrization even though it gives very good market fits, was not robust enough to be chosen. This ment that the raw SVI parametrization would in most cases generate arbitrage in its surfaces.

The SSVI model was concluded to be a very strong model compared to the raw SVI, since it was able to generate completely arbitrage free solutions with good enough results.

The eSSVI is an extended parametrization of the SSVI with purpose to improve its short maturity results. It was concluded to give small improvements but with the trade of making the optimization procedure harder. It was therefore concluded that the SSVI parametrization might be the better application.

To try to improve the results of the SSVI parametrization was a complementary procedure developed which got named the calibrated SSVI method. This method compared to the eSSVI parametrization would not change the parametrization but instead focusing on calibrating the initial fit that the SSVI generated. This method would heavily improve the initial fit of the SSVI surface but was less robust since it generated harder cases for the interpolation and extrapolation.

Sammanfattning

I det här examensarbetet undersöks frågan om hur man bör modellera implied volatilitetsytor på ett robust och arbitragefritt sätt.

För att kunna veta om lösningarna är arbigtragefria började arbetet med en undersökning inom arbitrageområdet. De mest heltäckande resultatet som hittades var två theorem av Roper i [14]. Baserat på dessa theorem kunde två applicerbara arbitragetester skapas som sedan kom att bli en av hörnstenarna i detta arbete.

Genom att undersöka de modellklasser som verkade vara de bästa inom området valdes den parametriseringsbeskrivande modellklassen.

I denna klass valdes sedan SVI parametriseringsfamiljen för vidare undersökning eftersom det verkade vara den familj av modeller som hade störst potential att uppnå jämnvikt mellan enkel applikation samt bra resultat.

För den klassiska SVI modellen i SVI familjen drogs slutsatsen att modellen inte var tillräcklig för att kunna rekommenderas. Detta berodde på att SVI modellen i princip alltid genererade lösningar med arbitrage i. SVI modellen genererar dock väldigt bra lösningar mot marknadsdatan enskilt och kan därför vara ett bra alternativ om man bara ska modellera ett implied volatilitetssmil.

SSVI modellen ansågs däremot vara ett väldigt bra alternativ. SSVI modellen genererar komplett aribragefria lösningar men har samtidigt rimligt bra marknadspassning.

För att försöka förbättra resultaten från SSVI modellen, var en kompleterande metod kallad den kalibrerade SSVI metoden skapad. Denna metod kom att förbättra marknadspassningen som SSVI modellen genererade men som resultat kom robustheten att sjunka, då interpoleringen och extrapoleringen blev svårare att genomföra arbitragefritt.

Contents

1 Introduction

Cinnober (now a part of Nasdaq) has a system for clearing of financial transactions, TRAD-Express™ RealTime Clearing. This system is used by clearing houses that insert themselves as the counterparty to both the buyer and seller. The clearing house calculates a risk margin value that the buyer and seller have to post as collateral while the trade is being cleared.

Using an FHS VaR (Filtered Historical Simulation) approach has become the standard way to determine the base Initial Margin (IM) component of a portfolio. This method is mainly used for markets such as Cash-Equity and Fixed Income. For derivatives markets consisting of mainly Futures and Options the standard way is to use SPAN.

SPAN stands for standard portfolio analysis of risk. It is an old method that to some extent is depended on risk managers subjective judgement. SPAN has therefore met a lot of criticism and so the market trend is to move to VaR for these markets as well.

One of the main reasons why SPAN is still used is because it is difficult to create reasonable and robust scenarios based on historical data for option contracts. In order to create a robust scenario it is important to have reliable implied volatility surfaces but how to construct these surfaces is not clear. There has been a lot of investigation in the area the last 20 to 30 years and more are still being done but one solution has not been set. This thesis will therefore try to find a reasonable solution to this problem using the already existing material but also investigate further into that model and see if there exist areas that can be improved.

1.1 Problem

Our problem is to investigate how to construct the implied volatility surface in a robust and arbitrage free way.

With robust we mean that the method should be able to generate good solution in most cases and with arbitrage free solution we mean that the generated surface should not introduce any arbitrage opportunities.

Apart from this is our aim to find a solution that is as simple and practical as possible to assure that the method is reasonable to apply.

2 Background

In this chapter we will go through very lightly the general concepts that will be discussed a lot in the remaining report.

2.1 Option

An option is a contract between two parties, which states that the buyer of the contract get the *option* to buy or sell a stock for a specific price, known as the *strike price*, at a later date or interval. Options that let you buy a stock for the strike price is known as *call* options meanwhile options that let you sell a stock for the strike price is known as *put* options.

The most well known option type is the *European option.* A European option only lets the buyer to exercise the option at the maturity date. To price a European option the standard way is to use the Black-Scholes formula, defined as

$$C = S_0 N(d_1) - Ke^{-rT} N(d_2) \tag{1}$$

where

$$d_1 = \frac{ln(S_0/K) + (r + \sigma^2/2)T}{\sigma\sqrt{T}} \tag{2}$$

and

$$d_2 = \frac{ln(S_0/K) + (r - \sigma^2/2)T}{\sigma\sqrt{T}} = d_1 - \sigma\sqrt{T}. \tag{3}$$

As we see according to the formula, is the option price depended on the underlying price S, the strike price K, time to maturity T, interest rate r and the underlying volatility σ. The only parameter that we cant directly observe is the volatility, σ.

2.2 Implied Volatility

The Black-Scholes pricing formula was derived from assuming that the underlying could be described as a geometric Brownian motion, with an underlying volatility σ that is constant over all different contract parameter combinations. If we would back-out the volatility from the Black-Scholes formula, in eq.(1), by using market data and corresponding parameters (S, K, T, r), we should according to the theory behind the formula, always get the same volatility σ, but this is not the case in reality. Instead of getting a constant volatility over all contracts we get a smile. This phenomenon is called the *implied volatility.*

$$\sigma_{imp} = C^{-1}(C, S, K, T, r) = f(K, T \mid (C, S, r)) \tag{4}$$

Here C, S, r are constants meanwhile K and T are variables. This means that we can define the implied volatility as a function depended on 2 variables, which generates the *implied volatility surface.*

2.2.1 Empirical Characteristics

The implied volatility surface has under the past decades been heavily investigated. It has been observed empirically that the surface has some general characteristics. In regards of how the surface usually look like, the following *profile characteristics* can be stated,

1. The surface has a so called *smile* profile in the strike price depended direction (K-dependency) of the surface.
2. In the maturity direction the surface has a quite linear leaning profile, this profile is known as the *term structure*.
3. The curvature of the smile will flatten out with longer maturities. This is also known as *deformation*.

The implied volatility surface also change in time. The observed *time dependent characteristics* are mainly:

1. Implied volatility display high (positive) auto-correlation and mean-reversion. This is also known as the *volatility clustering*.
2. Returns of the underlying asset and return of implied volatility are negatively correlated. This is also known as the *leverage effect*.
3. Relative movements within the implied volatility surface have little correlation with the underlying.
4. The variance of the daily log-variations in implied volatility can be described with two to three principal components.

For more details in this area we recommend [1].

2.3 Arbitrage

Arbitrage is a phenomenon in the market when an opportunity arises where you as an investor could make an investment that has no cost, the possibility to earn you money but has no chance to lose you money. In other words arbitrage is a risk-free investment, the so called *free lunch*.

Arbitrage can in mathematical terms be defined as in [13].

Definition 2.1. *An* ***arbitrage*** *possibility on a financial market is a self-financing portfolio h such that,*

$$\begin{aligned} V(0;h) &= 0, \\ P(V(T;h) \geq 0) &= 1, \\ P(V(T;h) > 0) &> 0 \end{aligned} \tag{5}$$

where $V(\cdot)$ is the value process, $P(\cdot)$ the probability measure and T time to maturity. We say that the market is ***arbitrage free*** *if there are no arbitrage possibilities.*

The definition of arbitrage can be divided into two sub-categories, the *dynamic arbitrage* and the *static arbitrage*. Static arbitrage is arbitrage that exist in the present time meanwhile dynamic arbitrage would be opportunities that occur on the life time of the investment. This means that you would change your invested position as time goes. In the aspect of constructing an implied volatility surface the *static arbitrage* becomes more important to handle since each surface is defined in a set time.

In both the static and dynamic arbitrage cases the reason arbitrage exist is that the available instruments on the market are miss-priced relative to one another. For call or put options this can happen in two ways, miss-pricing between contracts with different strike prices K or different maturities T. In the case of different strike prices the arbitrage is known as *butterfly spread arbitrage* and in the case of different maturities it is known as *calendar spread arbitrage*.

In the case of the general option meaning both the call and put option we get a third possibility of arbitrage and that is the internal relationship between call options and put options. This relationship is known as the *put-call parity*.

Definition 2.2 (Put-Call Parity)**.**

$$C(K,T) + KB(t,T) = P(K,T) + S_t, \tag{6}$$

where S_t is the spot price at t.

If this relationship does not hold up there exist internal arbitrage for the option. This relationship is an example of *replication arbitrage*. In other words, a replication arbitrage is the case when two different positions with equivalent payoff functions do not have the same cost.

3 The Applicable Arbitrage Tests

In this chapter we will discuss the concept of arbitrage in regards of constructing implied volatility surfaces. We will present important results from previous investigation in the area of setting up conditions for arbitrage free priced options and then conclude the results by presenting very applicable arbitrage tests based on these results. These tests will be heavily used in the remaining report and is a corner stone to generate the arbitrage free solutions that we are searching for.

3.1 Why care about Arbitrage?

It can be assumed that most participants on the market wants to earn as much money as possible. Taking advantage of arbitrage could therefore be a great strategy but in the same manner protecting your position against arbitrage is also a good strategy, since the money someone earns from taking advantage of the arbitrage is an amount some participants looses. The only "fair" price is therefore the price that is arbitrage free. If you are a bigger institution that puts out a lot of prices on instruments for buyers, then it becomes even more important that these prices are not wrongly priced relative to each other because that would mean a very big loss for the institution.

3.2 Arbitrage Conditions for Options

Arbitrage-bounds for option prices is something that was developed long ago, Merton in [25] gives the starting point with a lot of literature following, for example we have Fenglers work in [5] which presented boundaries that implied *monotonicity*, *convexity* and a general pricing boundary. Carr and Wu in [27] gives a good summary of the conditions given by Merton. Niu in [22] also refer to there work when reviewing Ropers closely connected work in [14] and call the conditions the *Merton's bounds*. Roper gives though the most comprehensive result of the condition in [14]. He states that this theorem is "a necessary and sufficient condition for a call price surface to be free of static arbitrage". The result is supposedly to follow Lemma 7.23 in [28] but Roper points out that his conditions differ a bit but allow $K = 0$. This is Ropers conditions for call options.

Theorem 3.1 (Ropers Result I). *Let $s > 0$ be a constant spot price and denote $C(K, \tau)$ as the price of European call options where K is the exercise price of the option and $\tau = T - t$ is time to maturity which is the difference between today t and the maturity T.*

(a) Let $C : (0, \infty) \times [0, \infty) \longrightarrow \mathbb{R}$ satisfy the following conditions.

(A1) (Convexity in K)

$$C(\cdot, \tau) \text{ is a convex function}, \quad \forall \tau \geq 0;$$

(A2) (Monotonicity in τ)

$$C(K, \cdot) \text{ is non-decreasing}, \quad \forall K > 0;$$

(A3) Large strike limit

$$\lim_{K\to\infty} C(K,\tau) = 0, \quad \forall \tau \geq 0;$$

(A4) (Bounds)

$$(s-K)^+ \leq C(K,\tau) \leq s, \quad \forall K > 0, \tau \geq 0, \text{ and}$$

(A5) (Expiry Values)

$$C(K,0) = (s-K)^+, \quad \forall K > 0.$$

Then

(i) the function

$$\hat{C} : [0,\infty) \times [0,\infty) \to \mathbb{R} \tag{7}$$

$$(K,\tau) \mapsto \begin{cases} s, & \text{if } K = 0 \\ C(K,\tau), & \text{if } K > 0 \end{cases} \tag{8}$$

satisfies assumption (A1)-(A5) but with $K \geq 0$ instead of $K > 0$; and

(ii) there exists a non-negative Markov martingale X with the property that

$$\hat{C}(K,\tau) = \mathbb{E}((X_\tau - K)^+ \mid X_0 = s)$$

for all $K, \tau \geq 0$.

(b) All of the listed conditions in part (a) of this theorem are necessary properties of $\hat{C}$ for it to be the conditional expectation of a call option under the assumption that X is a (non-negative) martingale.

Remark: These conditions is given from the view of a European call option, to get the equivalent conditions for a put option we apply the put-call parity relationship stated in eq.(6).

3.3 Arbitrage Conditions for Implied Volatility

Roper also gives the equivalent arbitrage conditions for implied volatility. Niu in [22] gives a comprehensive overview of these conditions and shows how the most significant results in the area are linked to Ropers result. It is concluded that Roper offers sufficient conditions for the implied volatility surface to be free from static arbitrage, and as Niu states "is in a practical sense also necessary".

This is Ropers arbitrage conditions in regards to implied volatility surfaces.

Theorem 3.2 (Ropers Result II)**.** *Let $s > 0$, $x = ln(\frac{K}{s})$ and $\Sigma = \sigma_{imp}(x,\tau)\sqrt{\tau}$ satisfy the following conditions:*

1. *(Smoothness) for ever $\tau > 0, \Sigma(x,\tau)$ is twice differentiable w.r.t x;*

2. *(Positivity) for every $x \in R$ and $\tau > 0$, $\Sigma(x,\tau) > 0$;*

3. *(Durrleman's Condition) for every* $\tau > 0$ *and* $x \in R$,

$$0 \leq g(x) = \left(1 - \frac{x\Sigma_x}{\Sigma}\right)^2 - \frac{1}{4}\Sigma^2\Sigma_x^2 + \Sigma\Sigma_{xx} \tag{9}$$

4. *(Monotonicity in* τ*) for every* $x \in R$, $\Sigma(x,\tau)$ *is non-decreasing w.r.t.* τ*;*

5. *(Large moneyness behavior) for every* $\tau > 0$

$$\lim_{x\to\infty} d_+(x, \Sigma(x,\tau)) = -\infty; \tag{10}$$

6. *(Value at maturity) for every* $x \in R$,

$$\Sigma(x, 0) = 0. \tag{11}$$

Then,

$$\tilde{C} : [0,\infty) \times [0,\infty) \to R \tag{12}$$

$$(K,\tau) \to \begin{cases} s\Phi\left(\frac{-x+\frac{1}{2}\Sigma^2(x,\tau)}{\Sigma(x,\tau)}\right) - K\Phi\left(\frac{-x-\frac{1}{2}\Sigma^2(x,\tau)}{\Sigma(x,\tau)}\right), & \text{if } K > 0, \\ s, & \text{if } K = 0. \end{cases} \tag{13}$$

is a call price surface parameterized by s that is free of static arbitrage. In particular, there exists a non-negative Markov martingale X with the property that $\tilde{C}(K,\tau) = E[(X_\tau - K)^+ \mid X_0 = s]$ *for all* K, $\tau > 0$.

Remarks: Roper also proves that if Σ satisfy condition 1 and 2 but violates any of the remaining conditions 3-6, $\tilde{C}$ will not be a call surface free from static arbitrage.

In this result Roper uses a form of implied volatility that is called *total implied volatility.* There is other forms that we will see in the report. These forms are presented in Appendix A.

3.4 Arbitrage Tests

With Ropers arbitrage conditions we have a strong foundation to stand on. Based on Ropers result we can, as in [30], group the conditions that is linked with butterfly and calendar spread arbitrage.

Definition 3.1 (Butterfly Spread Arbitrage)**.** *For a fixed and positive real* τ, *the implied volatility smile* $\sigma_{imp}(\tau, K) \mid_{\tau=\tau_0}$ *is free of butterfly arbitrage if and only if condition 3 and 5 in Theorem 3.2 are satisfied.*

Definition 3.2 (Calendar Spread Arbitrage)**.** *An implied volatility surface* $\sigma_{imp}(\tau, K)$ *is free of calendar spread arbitrage if and only if condition 4 and 6 in Theorem 3.2 is satisfied*

Remarks: Condition 2 will always be true since we use the Black-Scholes transformation. Also since the modeling of the implied volatility surface is depended on the market it seems strange to see anyone setting negative prices on the surface, i.e giving away instruments. Condition 1 is also assumed to always be satisfied. This condition has only to do with how our model is working. This means that some models give smooth solution and some don't. We assume that the model that we will choose will give smooth solutions and so this condition is automatically satisfied.

Based on these definition we can now state the applicable arbitrage tests.

Definition 3.3 (Butterfly Spread Arbitrage Test)**.** *By plotting $g(x)$ from condition 3 in Theorem 3.2 against the log-moneyness, $x = ln\left(\frac{K}{S}\right)$ we get a graph that will indicate arbitrage opportunities in points that fall below 0. In other words if*

$$g(x) < 0, \tag{14}$$

there exist butterfly arbitrage.

Proof. In Appendix A this test is proved to work by applying it on a miss-priced case on a real stock. There the reader can also see a demonstration of the test. □

Definition 3.4 (Calendar Spread Arbitrage Test)**.** *By plotting the total implied variance $\omega_{imp}(x,\tau) = \tau\sigma_{imp}^2$ against the log-moneyness $x = ln\left(\frac{K}{S}\right)$ for all maturities $\tau_1 < \tau < \tau_2$ involved in the test, we get a graph that indicates arbitrage opportunities if the lines intersect. In other words if,*

$$\omega_{imp}(x,\tau_1) \leq \omega_{imp}(x,\tau_2), \tag{15}$$

for all τ then the solution is free of calendar spread arbitrage.

Proof. As for Definition 3.3 is this test proved to work by demonstrating it on a miss-priced case on a real stock in Appendix A. □

With these tests now defined, we have created a strong tool for investigating if our upcoming surfaces are arbitrage free or not.

4 Modelling Overview

At this time we know about the background of the report, we know what the implied volatility smile and surface is and how we can test if there exist arbitrage in it.

In this section we will start investigating the question of how we actually should model the implied volatility. We will look into the general methodology to different categories of models and discuss the positive and negative sides with them. The aim is to choose one or more models to investigate further. The chosen model should be as simple as possible but in the same time generate arbitrage free solutions and good market fits. We also want our model to create smooth solutions to satisfy condition 1 in Ropers arbitrage theorem as we discussed in the previous chapter. Apart from this we also want our modelled surfaces to be easy to save and interact with.

4.1 Modelling Strategies

There exist a lot of different models in the area of modelling the implied volatility. There has also been as an result, investigations in the area to compare and showcase most of these models. A few examples would be [6, 7]. By investigating these articles and reports it is evident that the different models can be divided into two categories. These categories are defined by what general strategy the models are based on. The two categories are as follows:

- models based on pricing options directly,
- models based on directly modelling implied volatility.

4.2 Models based on directly pricing options

The most famous pricing model is the Black-Scholes formula presented in eq.(1). This model is unfortunately incomplete since it is based on a assumption that do not match with the market. The Black-Scholes formula assume that the underlying for some option has constant volatility for all different parameter combinations, (K, T). This assumption seems reasonable but it has been shown by backing out the volatility from the Black-Scholes formula, using market data, and getting the so called implied volatility, that this is not the case! The market sets its own prices depended on the risk of the contract and are not thinking of satisfying the reasonable mathematical theory that Black-Scholes imposes. Different models has therefore been developed through the years that aim to fix the erroneous assumption which the Black-Scholes model makes and in other words create a model that performs better then Black-Scholes. This is the main idea of the models included in this category.

The methodology for using these models to model implied volatility could be generalized to be as follows:

1. We have data on option prices.
2. Fit your model onto the market data.
3. Transform the modeled prices into implied volatility, using the Black-Scholes formula.

The big thing here to notice is that these models are not made for directly modelling the implied volatility but to try to model the market behavior of the contract price and underlying. The assumption is that if we can model the contract behavior well, this should indirectly also model the implied volatility well. As stated above will we arrive at our implied volatility surface by transforming the price surface that we get from these models, by using the Black-Scholes formula stated in eq.(1). The Black-Scholes formula is also as stated previously seen as a mere transformation between prices and implied volatility.

The maybe biggest and most promising model classes in this category is the *Tree models* and the *Stochastic volatility models*.

4.2.1 Tree Models

When discussing tree models we usually talk about the *Binomial model*. The Binomial model is a model which prices derivative (mainly options) by using binomial trees. The binomial tree are used to represent all different paths the underlying price can take. In fig.(1) we can see an example of how that could look like. The model assumes that the underlying is following a *random walk* with predetermined probabilities for moving up or down. From this price tree we calculate the option price by going backwards from the last step. When doing so we are using risk-neutral valuation to make sure that we get an arbitrage free price. Notice that the arbitrage that are taken into account is the dynamic arbitrage. To see the formulas and more details about the model, we recommend to read [12, 13]. It is also worth mentioning that there is other tree models, for example the *trinomial model* but the general approach by pricing the option by creating a tree of possible paths remains the same.

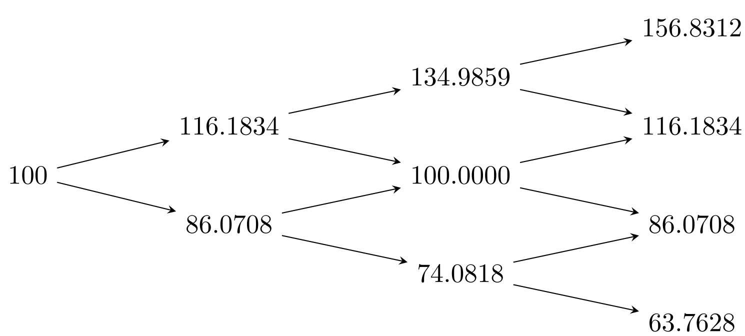

Fig. 1: Underlying dynamics with a binomial tree.

One big drawback to the tree models is that to get reasonable results we need to use many time steps. The problem is though that the trees grow exponentially depending on how many steps it uses. There is in other words a trade-off between very large trees but more accurate result and smaller trees and less accurate results. In the case of the Binomial model, 30 steps is a common practise and generates a tree with $2^{30} \approx 10^9$ different paths. If you would follow the regular calculation procedure you need to calculate a few operations at each node and so the number of operations that is needed for one option price can be very large!

If we would use this model to model the implied volatility it is worth noting that we would not only need one tree. The tree models similar to Black-Scholes model assume that the underlying behavior is constant for all different parameter combinations in the surface. This means that we would only use one tree for the whole surface, the problem is that if we did that it would amount to a surface that is a constant plane. To use the tree models we therefore need to fit a unique tree for each data point or for a number of sets of data points in the surface. This is not hard when you have data on some option prices but the big question is how to model the interpolated points which we do not have data for? In those cases you either need to make some assumption about the option price or use historical data and project or simulate its price. How to manage this part is a big question mark but the progress being made in the area of machine learning might have the solution.

Assuming that we have a way to generate all the trees that we need. Next step would be to generate the price surface. That is achieved by following the regular approach for these models [12]. The price surface is then transformed into the implied volatility surface by using the Black-Scholes formula.

Since this is a numerically determined surface, you need to save the surface in data points. This is another big drawback for these models. If we want to save a lot of surfaces then you will need a lot of memory to manage that. If you also want to be able to redo the calculations then you also need to save the generated trees which would amount to even more memory needed. On this reason alone these models might not be the best choice.

4.2.2 Stochastic Volatility Models

The stochastic volatility models are a class of models based on assuming that the implied volatility is a stochastic risk-factor. Examples of models is, Heston, Bates (SVJ), BNS, NIG-CIR and SABR. A good read for understanding how to apply these models is [38, 37]. The general idea is that each model make some assumption on how the underlying and the related risk-factor (the implied volatility mostly) is behaving. This behavior is represented in stochastic differential equations (SDE). From these SDEs you will in general not be able to derive a closed formula as in the Black-Scholes case, instead we use a complementary pricing method that uses the behavior stated in the SDE and with it we arrive at our price. The 3 mostly used pricing methods is Direct Intergration, Fast Fourier Transformation (FFT) and fractional FFT but there is also others for example Monte Carlo simulation. To be able to execute the calculations, different numerical methods is needed for example *Gaussian quadrature*.

The Stochastic volatility models, as the tree models, try to fit its stochastic behavior to how the historical behavior looks like. Having a model that mimics the market behavior is a very promising property and could be useful for interpolation problems in the surface but manage to fit the model well against the historical data is not the easiest. The devil is in the details and how to estimate the parameters is the biggest problem. Depending on what model you chooses this method will also look different but if you could get your fitting procedure to be effective enough, this class of models has a strong case to be chosen.

Also compared to the tree models, is the estimated surfaces saved with the corresponding parameters instead of having the whole surface in data points. This make it so that it is simpler to interact with the model. There is although cases when simulation is needed to get your result and so in those cases that makes it not as effective.

Apart from the fitting part of the model we are worried about how easy it is to control static arbitrage in the surfaces. The stochastic volatility models in general are mostly trying to mimic the stochastic behavior and adjust to make sure that we do not create dynamic arbitrage prices. We are worried that these models will therefore miss static arbitrage.

Assuming that we would go with the stochastic volatility models the question then only becomes which stochastic volatility model you should use. Since there is a lot of variation this is not an easy task, but for simplicity the Heston model seems to be somewhat of a standard choice even though it is not the most comprehensive. For more about that model we recommend to read [40, 41, 42]. Other models that seems promising is the Bates model which is an extension of Heston which allows for jumps to occur in the stochastic behavior. This is more aligned with the market since as we know is the market not behaving completely continuous which the Heston model assumes. With a model that allows for jumps in its stochastic behavior you can adapt to behaviors like when big amounts of assets are being bought and pushing the market to another level. The problem though with these models is that you need to make assumptions about how the jumps will behave and how to do that in a reasonable way is not clear. For more details about both the Heston and the Bates model we recommend [43].

4.3 Models based on directly model implied volatility

The idea of trying to model the option behavior directly and in-directly get the implied volatility seems like a reasonable approach but it might be an ineffective approach. If we have a model that we can directly apply on the implied volatility, the calculation time should in theory be minimized. What more is that in this way we can adapt better to how the implied volatility is behaving instead of trusting that there will not be any non-linear errors when transforming into the implied volatility.

The methodology for a model which is based on directly modelling implied volatility would look something like,

1. We have data on option prices.
2. Transform the price data into implied volatility
3. Fit your model onto the market data.

Note here that it is actually only step 3 where we are using our model meanwhile step 1 and 2 can be seen as preconditions.

The two maybe most successful subcategories of this model class is,

- non-parametric representation models,
- parametric representation models.

4.3.1 Non-Parametric Representation Models

The non-parametric representation models are a class of models where we use interpolation or direct fitting methods to create a curve that align with the market data. An example of this type is the *Penalized spline*, other examples of methods is [17, 44]. This class of models is very straight forward, all we really need is the market data and then we can estimate our solution but the strength in its simplicity is also its biggest weakness. Since the models are so adaptive and not restricted to any shape or assumption these models become very depended on that the initial market data is good. This means that if the market data not initially are arbitrage free will these model with a high probability keep that arbitrage. There is also high chances that the model might over fit. In those cases we might not even get solutions that are showing the implied volatility properties. Of course are these problems something each model wrestle with and try to control but compared to the other model classes this might have the weakest behavior keeping property.

Apart from the properties in the result is there also a big drawback with how we save the surfaces. As in the tree models will the surface be defined in data points meaning that to save the surface we need to save all those points.

If we do not want to save the surface in data points the fitting procedure needs to be redone each time we want the surface.

4.3.2 Parametric Representation Models

The parametric representation models is a class of models where the goal is to try to generate a good representation of the market data using a chosen parametrization. Examples of these type of models is, Polynomial Parametrization, The Stochastic Volatility Inspired (SVI) model family, RFSV and IVP. Some of these models are just based on an ansatz that seems reasonable meanwhile others are based on trying to mimic the behavior of the market. One example and maybe the most interesting one is the Stochastic Inspired Volatily model which is a parametrization that is inspired by the Heston model which we previous talked about. The difference between the SVI and the Heston model is that the SVI have a closed formula that we try to fit directly on the market data. This solves a big part of the problems with the Heston model or any other stochastic volatility model and is a strong argument for the SVI model.

After we have chosen a parametrization to use and fitted it against the marked data we get the implied volatility surface by interpolate in between the fitted slices. Some of the models in the parametric representation class have built in interpolation in its parametrization and makes this step very simple meanwhile others don't. In those cases the interpolation in between slices is a area where the chance for introducing arbitrage is higher.

Apart from the interpolation problem the parametric representation models seem to be the class of models where we have the most direct control of the smile and surface. In our view we think this is the class that has the biggest potential for handling the static arbitrage.

Another strong argument for using parametric representation models, is that compared to the non-parametric models and the tree models will these type of models be saved in a couple of parameters. This makes it so that the generated surface is very easy to interact with and will not require a lot of memory to save. What also is a nice property, is that since we have a parametrization for the surface, we can by knowing the parameter values, take out whatever volatility we want. This was not the case when we got surfaces defined in data points. In that case you need before generating your surface, know what points you want.

4.4 Choice of Model

The two categories of models that we see as the contenders is the stochastic volatility models and the parametric representation models. Both classes is very similar in a way apart from being located in different domains (option price and implied volatility).

We would say that the stochastic volatility models is the more sophisticated model class and probably have a higher chance of achieving a nicer looking surface but the big problem with this class is that to arrive at the surface the computations might be very heavy and not that effective. Here the parametric representation models have a big advantage. They usually are not depended on heavy numerical methods and approximations to achieve there result, usually there is only some big optimization that need to be completed. So in regards to our initial condition of simplicity the parametric representation models seem to win against the stochastic volatility models.

In regards of arbitrage the parametric representation models also seem to be the class of models that have a higher potential to control the static arbitrage meanwhile the stochastic volatility models are more focused on the dynamic arbitrage. For generating the implied volatility surface the static arbitrage is more important since the surface is defining the market at a set time. So in this regard the parametric representation models seem to win as well.

When it comes to modelling the market behavior in the other hand the stochastic volatility models seems to win over the parametric representation models. Most of the parametric representation models are based on trying to mimic the look of the implied volatility surface meanwhile the stochastic volatility models are based on describing the underlying stochastic behavior of the market. Even though if the underlying behavior is modelled wrongly the idea of actually trying to understand the market behavior is a nice property and gives the stochastic volatility models a bigger potential of performing better then the parametric representation models. The choice between the stochastic volatility models and the parametric representation models seem to be a question about a trade-off between simplicity and more sophisticated results. In other words the tipping point in regards of what model class we should go with is depended on how well the parametric representation models can perform. If a parametric representation model would perform well then the easy applicability of the fitting procedure and the easier control of the static arbitrage would be a strong argument for using these types of models.

The model that seems to have a high potential to achieve this is the SVI model family. As mentioned previously is this model based on the Heston model and so therefore it gets more of the market modelling property that the parametric representation models in general lack.

We will therefore conclude this chapter by choosing to investigate further the parametric representation model of the SVI model family and hope to find a way to achieve improved results and solving the trade-off question.

5 The SVI parametrization

In this chapter we will present a more detailed explanation about the SVI parameterization family. We will go through the different parametrization that is of interest and show strategies to fitting them against market data. We will show a weighting strategy that will be crucial to focus the fit ATM and we will discuss how to interpolate and extrapolate the surface. Lastly we will present a calibration method that can eliminate arbitrage and improve some fits.

5.1 Background

The stochastic volatility inspired (SVI) model is a parametric representation model for stochastic implied volatility. It was developed at Merill Lynch 1999 and was made public 2004 through Gatherals presentation in [15]. Since then a lot of investigation has been made regarding the model. One particular interesting development was the Quasi-Explicit parametrization introduced 2009 in [31] which allowed the procedure of finding the models parameters become faster. Roper in [14] introduced 2010 the comprehensive arbitrage theorem that we showed in Theorem 3.2. In that paper he showed that Gatherals SVI parametrization were in fact in general not arbitrage free as they claimed. Gatheral introduced therefore a new parametrization, 2013 in [16], called the surface SVI or SSVI. This parametrization was a simplification of the SVI parametrization which uses an ATM dependency, making the fit in general, free of calendar arbitrage. The problem with the SSVI parametrization is though its stale fitting property. To solve this problem Seba Hendriks in [34] introduced the extended SSVI parametrization, eSSVI for short, which made the SSVI parametrization a bit more flexible.

5.2 Parametrization

The SVI parametrization, is a family of parametrization, meaning there is multiple formulas that builds on the same framework but has minor differences for solving specific problems. The general form - the formula you will most likely find if you search on the SVI - also known as the *raw parametrization*, reads as follows.

Definition 5.1. *The raw SVI parametrization of the total implied variance for a fixed time to maturity reads,*

$$\omega_{imp}^{SVI}(x) = a + b\left(\rho(x-m) + \sqrt{(x-m)^2+\sigma^2}\right), \tag{16}$$

where x is moneyness and $\{a, b, \sigma, \rho, m\}$ is the parameter set.

Note that the SVI parameter σ is not to be confused with the volatility of the underlying's price process, which is also denoted σ.

The parameters of the raw parametrization is just adaptable parameters to fit onto the market data. The different parameters affect the smile in different ways,

a changes the vertical translation of the smile in the positive direction,

b affect the angle between the put and call wing,

ρ rotate the smile,

m changes the horizontal translation of the smile,

σ reduces the at-the-money curvature of the smile.

These parameters can be tricky to get full control over and so to be more intuitive to trader we have the *jump-wing parametrization*. This parametrization does not build on one form that will define the one smile, instead it describes that smile by 5 different values. You will in other words not get a "interpolation form" just a representative explanation of the important aspects of the smile. This property can be used for easier adjustment of the respective smiles.

Definition 5.2. *The jump-wing (JW) parametrization defined in terms of the raw parameters, is defined as,*

$$\begin{aligned}
v_\tau &= \frac{a + b\left(-\rho m + \sqrt{m^2+\sigma^2}\right)}{\tau}, \\
\psi_\tau &= \frac{1}{\sqrt{\omega_\tau}}\frac{b}{2}\left(\rho - \frac{m}{\sqrt{m^2+\sigma^2}}\right), \\
p_\tau &= \frac{1}{\sqrt{\omega_\tau}} b(1-\rho), \\
c_\tau &= \frac{1}{\sqrt{\omega_\tau}} b(1+\rho), \\
\hat{v}_\tau &= \frac{1}{\tau}\left(a + b\sigma\sqrt{1-\rho^2}\right),
\end{aligned} \tag{17}$$

where $\omega_\tau = v_\tau \tau$ and τ is the time to maturity.

This parametrization depends explicitly on the time to maturity τ. These values has the following interpretation,

v_τ gives the ATM implied total variance, $\omega(0)$,

ψ_τ gives the ATM skew, $\partial_x \omega(0)$,

p_τ gives the slope of the left wing,

c_τ gives the slope of the right wing,

$\hat{v}_\tau$ is the minimum implied total variance, $\min(\omega(x))$.

The inverse transformation of eq.(17) back to the *raw* parameters in eq.(16) is given by the following Lemma.

Lemma 5.1. *Assume that $m \neq 0$. For any $\tau > 0$, define the (t-dependent) quantities:*

$$\beta = \rho - \frac{2\psi\sqrt{\omega_\tau}}{b} \qquad \textit{and} \qquad \alpha = sign(\beta)\sqrt{\frac{1}{\beta^2} - 1}. \tag{18}$$

where we have further assumed that $\beta \in [-1, 1]$ (this is equivalent to the condition for convexity of the smile). Then, the raw SVI and SVI-JW parameters are related as follows:

$$\begin{aligned} b &= \frac{\sqrt{\omega_\tau}}{2}(c_\tau + p_\tau), \\ \rho &= 1 - \frac{p_\tau\sqrt{\omega_\tau}}{b}, \\ a &= \tilde{v}_\tau\tau - b\sigma\sqrt{1-\rho^2}, \\ m &= \frac{(v_\tau - \tilde{v}_\tau)\tau}{b\{-\rho + sign(\alpha)\sqrt{1+\alpha^2} - \alpha\sqrt{1-\rho^2}\}}, \\ \sigma &= \alpha m. \end{aligned} \tag{19}$$

If $m = 0$, then the formulae above for b, ρ and a still hold, but $\sigma = (v_\tau\tau - a)/b$.

Proof. Proof for this lemma is omitted to Gatherals work in [16]. □

Note that by Definition 5.2 and Lemma 5.1 we have the possibility to jump between the raw and the jump-wing parametrization. This is a strong tool that will be used a lot in the application of the model.

Gatheral in [16] introduces a new parametrization called the *Surface SVI (SSVI)*. This is a parametrization that compared from the raw parametrization in eq.(16) takes the corresponding smiles into account when fitting the parametrization. It does this by depending on the ATM total implied variance, denoted as θ_τ. Note that it uses the same symbol as the total implied volatility in eq.(39) and should not be confused with.

Definition 5.3. *The Surface SVI (SSVI) parametrization of the total implied variance for a fixed time to maturity reads,*

$$\omega_{imp}^{SSVI}(x;\theta_\tau) = \frac{\theta_\tau}{2}\left(1 + \rho\varphi(\theta_\tau)x + \sqrt{(x\varphi(\theta_\tau)+\rho)^2 + (1-\rho^2)}\right). \tag{20}$$

where x is moneyness, θ_τ is the total variance ATM ($x = 0$), $\rho \in [-1, 1]$ and is constant over all smiles and $\varphi(\theta_\tau)$ is some smooth function depended on θ_τ.

The choice of $\varphi(\theta_\tau)$ is arbitrary and up to the practitioner. A good choice is the power law family where $\varphi(\theta_\tau) = \eta\theta_\tau^{-\lambda}$ where from our experience $\lambda \geq 0$ and $\eta \geq 0$. As the raw parametrization, is the SSVI parametrization also linked to the jump-wing parametrization according by the following Lemma.

Lemma 5.2. *The JW parameters corresponding to the SSVI parametrization read as follow,*

$$\begin{aligned} v_\tau &= \frac{\theta_\tau}{\tau}, \\ \psi_\tau &= \frac{1}{2}\rho\sqrt{\theta_\tau}\varphi(\theta_\tau), \\ p_\tau &= \frac{1}{2}\rho\sqrt{\theta_\tau}\varphi(\theta_\tau)(1-\rho), \\ c_\tau &= \frac{1}{2}\rho\sqrt{\theta_\tau}\varphi(\theta_\tau)(1+\rho), \\ \hat{v}_\tau &= \frac{\theta_\tau}{\tau}(1-\rho^2). \end{aligned} \tag{21}$$

Proof. The proof for this result is omitted to Alexanders work in [30]. □

To make the SSVI parmetrization more flexible but still have its nice arbitrage properties, Hendriks in [34] extended the SSVI parametrization by changing the constant parameter ρ into as function depending on θ_τ, $\rho(\theta_\tau)$.

Definition 5.4. *The extended SSVI (eSSVI) parametrization of the total implied variance for a fixed time to maturity reads,*

$$\omega_{imp}^{eSSVI}(x;\theta_\tau) = \frac{\theta_\tau}{2}\left(1+\rho(\theta_\tau)\varphi(\theta_\tau)x + \sqrt{(x\varphi(\theta_\tau)+\rho(\theta_\tau))^2 + (1-\rho(\theta_\tau)^2)}\right). \tag{22}$$

where x is moneyness, θ_t is the total variance ATM ($x = 0$), $\rho(\theta_\tau) \in [-1,1]$ and $\varphi(\theta_\tau)$ is some smooth function depending on θ_τ.

A recommendation for the practitioner is to use $\rho(\theta_\tau) = ae^{-b\theta_\tau} + c$ which is an direct result from Hendriks own work in [34].

As the SSVI parametrization can the eSSVI parametrization use Lemma 5.2 to transform the parametrization into the jump-wing parametrization. Also note that this link makes it possible to transform the SSVI and the eSSVI into the raw parametrization by combining Definition 5.2 and Lemma 5.2.

5.3 Fitting the smile

For this subsection we will look at how to fit our parametrization onto market data.

5.3.1 Method 1: The SVI fit (slice-to-slice)

This method is based on using the raw parametrization in eq.(16). We want to find the best fit for the parametrization against the given market data and so by using least-squares we can define the optimization problem as,

$$\min_{a,b,\sigma,\rho,m} \sum_{i=1}^{n} w_i \left(\omega_{raw}(x;a,b,\sigma,\rho,m) - \hat{\omega}_i\right)^2, \tag{23}$$

where ω_{raw} is the raw parametrization formula in eq.(16) depending on the parameter set (a, b, σ, ρ, m), $\hat{\omega}_i$ the given market data defined in total implied variance according to eq.(41) and w_i is weights for defining the goodness of different data points.

This non-linear optimization problem can be quite computational heavy to solve straight on. To make the problem simpler we apply the *Quasi-Explicit parametrization* proposed in [31]. Let

$$y(x) = \frac{x - m}{\sigma}, \tag{24}$$

under this change of variable, the total implied variance in the raw SVI parametrization reads,

$$\begin{aligned} \omega_{raw}(x) &= a + b\sigma(\rho y(x) + \sqrt{y(x)^2 + 1}) \\ &= \hat{a} + dy(x) + cz(x), \end{aligned} \tag{25}$$

where

$$\begin{aligned} \hat{a} &= a, \\ c &= b\sigma, \\ d &= \rho b\sigma, \\ z(x) &= \sqrt{y(x)^2 + 1}. \end{aligned} \tag{26}$$

This means by picking a (σ, m)-pair we transform our non-linear problem into a multi linear regression problem which can be solved very fast by one matrix operation. The proof for this solution is demonstrated in Appenix A. We state the solution directly here.

With (σ, m) picked the optimization problem is solved by

$$\boldsymbol{\beta} = \left(\boldsymbol{X'WX}\right)^{-1} \boldsymbol{X'WY} \tag{27}$$

where

$$\boldsymbol{\beta} = \begin{bmatrix} \hat{a} \\ d \\ c \end{bmatrix}, \quad \boldsymbol{X} = \begin{bmatrix} 1 & y(x_1) & z(x_1) \\ \vdots & \vdots & \vdots \\ 1 & y(x_n) & z(x_n) \end{bmatrix}, \quad \boldsymbol{Y} = \begin{bmatrix} \omega_1 \\ \vdots \\ \omega_n \end{bmatrix}, \tag{28}$$

and $\boldsymbol{W}$ is the weight matrix defined as a diagonal matrix with each corresponding weight w_i as its diagonal elements. With this procedure we will find the best fit possible for the raw SVI parametrization.

The choice of (σ, m) can be done with Nelder-Meads algorithm or other non-linear optimization algorithms.

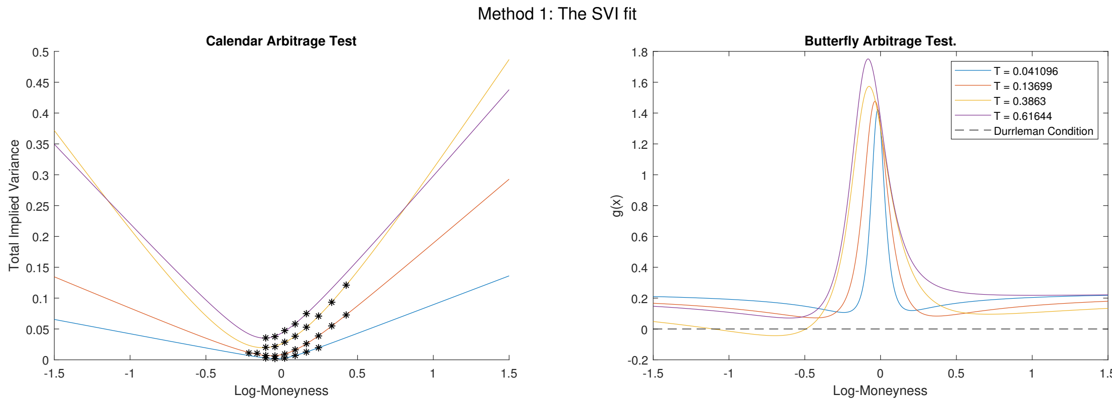

Fig. 2: Example of Method 1. *(Left)* Plot of the total implied variance smiles against market data, if any curve cross each other there exist calendar arbitrage between them. *(Right)* Butterfly arbitrage test, if the smile fall below 0 there exist butterfly arbitrage.

Note that since each fit of the SVI is done independently, each fit becomes very good but if the market data is not perfectly aligned there is a high chance that the fit introduces static arbitrage. This can be seen in fig.(2) where the SVI fit has introduced both calendar arbitrage and butterfly arbitrage according to the arbitrage tests that we defined in section 3.

In fig.(3) we can see the corresponding surfaces. In this case we have not interpolated in between the slices that we have fitted but just plotted them using MATLABs command *mesh*. The function default solution is to apply linear interpolation in between and so that is what we see.

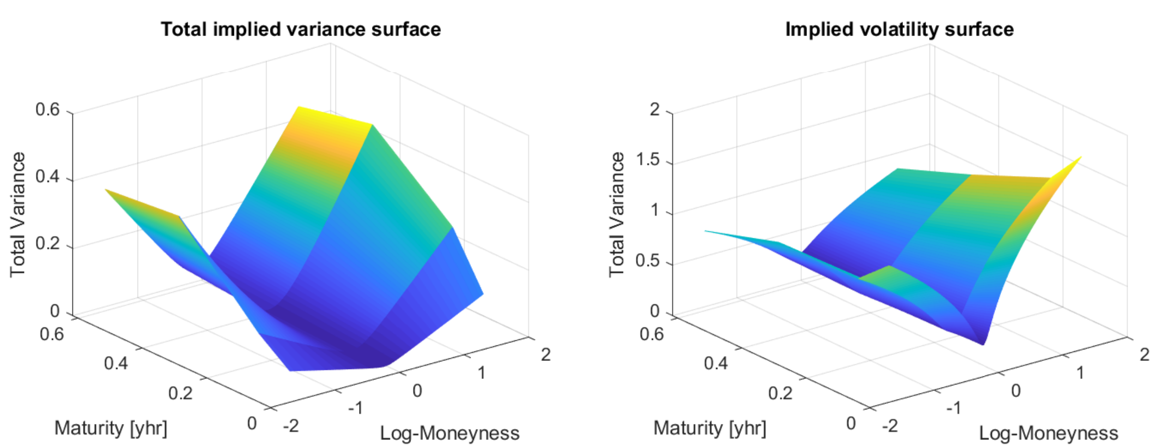

Fig. 3: Surfaces corresponding to fig.(2) using linear interpolation in between slices. *(Left)* Total implied variance surface. *(Right)* Implied volatility surface.

5.3.2 Method 2: The xSSVI fit.

This method is based on the SSVI and the eSSVI parametrization. We start by observing directly the ATM total implied variance. This can be done in different ways, if your are lucky the data might have a point that is located ATM but if not some interpolation method needs to be used. In this case we use method 1 to fit the best parametrization on the data and then pick out the ATM value, θ_τ. In general the values should be ordered in a non-decreasing order, if that is not the case the market data will in its definition already introduce calendar arbitrage. Assuming this is not the case, we continue and try to fit the parametrization onto the whole market data. The SSVI and eSSVI parametrization is two forms of the same parametrization and its only difference is what parameters we define with a function. The general optimization problem can therefore we stated as,

$$\min_{\rho(\theta_\tau),\varphi(\theta_\tau))} \sum_{s=1}^{S}\sum_{i=1}^{n} w_{s,i}\left(\omega_{s,i}(x,\theta_\tau;\rho(\theta_\tau),\varphi(\theta_\tau)) - \hat{\omega}_{s,i}\right)^2, \tag{29}$$

where S is the number of maturities we have in our data, n is the number of data points per maturity, $\hat{\omega}_{s,i}$ is the market data and $\omega_{s,i}(x,\theta_\tau;\rho(\theta_\tau),\varphi(\theta_\tau))$ is the parametrization.

If we defined $\rho(\theta_\tau) = \rho$ and $\varphi(\theta_\tau)) = \eta\theta_\tau^{-\lambda}$ we have the SSVI parametrization. Which generate the following optimization problem,

$$\min_{\rho,\eta,\lambda} \sum_{s=1}^{S}\sum_{i=1}^{n} w_{s,i}\left(\omega_{s,i}^{SSVI}(x,\theta_\tau;\rho,\eta,\lambda) - \hat{\omega}_{s,i}\right)^2. \tag{30}$$

where η and λ is constants.

If we define $\rho(\theta_\tau) = ae^{-b\theta_\tau} + c$ and $\varphi(\theta_\tau)) = \eta\theta_\tau^{-\lambda}$ we get the extended SSVI (eSSVI) parametrization. Which generates the following optimization problem,

$$\min_{a,b,c,\eta,\lambda} \sum_{s=1}^{S}\sum_{i=1}^{n} w_{s,i}\left(\omega_{s,i}^{eSSVI}(x,\theta_\tau;a,b,c,\eta,\lambda) - \hat{\omega}_{s,i}\right)^2, \tag{31}$$

where a, b, c is also constants.

Note that the function defined for $\rho(\theta_\tau)$ and $\varphi(\theta_\tau)$ can be changed to whatever arbitrary function needed. The function presented here is though the ones mostly used in academic investigations.

The previous optimization problems is non-linear least-square problems and so to solve them, non-linear optimization algorithm is needed. What choice that is the best is up to the practitioner to decide depended on there situation. In our case we have used MATLAB's own built in optimization algorithm *lsqnonlin()* which uses a Levenberg-Marquardt optimization algorithm, for more details about this algorithm we refer to [39].

In fig.(4) we can see the SSVI fit on the same market data as in fig.(2) and in fig.(5) we can see that market fits corresponding surfaces.

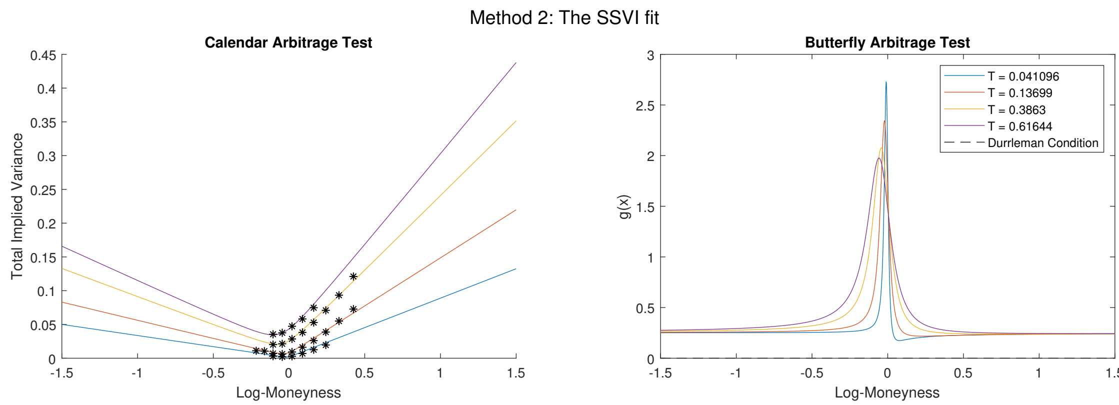

Fig. 4: Example of Method 2 using $\rho(\theta_\tau) = \rho$ and $\varphi(\theta_\tau)) = \eta\theta^{-\lambda}$. *(Left)* Plot of the total implied variance smiles against market data, if any curve cross each other there exist calendar arbitrage between them. *(Right)* Butterfly arbitrage test, if the smile fall below 0 there exist butterfly arbitrage.

The eSSVI fit gives pretty much the same result but with a small difference for the shorter maturities and so we only show the SSVI fit here but for comparison between all three fits we have done an experiment that is presented in Appendix A.

We can see compared to the SVI fit in fig.(2) that the both the calendar arbitrage and the butterfly is gone.

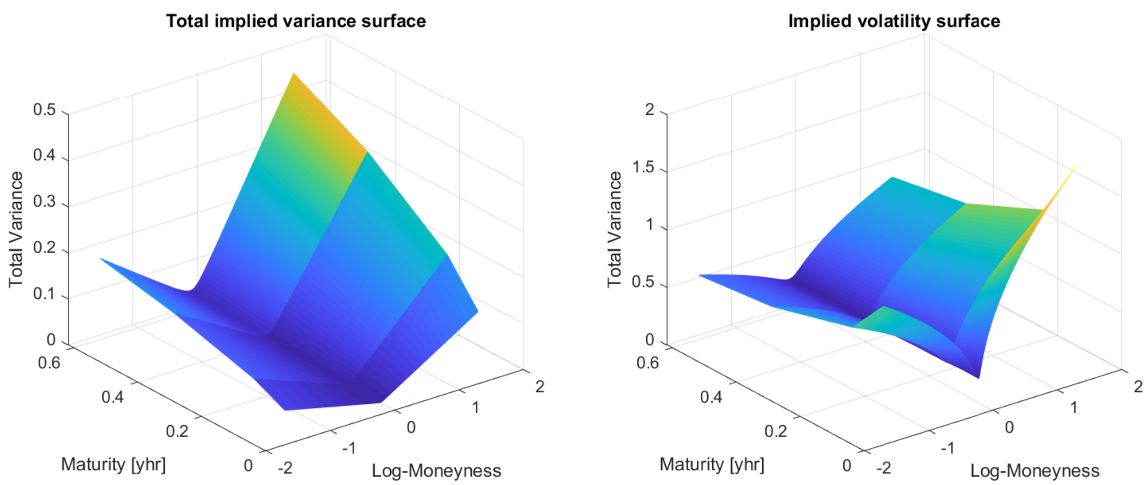

Fig. 5: Surfaces corresponding to fig.(4) using linear interpolation in between slices. *(Left)* Total implied variance surface. *(Right)* Implied volatility surface.

5.4 Weights

To handle different qualitative data points we apply weights in the optimizations procedure. How the weight is determined is up to the implementer but there is a couple of ways that are quite reasonable. Cited in [30] and in [16] the usage of the greek Vega is something of a standard practitioner choice. For the reader that is not briefed on the greeks, they are values that explain the options price change depending on some parameter. In the case of the vega, ν, it is a value on the options price change in regards of the volatility change. The vega is defined in mathematical terms as reads,

$$\nu = S\sqrt{T}\phi(d_1) = S\sqrt{T}\frac{e^{\frac{-d_1^2}{2}}}{\sqrt{2\pi}} \tag{32}$$

where S is the underlying, T the time to maturity, $\phi(\cdot)$ the normal density function and d_1 defined as follows,

$$d_1 = \frac{ln\left(\frac{S}{K}\right) + \left(r + \frac{\sigma^2}{2}\right)T}{\sigma\sqrt{T}}. \tag{33}$$

It happens that the majority of the traded volume is located in the area ATM. Therefore weights that are heaviest there sounds like a good idea. If we look at the vega behavior plotted against moneyness in fig.(6) we can see that it gives a good approximation on where we would like the weight to be located. Also since the vega in itself give the understanding that the price around the area ATM is the most affected by volatility changes, that indicates that those points are more prevalent for error. This means that we want our fits to have the lowest error against the market data in the area around ATM.

There is also other methods of applying weights. Aurell in [30] uses a combination of the vega weights and the traded volume of each data point. This alternative is though depended on that you actually have the traded volume which is not always the case.

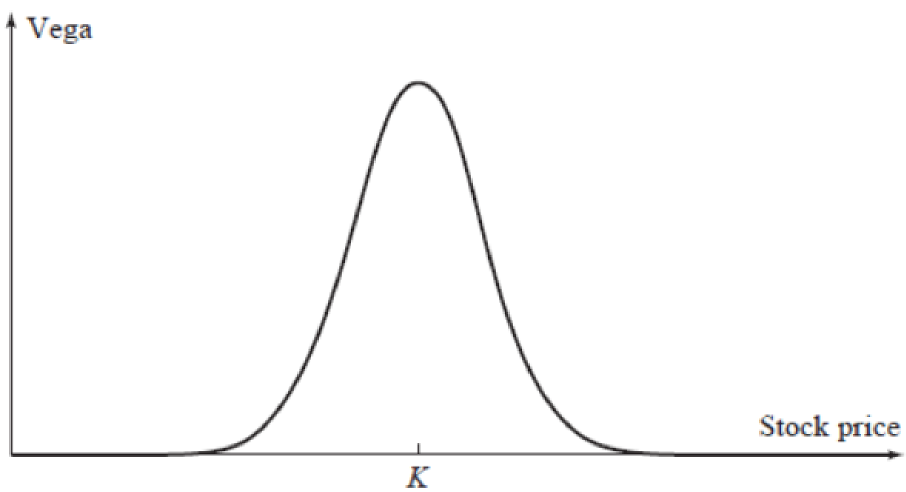

Fig. 6: This is a reprint of figure 18.11 in [12] which demonstrate how the vega changes for an option over its strike price K.

5.5 Interpolation

With market data fits that we are happy with, next step is to generate the complete implied volatility surface by interpolating the areas in between the fitted slices. This an important area if we want our surface to be completely arbitrage free. It is important that apart from satisfying our arbitrage tests, that we presented in section 3, that the interpolation method also satisfies the smoothness condition that Roper stated as condition 1 in theorem 3.2. As we can see in fig.(3) and fig.(5), the surfaces are full of edges which indicates that they are not satisfying the smoothness condition. It's therefore also clear that linear interpolation even though that it's a simple approach do not cut it as the interpolation method of choice.

We need an interpolation method of a higher order that by knowing the fitted slices can give us a continuous surface without edges. What more is that we also want our interpolation method to not introduce new arbitrage opportunities when our market data fit is arbitrage free.

In the investigation around the interpolation it became clear that most thesis and presentations in the area are usually not doing anything more then using the linear interpolation approach. For this thesis our goal was to find a method that could generate a completely arbitrage free surface and so for the case of interpolation we had to do our own empirical investigation. Our investigation was mainly based on trial-and-error. This means that we tried different approaches and by experiencing the different results we arrived at some understanding about the area. In this section we will present these findings but they will not be proved here with empirical evidence and therefore can only be seen as a recommendation for the reader.

5.5.1 Empirical findings

There are two general strategies for interpolating between the smiles, one alternative is to directly interpolate between the slices defined in total implied variance and the second approach is to interpolate between the smiles parameters.

The first approach have in general a higher capability to generate calendar arbitrage free solutions, assuming that the slices used are not already introducing calendar arbitrage but this approach do not take into account the relationship between each point in the interpolated smiles and so as an trade-off it has a very high chance to generate uncontrollable butterfly arbitrage opportunities.

By interpolating in the parameters instead this property diminishes but instead there are higher chances of introducing calendar arbitrage. The chance of generating calendar arbitrage in this case are not as big as the chance of generating butterfly arbitrage in the previous approach and so it can be concluded that the parameter interpolation approach is the recommended one. This approach is also much easier to apply.

When applying the parameter interpolation on the SVI family we have two cases, the $xSSVI$ case (the case of using either the SSVI or the eSSVI parametrization) and the *raw-JW* case. In the case of working with an xSSVI fit we have only one parameter θ_τ i.e the total implied variance ATM which the smiles are depended on. Therefore we only need to interpolate in between our θ_τ. This is one of the strengths of the xSSVI fit. What more is that since the xSSVI form is following a nice pattern it almost never introduces initial fits that create tough cases for the interpolation method. This results in that the interpolation in general achieves completely arbitrage free solutions. In fig.(7) we can see an example of how the parameter movement in the xSSVI case can look like when using this strategy.

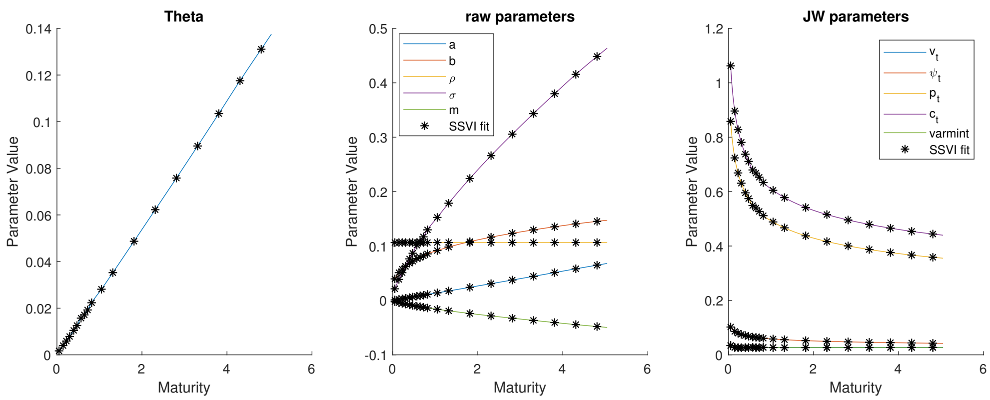

Fig. 7: Example of how parameters look with SSVI parameter interpolation. The data used for this example is on Toppix and the interpolation method used is the *monotonic spline* interpolation.

For the case of interpolation in the raw or JW parameters the initial fit do not have the nice dependence between parameters as the xSSVI parameters have. Instead will every parameter have a semi-independent fit which means that they can behave almost however they want.

We recommend to interpolate in the raw parameters instead of the JW parameters. In this way the interpolation is easier to control. If interpolation is being done on the JW parameters we have experienced strange behaviors that we want to eliminate. In fig.(8) we can see an example of how the parameter movement can look like when interpolating in this case.

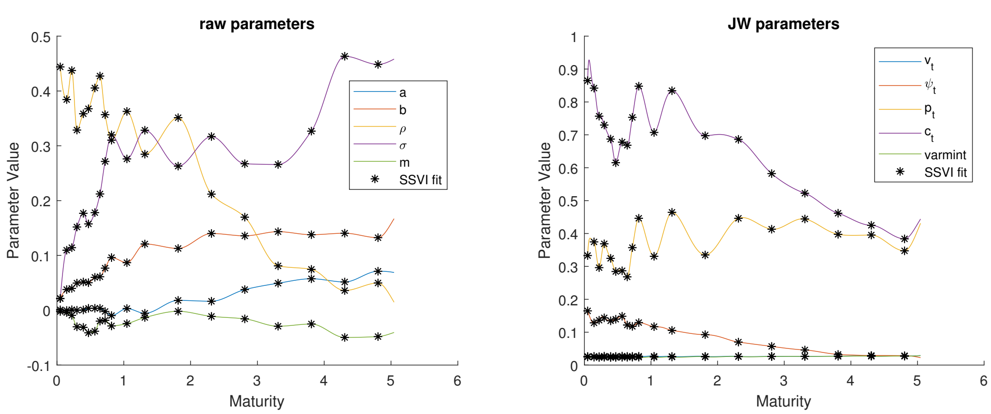

Fig. 8: Example of how parameters look with calibrated SSVI parameter interpolation. The data used for this example is on Toppix and the interpolation method used is the *monotonic spline* interpolation.

Our interpolation method of choice is the *Monotonic Spline* interpolation. This is a interpolation method of the higher order that generates smooth interpolation solutions. When using higher order interpolation methods it becomes clear that keeping the monotonicity in the market data fits is a very important property. Because of this reason the monotonic spline interpolation is preferred over regular spline interpolation. Jim in [43] recommended *Stineman* interpolation which also is an option. Both the Stineman and the monotonic spline interpolation gives similar result and so the choice here is up to the practitioner. In fig.(9) we can see an example of the SVI surface and in fig.(10) we can see an example of a SSVI surface, when we applying the monotonicity spline interpolation.

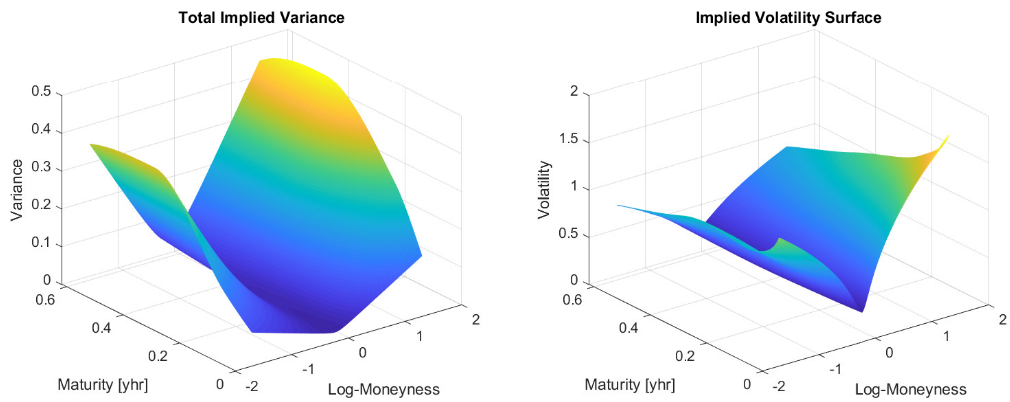

Fig. 9: Example of how a SVI surface can look like when using the monotonic spline interpolation. This is the same surface that was shown in fig.(3).

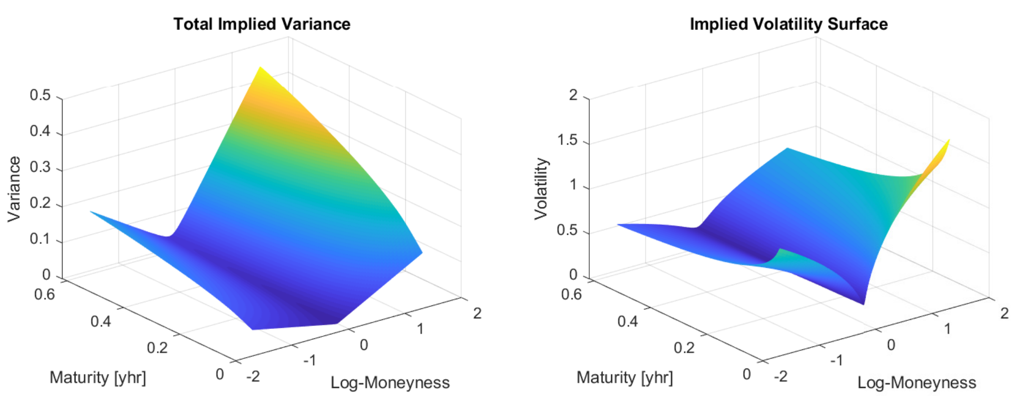

Fig. 10: Example of how a SSVI surface can look like when using the monotonic spline interpolation. This is the same surface that was shown in fig.(5).

As the reader can see from these figures, is the monotonic spline interpolation giving us very smooth surfaces but even though that the results are satisfying they are not perfect.

A very interesting observation we made under our investigation was a phenomenon that we call the *closeness interpolation problem*. When two slices in the market fit is very close to each other will the monotonic spline interpolation not manage to keep the monotonicity. This means that the interpolated slices in that area will with a high possibility generate calendar arbitrage. Example of a case when this is happening is in fig.(11). In that case we can see that a few slices are located very close to each other on the left side. When we apply the monotonic spline interpolation some slices will therefore cross others and so we get small local calendar arbitrage areas.

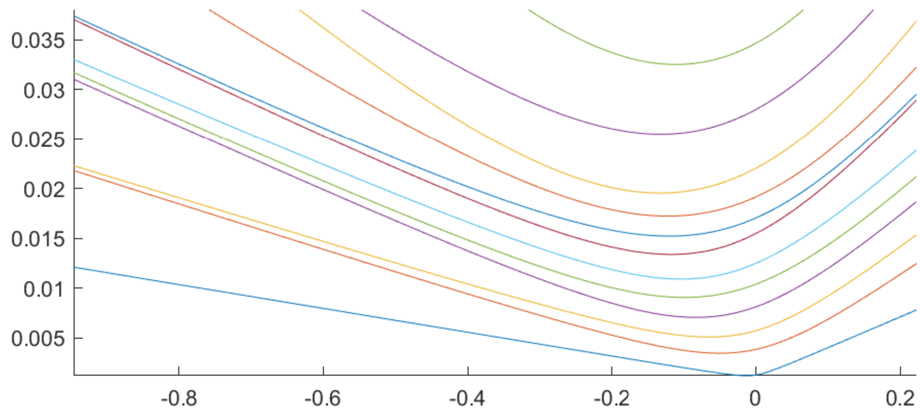

Fig. 11: Example of fits that become very close and makes it tough for the monotonic interpolation to create local calendar arbitrage free solutions.

The same phenomenon occurred in all different interpolation methods we have investigated this far and at this moment we do not have a solution to fix it. The practitioner should have this in mind when using the surfaces.

It is worth mentioning that this problem is only happening since we are searching for a completely arbitrage free solution. If the practitioner want volatility on instruments that are divided with not to small distances this problem will not occur in between those prices.

5.6 Extrapolation

There could be cases when we want volatility outside of the range of the market data. In those cases we need to extrapolate our surfaces. As in the previous section is this an area that is not well documented and so what method seems to be the best choice is hard to say.

There is two cases of extrapolation for the SVI family surfaces, the *short term* extrapolation which is extrapolation towards 0 maturity and the *long term* extrapolation which is extrapolations towards longer maturities. There is of course extrapolation that can be done in the log-moneyness direction but this is already determined with the parametric smile that the different SVI models are defining.

5.6.1 Short Term Extrapolation

From Ropers arbitrage result in theorem 3.2 we know that in the short term extrapolation, the total implied variance is suppose to become 0. This is also reasonable when thinking about how option contracts work. We define this in mathematical terms as for every $x \in \mathbb{R}$, $\Sigma(x, 0) = 0$ where $\Sigma = \sigma_{imp}(x, \tau)\sqrt{\tau}$. Even tough this might be true the theorem is an ideal result of an ideal market that follow the "mathematical rules" the only problem is that the real world does not work like that. It is of course hard to say how the market works in detail but it is not unreasonable to say that most practitioners on the market want to earn or save money and therefore it it also reasonable to assume that when some practitioner puts out a price of a contract or accepts a price of a contract and buys it, he is not thinking about if he is doing this in the framework of arbitrage theory, he just thinks the price is reasonable! What we are left with is therefore a dilemma of basing our extrapolation (our guess) on the mathematical condition or the behavior of the market.

In practical sense this means that for extrapolating in the short term we must choose to either force the extrapolation to go towards 0 or forget about this mathematical property and just try to project the surface depending on the market data.

If we do not want to use the mathematical property and rely on the market data fully then the extrapolation method of our choice have been to try to project a new parameter set using the initially fitted parameters on our market data. The projection method that seem to be working best is using linear projection but there is arguments to use non-linear projection as well.

With short term extrapolated parameters, we can see this smile as part of the market data fit and then get the complete surface, including the extrapolating part, by interpolating as discussed previously.

In our investigation in this area we came across an interesting observation. We observed that depending on the speed that the total implied variance converges towards 0 the implied volatility surface will show one of three behaviors. The most common behavior is that the implied volatility will explode ATM together with its wing and show very high value close to 0 maturity, the second behavior is a constant behavior where the implied volatility ATM will behave almost constant close to 0 maturity and the third behavior is when the implied volatility ATM drops and gives a shape reminding of a beak.

5.6.2 Long Term Extrapolation

Compared to the short term extrapolation is the long term extrapolation less bounded by mathematical conditions. We know from empirical results that the wings of the volatility surface will flatten out with longer maturities. In [1] they cited Rogers and Tehranchi (2009) which showed the following mathematical limit as a description for long maturities,

$$\lim_{\tau \to \infty} \sup_{x_1, x_2 \in [-X, X]} \mid \hat{\sigma}(x_1, \tau) - \hat{\sigma}_(x_2, \tau) \mid = 0. \tag{34}$$

This behavior has already been seen in the surfaces that we have generated this far. I think it is therefore reasonable to try keep that same behavior.

Other systems and institution lean towards using a linear extrapolation approach, this is shown example in [35, 16, 30]. I think this is a reasonable approach but how you determine a reasonable slope is a question mark.

As in the short term extrapolation could a linear trend projection be a more reasonable approach alternatively using a second order trend projection to try catch the flattening in the wings.

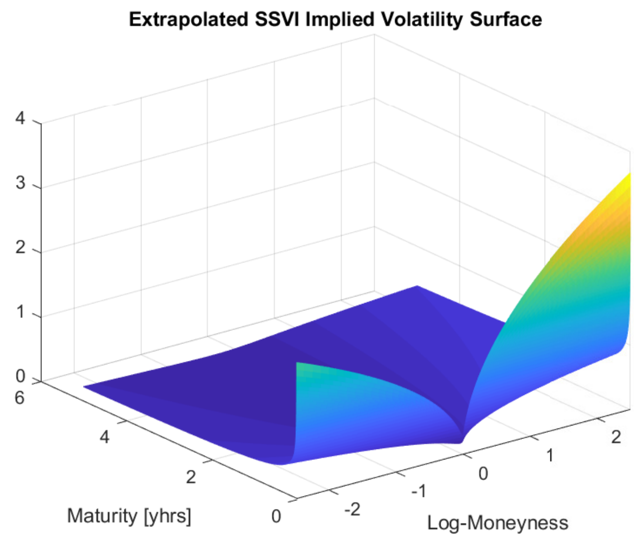

Fig. 12: Example of an extrapolated surface. Notice the explosion that happens in the wings and ATM close to the 0 maturity.

5.7 Calibration

At this point we have gone through all aspects of constructing an implied volatility surface using the SVI parametrization family. We could be satisfied with our results and set up a complete procedure using some of the alternatives that we have investigated but for this section we want to focus on a complementary method that will enable us to both adjust butterfly arbitrage and improve the initial fit. We call it the *calibration procedure.*

The inspiration for this procedure comes from a result that Gatheral presented in [16] regarding the raw SVI parametrization. In that method as we have seen is each slice fitted independently, which makes it highly possible that the slices will either introduce butterfly arbitrage or calendar arbitrage. Gatheral presented a method of calibrating or adjusting the SVI slices in a way that eliminated the butterfly arbitrage. The procedure is based on that we first fit the raw SVI parametrization and then define that smile in both the raw parameter form (Definition 5.1) and the JW parameter form (Definition 5.2). Gatheral then defined the following lemma which describes the basics of the arbitrage elimination procedure.

Lemma 5.3 (Gatherals Butterfly Arbitrage Elimination)**.** *Suppose we choose to fix the SVI-JW parameters v_t, ψ_t and p_t of a given SVI smile, we may guarantee a smile with no butterfly* *arbitrage by choosing the remaining parameters c'_t and $\tilde{v}'_t$ as,*

$$
\begin{aligned}
c_\tau &= p_\tau + 2\psi, \\
\hat{v}_\tau &= v_\tau \frac{4 p_\tau c_\tau}{(p_\tau + c_\tau)^2}.
\end{aligned}
\tag{35}
$$

In other words, given a smile defined in terms of its SVI-JW parameters, we are guranteed to be able to eliminate butterfly arbitrage by changing the call wing c_t and the minimum variance $\tilde{v}_t$. Gatheral would then show this procedure in action and as he states shows how the adjustment give an arbitrage free result. We have recreated his result in figure.(13), where we can see that his repicking strategy in fact eliminates the butterfly arbitrage.

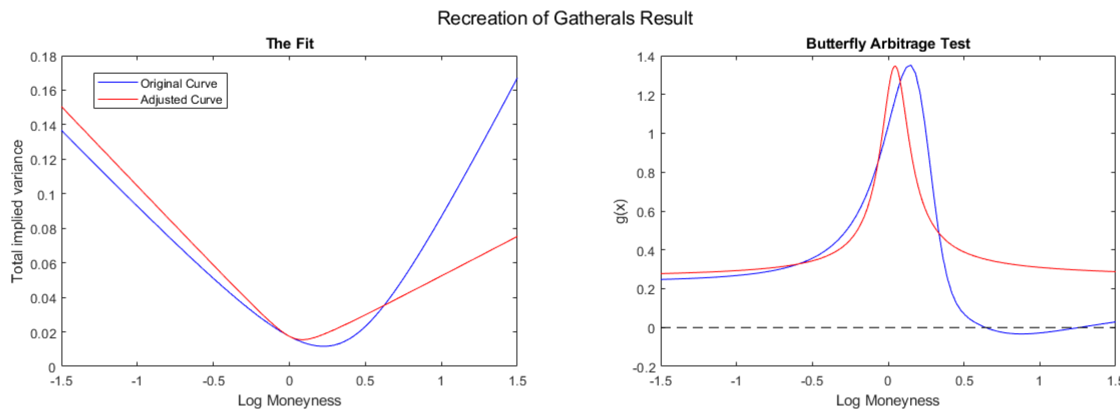

Fig. 13: This is a recreation of Gatherals result in [16] in Example 5.1, Figure 2. The upper plot is on the total implied variance curve that is estimated by the parameter set given by Gatheral and the bottom plot is on the Durrlemans Arbitrage Condition. If $g(x)$ is below 0 there exist arbitrage opportunities there.

We applied this strategy ourselves when investigating the area, but observed that there are cases when this repicking strategy in fact doesn't guarantee an a arbitrage free smile. As seen in fig.(14) even though we apply the repicking strategy the new curve has arbitrage. When inspecting this result what we can see is that the arbitrage, that was located on the left side of $x = 0$, has been eliminated but a new area of arbitrage opportunity has been created to the right instead.

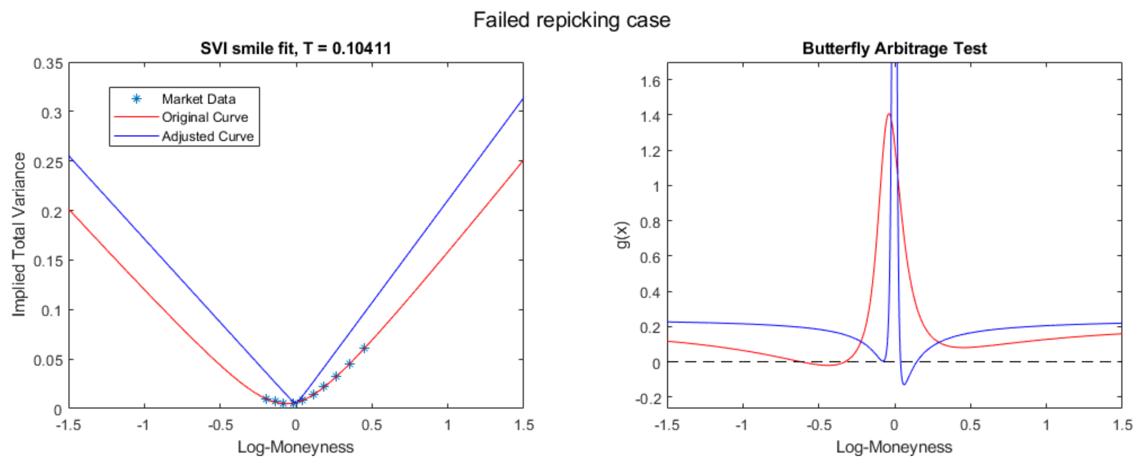

Fig. 14: This is an example for a case when Gatherals repicking strategy doesn't work. Observe that the arbitrage is located to the left of 0.

We did therefore an empirical investigation and observed an interesting property of the JW paramatrization (Definition 5.2). It is unclear if Gatheral was aware of this and just did not explain this further in his paper since the result might seem quite trivial.

For the repicking strategy Gatheral specifically explain that we should fix the values, v_τ, ψ_τ, corresponding to the ATM data. This is reasonable since we know from other studies and practitioners that the data in that area is in general the most liquid area or in other words the best data we have. So by fixing v_τ, ψ_τ there remains 3 parameters that can be calibrated to adjust for the arbitrage. In Appendix A we can see how each parameter affects the arbitrage curve $g(x)$. What we can conclude from inspecting those results, is that:

> *Lowering* p_τ, raises the left wing of the arbitrage curve, (left of the 0 point). It seems like it has a small affect on the right side as well. There were cases when by eliminating the left wings arbitrage we created arbitrage opportunities on the right side.
>
> *Lowering* c_τ, raises the right wing of the arbitrage curve. It seems to have a smaller affect on the left wing then p_τ had on the right.
>
> *Lowering or raising* $\tilde{v}_\tau$, can raise the whole arbitrage curve. Its upper limit is intuitively v_τ.

With these observations we would like to propose a loser condition for finding the guaranteed free of butterfly arbitrage curve.

Definition 5.5 (Repicking Strategy)**.** *With a defined JW parametrization for a given case, where we know that butterfly arbitrage exist, we can find an adjusted JW parametrization without arbitrage by fixing the ATM values,* v_τ, ψ_τ *and then adjusting* $p_\tau, c_\tau, \tilde{v}_\tau$.

Depending on the arbitrage location we know what parameters will have the best elimination effect, this means that we can define the adjustment methodology.

Definition 5.6 (Adjustment Methodology)**.** *In regards of definition 5.5, if the arbitrage is located on the left side of the ATM values,* v_τ, ψ_τ*, the adjustment should mainly be done on* $p_\tau, \tilde{v}_\tau$ *and if it is located on the right side of the ATM values, the adjustment should mainly be done on* $c_\tau, \tilde{v}_\tau$*. Note that all three parameters can be adjusted but to minimize computational power two variables is usually enough.*

Apart from enabling us to eliminate butterfly arbitrage by adjusting the smile, this strategy of "calibrating" our fit by changing the corresponding JW parameters also opens up possibility to improve fits.

Assume that we have an initial fit ω. We can transform that fit into JW parameters $(v_\tau, \psi_\tau, p_\tau, c_\tau, \hat{v}_\tau)$. We assume that the ATM fit is good, this is usually the case if we use the SSVI parametrization or fitting strategies that put a lot of weight in that area, and then we fix v_τ, ψ_τ. Our goal is now to adjust $p_\tau, c_\tau, \hat{v}_\tau$ in a way that keeps the fit butterfly arbitrage free but improve the fit. We can therefore set up the optimization problem,

$$\begin{aligned} \min_{c_\tau, p_\tau, \hat{v}_\tau} \quad & \sum_{i=1}^{n} w_i \left(\omega_i(c_\tau, p_\tau, \hat{v}_\tau) - \hat{\omega}_i\right)^2, \\ \text{such that} \quad & g(x) \leq 0 \end{aligned} \tag{36}$$

where $\omega_i(c_\tau, p_\tau, \hat{v}_\tau)$ is the JW-calibrated parametrization transformed into the raw parametrization, $\hat{\omega}_i$ is the given market data defined in total implied variance according to eq.(41), w_i is weights for defining the goodness of different data points, n is number of data points and $g(x)$ is the Durrleman condition in eq.(9) which we also stated in Definition 3.1 and 3.3.

To make the calibration even more robust we can add the condition of not introducing any calendar arbitrage. For a predetermined interval for log-moneyness we state the weak (local) condition as follows. If $\tau_1 < \tau_2 < \tau_3$ then there is no calendar arbitrage if $\omega_1 < \omega_2 < \omega_3$ where ω_2 is the smile that we are calibrating. If we want the complete arbitrage free solution then we add the condition of for $x \to -\infty$, $\frac{d\omega_1}{d\tau} \leq \frac{d\omega_2}{d\tau} \leq \frac{d\omega_3}{d\tau}$ and for $x \to \infty$, $\frac{d\omega_1}{d\tau} \geq \frac{d\omega_2}{d\tau} \geq \frac{d\omega_3}{d\tau}$. In this way the different smiles will never cross.

Note that if we add the calendar arbitrage condition the calibration will become more biased to the initial fit, which means if the initial fit is bad, the calibration might not converge at all! To handle cases like this a more rough adjustment strategy is needed (more on this later).

Our recommendation is to use the weaker calendar condition for the calibration, but use a quite large interval for log-moneyness, x. An initial interval if the reader does not have any experience in the field would be to use $(-1.5, 1.5)$, it can be lowered or raised if needed but that is depended on what type of instrument we are looking at i.e how far from ATM the instrument is being bought?

With a given data set of a number of maturities, we can apply the calibration procedure in a few different ways. We can calibrate each fit independently starting from smallest maturity and stepping forward through each slice, we call this *forward calibration.* We can also do the reverse and start with the longest maturity and stepping backwards through each slice, we call this strategy *backward calibration.* Lastly we can do a global calibration and do all slices at the same time, a so called *global calibration*. For the global calibration the optimization problem get changed into,

$$\begin{aligned} \min_{c_\tau, p_\tau, \hat{v}_\tau} \quad & \sum_{s=1}^{S} \sum_{i=1}^{n} w_{s,i} \left(\omega_{s,i}(c_{s,\tau}, p_{s,\tau}, \hat{v}_{s,\tau}) - \hat{\omega}_{s,i} \right)^2, \\ \text{such that} \quad & g(x) \leq 0 \end{aligned} \tag{37}$$

where S is the number of maturities. Note that each slice has its own set of parameters $(c_s, \tau, p_s, \tau, \hat{v}_s, \tau)$.

To investigate what strategy seems to be the best we made an experiment in Appendix A. The following diagram shows the overall result to get a good indication of what procedure performed the best.

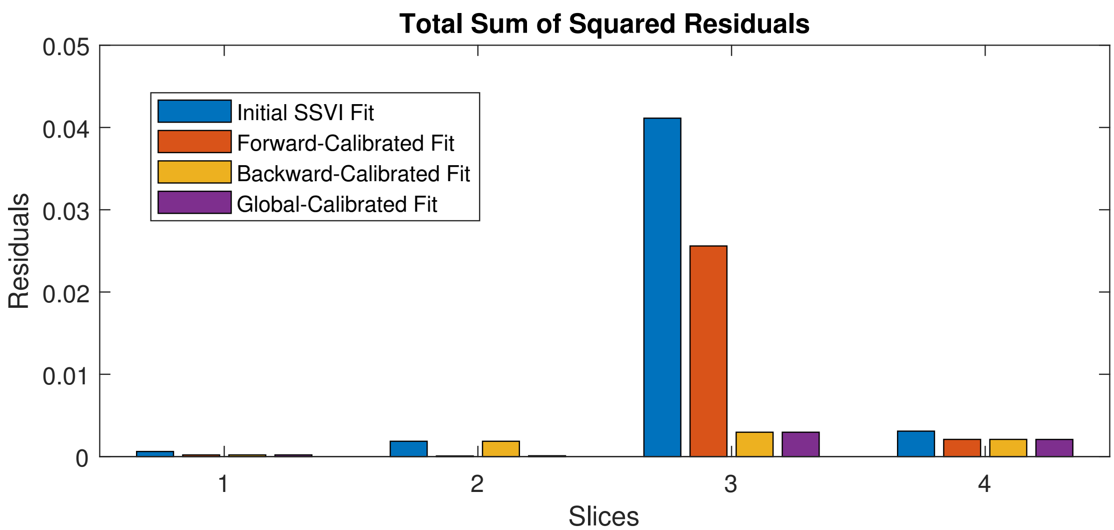

Fig. 15: Comparing the TSSR for all calibration strategies on the same slices on the same market data.

5.8 Adjustment Procedure

In regards to the calibration method presented in the previous section, we mentioned that if calendar arbitrage conditions is added to the optimization problems, the fitting get a higher bias towards the initial fit. This can then create problems when the initial fit is not giving a good fit. The result is that the calibration method wont improve the initial fit specially much.

The problem is coming from that we fix v_t and ψ_t, and only calibrate the reaming parameters c_t, p_t and $\tilde{v}_t$. This means that we fix the ATM behavior and only allow to adjust the outer behavior in the wings. If the initial fit is not good enough this will mean that we fix a behavior that is bad!

For cases like this we need to apply an adjustment procedure that will allow to re-fit each smile independently but still not introduce arbitrage. The goal is to shake up the fit so that if we are lucky, the optimization will converge to a good market fit. The following flowchart is the method we have used for adjusting:

1. We have an initial fit that we have gotten from fitting the SSVI parametrization and then calibrated to improve the fit. The fit gives still relative high errors and so we suspect that we have a bias problem.

2. We initiate an adjustment procedure,

   a. *Backward-Adjustment:* Start from largest maturity and go backwards until the first maturity. At each maturity, T_n try to re-fit that smile with the boundaries of the smile before T_{n-1} and after T_{n+1} and also the arbitrage conditions. If we find some fit that is in the set of conditions and are an improvement to the current smile then we swap them and move on to next maturity (if not an improvement do not swap and move on). When the whole surface fit is done go to step b.

   b. *Forward-Adjustment*: Start from the smallest maturity and go forward until the last maturity. As in step a. try to re-fit the smile, and if the smile is an improvement swap to that one as long as it is in the set of conditions. If the re-fit is not an improvement or that there is no solution at this point do not swap. After all smiles are done move on to step c.

   c. If there is any improvement from step a. and b. meaning that we have swapped some fits, re-start the adjustment procedure from step a., if there is no improvements then the adjustment procedure is complete.

This adjustment procedure is improving the overall method that we have shown this far and is even improving most calibration cases that is also good! Using an independent adjustment procedure could therefore not only be a way to handle bias problems but also to improve the fit that we are getting.

In fig.(16) we can see an example of when we have applied the calibration and the adjustment procedure on an SSVI surface. In fig.(17) we can then see the the difference in fit against the each market fit slice.

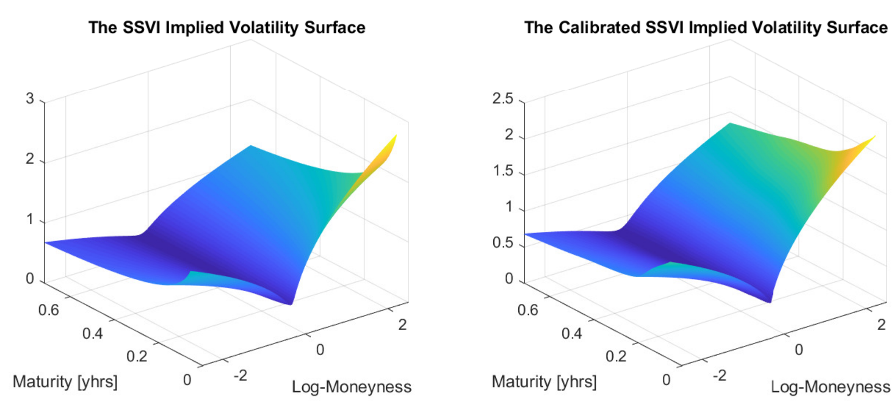

Fig. 16: Example of the calibrated SSVI compared to the inital SSVI surface.

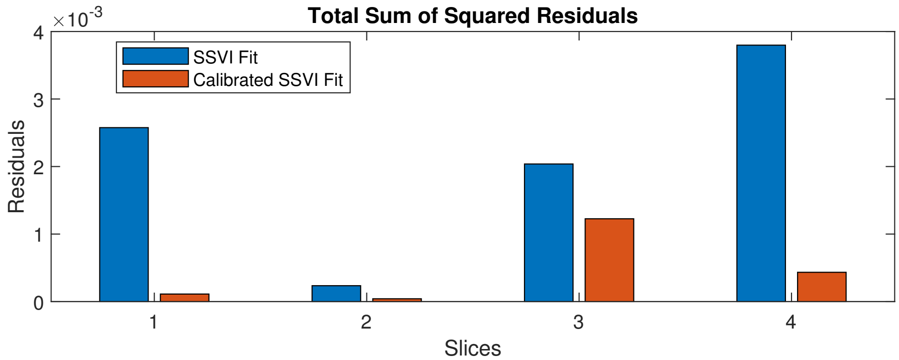

Fig. 17: Complementary result to fig.(16) describing the difference in fit against the market data.

5.9 Flowchart

We will in the following section give a complete summary of the procedure to generate an implied volatility surface based on using the SSVI model and the complementary calibration and adjustment procedure. We call the following method of combining the SSVI and the calibration procedure the *calibrated SSVI method*.

5.9.1 The xSSVI method

We assume that we have transformed our price data on options into market implied volatility data. We also assume that we have the corresponding log-moneyness and maturity for that price data.

This data is then transformed into total implied variance according to eq.(41). Then the procedure goes as follows,

1. Pick out from the data or approximate the ATM θ (total implied variance at $x = 0$). This can be done in different ways, a recommended way is to fit the SVI parameterization of one maturity explained in section 5.3 and then pick the ATM point $x = 0$.

2. Fit the xSSVI paramterization for the whole data set by solving the optimization problem in eq.(31).

3. Test if there is any butterfly arbitrage in the fit by applying the arbitrage test in Definition 3.3.

    a. If there exist butterfly arbitrage; try to adjust the smiles by applying the adjustment methodology in Definition 5.6. If the butterfly arbitrage is eliminated successfully move to step 3 otherwise go directly to 4a and re-fit the smile.

    b. If there is no butterfly arbitrage; continue to step 4.

4. Test if there is any calendar arbitrage in the fit by applying the arbitrage test in Definition 3.4.

a. If there is calendar arbitrage; re-fit the smile, that has arbitrage, independently. Then go back to step 3.

b. If there is no calendar arbitrage; continue to step 5.

5. At this point we have an arbitrage free set of slices. Either we calibrate this fit and get the calibrated SSVI fit or we continue with the SSVI fit and construct the surface. For the calibrated SSVI fit see next subsection and for the SSVI fit continue to step 5.

6. Short term extrapolate the initial fit by applying either the behavior keeping extrapolation or condition keeping extrapolation. For the case of condition keeping add the constructed data point $\theta(\tau = 0) = 0$ and in the case of behavior keeping use some linear trend projection on θ. Make sure that the short term extrapolated slice is not introducing arbitrage.

7. Long term extrapolate θ by applying linear trend projection. Make sure that the long term extrapolated slice is not introducing arbitrage.

8. Add the short term and long term extrapolated parameter sets corresponding to two new slices, to the initial market data fit. This means that we added two more slices.

9. Construct the interpolated surface by interpolating θ using monotonic spline.

10. Get the implied volatility surface by transforming the interpolated total implied variance surface into implied volatlity using eq.(41).

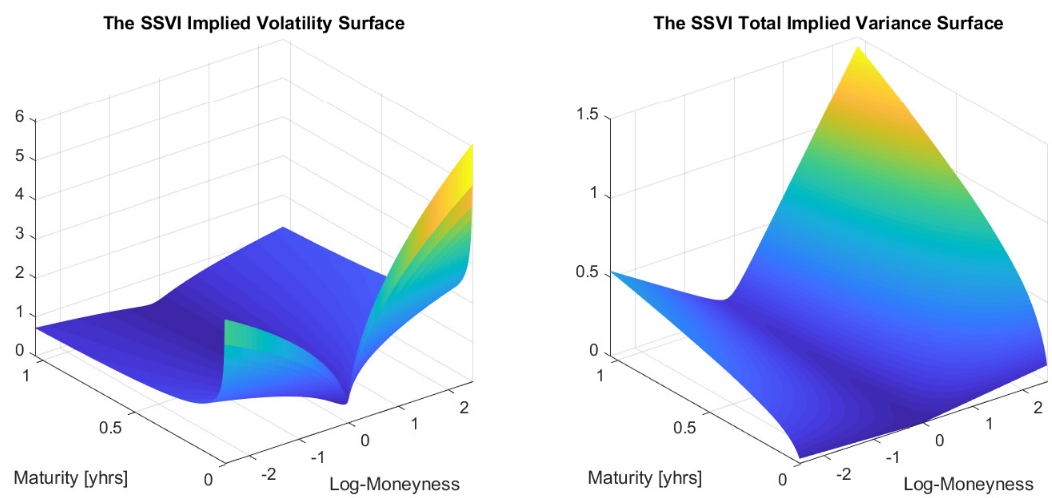

Fig. 18: Example of a surface generated by the xSSVI method.

5.9.2 The Calibrated SSVI method

It is assumed that we have followed the previous flowchart and have fitted the SSVI fit onto the whole data set i.e we are at step 5.

6. Transform the SSVI fit into raw and JW parameters using Lemma 5.2 and 5.1.
7. Calibrate the SSVI fit by fixing v_τ, ψ_τ for all slices and then solving the optimization problem eq.(37), use backward calibration or global calibration.
8. Test for arbitrage in the calibrated fit and adjust if it exist following step 3-4. When arriving at an arbitrage free calibrated SSVI fit move to step 9.
9. To minimize bias and improve the fit even more, apply the adjustment procedure in section 5.8.
10. Short term extrapolate the initial calibrated fit by applying either the behavior keeping extrapolation or condition keeping extrapolation. Compared to the SSVI fit is this now done on all raw parameters. Make sure that the short term extrapolated slice is not introducing arbitrage.
11. Long term extrapolate the raw parameters by applying linear trend projection. Make sure that the long term extrapolated slice is not introducing arbitrage.
12. Add the short term and long term extrapolated parameter sets corresponding to two new slices, to the initial market data fit. This means that we added two more slices.
13. Construct the interpolated surface by interpolating our parameters using monotonic spline. Be aware of the *closeness problem* that was discussed in section 5.5.
14. Get the implied volatility surface by transforming the interpolated total implied variance surface into implied volatility using eq.(41).

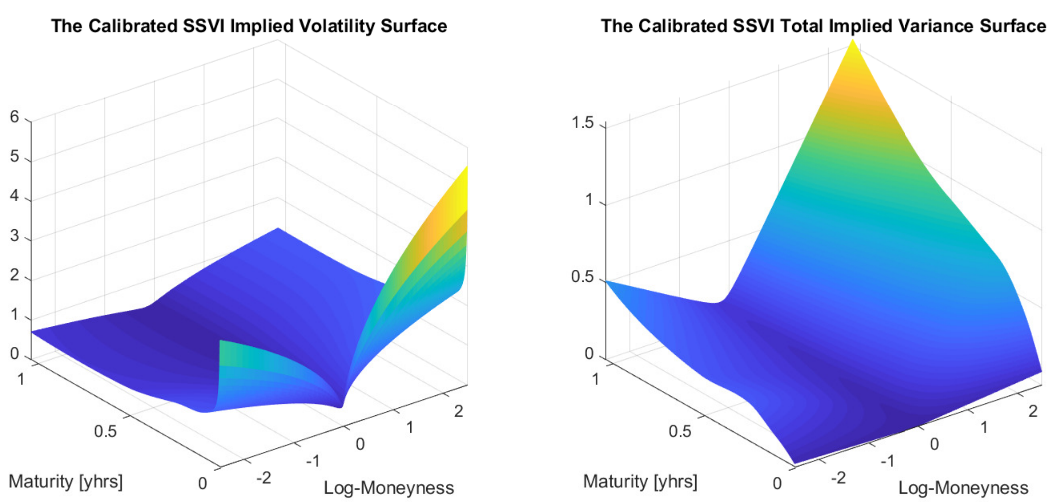

Fig. 19: Example of the same surface as in fig.(18) but generated by the calibrated SSVI method.

6 Performance Analysis

The purpose with this thesis was to present a method that would model the implied volatility surface for an option. The goal was to do this in a reasonable and robust way. That meant that we wanted our surfaces to generate arbitrage free solutions which catches the behavior of the market by given good market fits. This far we have taken fourth a method that hopefully satisfies these goals. In this section we will test how well the concluded method, summarized in section 5.9, is performing in regards to these goals.

First we will test how robust the methods are, this means that we will test how often the method will generate solutions that introduces static arbitrage.

The second test will be in the view of quality. To test the quality we will look at the residuals against the market data and compare each method.

6.1 Data

For the upcoming tests will we use market data from 5 different assets. First we have 3 different stocks with the ID 1330, 1343 and 6770 and then we have 2 indices of Nikkei and Toppix. Each asset data have a total of 122 days that we will loop through and generate a surface per day. There is some data sets mainly in the indices that are very large compared to the stocks, and have not been cleaned. That the data set has not been cleaned means that the data with a high possibility have bad data in them and so they will have a lot of initial arbitrage in the data.

6.2 Robustness Test

For the robustness test, we will look at how many cases of the iterated data sets introduces static arbitrage. We will also see in what area of the surface the arbitrage is created. The different areas is the initial fit which is the fit against the given market data, the extrapolation which is including the whole extrapolated area outside of the range of the market data and the interpolation which is the interpolated part inside of the market data range.

In tab.(1) we have the overall result from the robustness test, in tab.(2) we have complementary result which shows what type of arbitrage is created and in tab.(3) we can see what type of extrapolation generates the most arbitrage.

Underlying	SSVI			calib SSVI		
	Static Arbitrage			**Static Arbitrage**		
	M.fit	Extrap.	Interp.	M.fit	Extrap.	Interp.
1330		**0**			**118**	
	0	0	0	0	53	116
1343		**0**			**118**	
	0	0	0	0	59	111
6770		**0**			**120**	
	0	0	0	0	115	92
Nikkei		**105**			**122**	
	100	28	100	98	98	122
Toppix		**5**			**122**	
	1	5	1	8	122	115

Table 1: Summary of number of data set (days) where the SSVI and calibrated SSVI has arbitrage in its solution. *M.fit* stands for market fit and gives the number of cases when the arbitrage is located in the fit against the market data. *Extrap.* stands for extrapolation and present number of cases when there exist arbitrage in the extrapolated areas. *Interp.* stands for interpolation and gives the number of cases when there exist arbitrage in the interpolation between market fitted smiles and the 2 extrapolated smiles.

	SSVI		**calib SSVI**	
Underlying	Butt Arb.	Cal Arb.	Butt Arb.	Cal Arb.
1330	0	0	6	118
1343	0	0	1	118
6770	0	0	36	120
Nikkei	12	102	36	122
Toppix	5	1	54	122

Table 2: Complementary result to table 1. *Butt Arb.* stands for butterfly arbitrage and show how many cases there where butterfly arbitrage introduced. *Cal Arb.* stands for calendar arbitrage and show how many cases there where calendar arbitrage introduced.

	SSVI		**calib SSVI**	
Underlying	Short Exp.	Long Exp.	Short Exp.	Long Exp.
1330	0	0	42	26
1343	0	0	42	29
6770	0	0	75	67
Nikkei	18	10	83	34
Toppix	5	0	114	14

Table 3: Complementary result to table 1 which shows how the arbitrage introduced in the extrapolation is divided between the short term extrapolation and the long term extrapolation.

As stated in section 6.1 is the data set Nikkei not cleaned. It is evident that this is affecting the result of the methods. This is the only case when the SSVI method do not manage to generate generally arbitrage free solutions! This is very clear when inspecting table 1.

It is also clear that the interpolation and extrapolation is not able to generate consistently arbitrage free solutions.

The type of arbitrage that is mostly present in the surfaces is the calendar arbitrage, even though it is not present in the inital fit is it created in either the interpolation or the extrapolation.

Both the SSVI method and the calibrated SSVI method are generating consistently arbitrage free market data fits.

6.3 Quality Test

To test the quality of the methods, we will compare the goodness of the fit for the surfaces against our market data. The goodness of the fit will be defined using the absolute value of the residuals. We will compare the SSVI fit and the calibrated SSVI fit.

In tab.(4) we have the overall result from the quality test and in tab.(5) we can see how the error is located between different maturity areas.

To get a nice feel of how the method compare to each other, we have also plotted the development of the mean error for both methods in the same plots. These plots can be found in Appendix A.

	SSVI			**calib SSVI**		
Underlying	Tot.err	Mean.err	S.mean.err	Tot.err	Mean.err	S.mean.err
1330	0.7547	0.006186	0.001563	0.25102 (0.66739)	0.0020576 (0.66739)	0.00052274 (0.66556)
1343	0.40988	0.0033596	0.00084954	0.25224 (0.3846)	0.0020675 (0.3846)	0.0005258 (0.38107)
6770	8.3258	0.068244	0.017314	3.3751 (0.59462)	0.027665 (0.59462)	0.0069905 (0.59625)
Nikkei	1018.7617	8.3505	0.34983	808.7671 (0.20613)	6.6292 (0.20613)	0.27706 (0.208)
Toppix	159.3917	1.3065	0.068763	81.6849 (0.48752)	0.66955 (0.48752)	0.035239 (0.48752)

Table 4: Summary of quantified goodness of fit. *Tot.err* stands for total error and give the total error for all 122 data sets for each underlying. *Mean.err* stands for mean error and give the mean error for each data set. *S.mean.err* stands for smile mean error and give the mean error for every smile in the data.

	SSVI			**calib SSVI**		
Underlying	Short	Medium	Long	Short	Medium	Long
1330	0.0016019	0.0014746	NaN	0.0004196 (0.73805)	0.0010391 (0.29533)	NaN (NaN)
1343	0.00090178	0.00060227	NaN	0.0005569 (0.38245)	0.0004247 (0.29483)	NaN (NaN)
6770	0.019688	0.0078246	NaN	0.0080672 (0.59024)	0.0029783 (0.61937)	NaN (NaN)
Nikkei	0.24118	0.59173	0.26733	0.21889 (0.092432)	0.47475 (0.19768)	0.16243 (0.39241)
Toppix	0.080205	0.081476	0.042989	0.024673 (0.69238)	0.058853 (0.27767)	0.018 (0.58128)

Table 5: Complementary result to table 4. Result is showing in what interval of maturity the smile mean error i located. *Short* is defined as the maturity interval $(0, 0.5)$, *Medium* is defined as the maturity interval $(0.5, 2)$ and *Long* is defined as the maturity interval $(2, \infty)$.

It is evident that both the SSVI and the calibrated SSVI method is generating very good surfaces for stocks. For the indices is are both methods not performing as well. The Nikkei case as we know is not a singular case since the data is not cleaned and it is clear that the arbitrage in the data is affecting the results. For the Toppix case are both method performing better but it can still not be considered good.

It is although clear that in all cases is the calibrated SSVI generating much better then the SSVI method. In general the calibrated SSVI method is generating solutions that is approximately 40-60% better then the SSVI method.

7 Discussion

Our mission for this thesis was to investigate how to construct implied volatility surfaces. We wanted our surfaces to be arbitrage free and robust. This meant that our model of choice should be able to generate arbitrage free solutions for the majority of the data sets that we use it for.

In the performance analysis made in Section 6, we can see that the SSVI method is a very robust model. Pretty much on all data sets was the SSVI able to generate completely arbitrage free surfaces. Because of how the parameters in the SSVI is following a predetermined function is also the interpolation and extrapolation keeping the arbitrage free state that the initial fit has.

Compared to the calibrated SSVI is the SSVI outperforming in this regard. The calibrated SSVI parameters is not following some predetermined function, as in the SSVI model, and so to try to find some trend in the parameters for extrapolating is hard and therefore we see a lot of arbitrage introduced in the extrapolated parts of the calibrated SSVI surface.

The interpolation has also a higher chance of generating arbitrage in the calibrated SSVI method compared to the SSVI method. This is because the calibrated SSVI has the property to find local behaviors which the SSVI does not. This means that there can be cases when the initial fit have slices that are very close to each other. We saw this phenomenon in Section 5.5 and called it the *closeness* problem.

We can conclude that the interpolation method of choice, the *Monotonic Spline* interpolation, even though it is the best choice at the moment, is not a good enough choice to generate completely arbitrage free solution for the calibrated SSVI method. For the SSVI case on the other hand is it performing excellent!

Because of less robust interpolation and extrapolation is the calibrated SSVI at this moment not as robust as the SSVI but where the calibrated SSVI outshines the SSVI is in the quality of the fit. From tab.(4) we can see that the calibrated SSVI performs between 40 to 60 % better then the SSVI. This becomes even more clear when observing fig.(30)-fig.(39).

In our view is the most important property for the surface method, to be able to give good fits against the market data. This part is the calibrated SSVI method doing very well.

What also is interesting with the calibrated SSVI method is that it improves the initial fit so much that it is closing in to the SVI model which as we know from Section 5.3 gives very good fits but is not a good enough model for generating implied volatility surfaces since it generates static arbitrage in most of its market fits. The calibrated SSVI is in general not introducing static arbitrage in the initial fit. In fact in regards to only the initial fit is the calibrated SSVI on the same level as the SSVI!

From the performance analysis it became clear that both the SSVI method and the calibrated SSVI method struggles with larger data sets, as in the case of indices. For both methods was the fit on Nikkei and Toppix quite underwhelming. It seems that the initial fit with the SSVI is not really finding the close fit as it does in the case of the stock options. Reason for that is hard to point out but there is a chance that since the index data sets is much larger then the stock option data sets, is the optimization procedure requiring much longer computational times and better stopping criteria. It can be the case that the optimization algorithm and stopping criteria used was not enough.

It can be concluded that with the optimization procedure used for this thesis is both the SSVI and the calibrated SSVI method performing much better on smaller data sets. This property could possibly be used to improve the fitting procedure for the larger data sets of the indices. If we could divide the larger data sets into multiple smaller data sets, similar to the stock options, then by fitting each divided data set independently and then combining them into the complete surface, there might be a possibility that the fit will be improved. In this way should the bias problem, in theory, be less of a problem as well. This suggested improvement is of course still just speculation and up for upcoming investigations.

The underwhelming fits on the Toppix and Nikkei case could also stem from the bad data quality. The data sets of the indices that we used where in many cases introducing calendar arbitrage in the initial data. For the 2 methods it is clear that cleaning the data before applying the modelling part is very important to make sure that the methods perform optimally.

A condition that could be used for the cleaning the data is to not allow the data to introduce calendar arbitrage between different maturities. If the data is introducing calendar arbitrage will the model have a very hard time to adapt and so this arbitrage opportunity will with high possibility remain in the modelled surface.

How to clean the data would be to delete the data points that is introducing the calendar arbitrage. When doing so it could be useful to have more information about the data, for example traded volume, to make it easier to justify what data points, in an arbitrage case, should be deleted.

Apart from finding a method that could construct the implied volatility surface in a robust and arbitrage free way where our aim also to find a solution that could be considered simple to apply.

The choice of investigating the parametric representation models was a good choice. As discussed in Section 4.4 is this the class of models that still seems to have the biggest potential.

From finishing the investigation around the SVI parametrization family, we can conclude that the SSVI model is highly recommended if you want to generate complete surfaces. It might be the model that gives the best trade-off between simplicity and good results.

If you do not want to generate complete surfaces and are fine with a discrete number of smiles then the complementary method of using the calibrated SSVI method is a very good choice.

7.1 Future Work

Potential investigations following the results from this report is:

- Investigate how to interpolate and extrapolate the calbrated SSVI method in a robust way, meaning that the method should not introduce arbitrage even though the *closeness* problem would occur in the initial fit.

- Compare the quality between the SSVI method, the calibrated method, the Heston model, the Bates (SVJ) model and the penalized spline method to get a better indication on how well the SSVI and the calibrated SSVI is performing.

- Investigate how to improve the fitting procedure for the SSVI method and the calibrated SSVI method on larger data sets. An idea is to incorporate a tuning procedure where we would fit smaller parts of the whole data set independently and then put them together. In a way you could see it as if the whole surface would be divided into smaller surfaces.

8 Appendix A

8.1 Different forms of Implied Volatility

Log forward-moneyness is defined as,

$$x = log\left(\frac{K}{F_{[t,t+\tau]}}\right). \tag{38}$$

The **total implied volatility** is defined as,

$$\theta_{imp} = \sqrt{\tau}\sigma_{imp}. \tag{39}$$

where τ is the time to maturity. The **implied variance** is defined as,

$$v_{imp} = \sigma_{imp}^2. \tag{40}$$

And lastly the **total implied variance** is defined as,

$$\omega_{imp} = \tau\sigma_{imp}^2. \tag{41}$$

8.2 The Butterfly Arbitrage Test Proof

There is many different investment strategies for exploiting butterfly arbitrage. One example is the *butterfly spread* investment strategy.

Definition 8.1 (Butterfly Spread)**.** *A butterfly spread is an investment when you go long on two calls with two different strike prices,* K_1, K_3 *and short in two calls with the strike price,* K_2*, in between the two previous calls. All four calls have the same time to maturity,* T*.*

The cost of a butterfly spread is depending on how the three calls are priced, i.e according to,

$$price = C_T(K_1) - 2C_T(K_2) + C_T(K_3), \tag{42}$$

where

$$K_2 = \frac{K_3 + K_1}{2}.$$

For a butterfly spread the biggest loss you can do is the initial price of the investment. This means, according to eq.(42), that if the price is negative or equal to 0 we will never lose money. In that case we have found an opportunity to abuse the arbitrage that exist in the call option price relationship.

Now according to the butterfly arbitrage test defined in Definition 3.3 we know that if $g(x) < 0$ those points are introducing butterfly arbitrage and means that the butterfly spread investment strategy should be able to get a price which is negative or equal to 0.

In fig.(20) we have an example of an implied volatility smile that is introducing butterfly arbitrage in between x = [-0.948 -0.378]. By using points $x_1 = -0.948$ and $x_2 = -0.378$ which corresponds to $K_1 \approx 1226.5$ and $K_3 \approx 2168.8$ we get the price -2.1446. This means that we have found an arbitrage opportunity. By adjusting the implied volatility smile using the calibration procedure presented in Section 5.7 we get the green line which should according to Definition 3.3 not introduce butterfly arbitrage. If we look at the same points our investment is now priced 1.8324 meaning that the butterfly arbitrage is eliminated and the arbitrage test is correct.

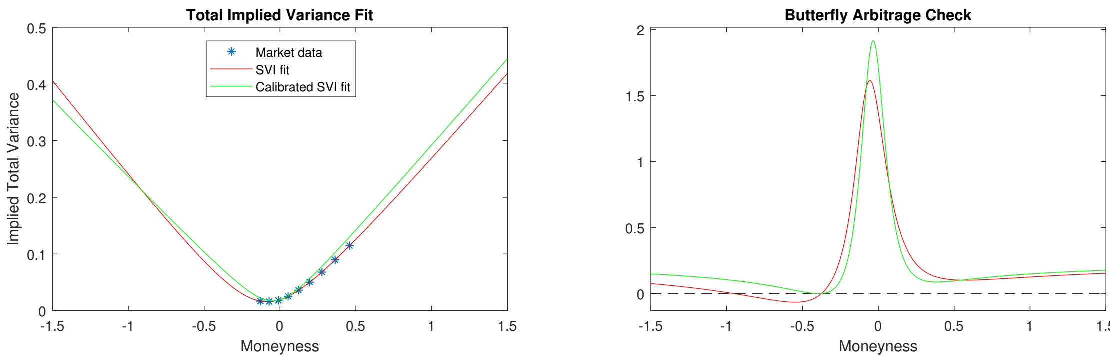

Fig. 20: Example of usage of the butterfly arbitrage test that was presented in Definition 3.3. Red line demonstrates a case of butterfly arbitrage. Green line corresponds to the adjusted case when the arbitrage is eliminated.

8.3 The Calendar Arbitrage Test Proof

Assume that the price for two call option on the same strike price K with different maturities is defined as,

$$C(K, T_1) > C(K, T_2) \tag{43}$$

where $T_1 < T_2$. As an direct link from the Definition 3.2 to the Black-Scholes-Merton price formula, this relation indicates that there exist calendar arbitrage in between the options. One way to take advantage of the arbitrage is to short $C(T_1)$ and long $C(T_2)$. The initial profit is then $x = C(T_1) - C(T_2) > 0$ and the investment strategy has three outcomes. At T_1, if $S \leq K$ then the shorted call option $C(T_1)$ will not be exercised, this means that our possible profit at T_2 is $x + (S - K)^+$. If $S > K$ at T_1 then shorted call option will be exercised, meaning we need to provide the promised stock S. This is done by short sell the promised stock giving us an relative loss om $S - K$ but an absolute profit of K. At T_2 we now have two different outcomes, if $S > K$ we will exercise our call option and buy the stock S for the strike price K which we then can return leaving us with a total profit of x. If instead $S \leq K$ then we do not exercise the option and buy instead the same stock S for a lower price then K. We return the stock and end up with the profit of $x + (K - S_{T_2})$.

Now for this arbitrage example to be possible the call options must be wrongly priced as in eq.(43). It follows that the relationship only happens if as stated in Definition 3.2, the implied volatility is sorted wrongly. This can then be tested as stated in Definition 3.4.

In fig.(21) we have an example of two implied volatility smiles for two different maturities with parameters $S = 3165$, $r = 0$ and $x = -0.6$. For the un-adjusted case $C(T_1) \approx 1429.3$ and $C(T_2) \approx 1428.7$ and for the adjusted case we have $C(T_1) \approx 1428.2$ and $C(T_2) \approx 1428.7$. This means that in the un-adjusted case we get the initial profit of $x = C(T_1) - C(T_2) = 0.6$ and in the adjusted case we get the initial loss $x = C(T_1) - C(T_2) = -0.5$.

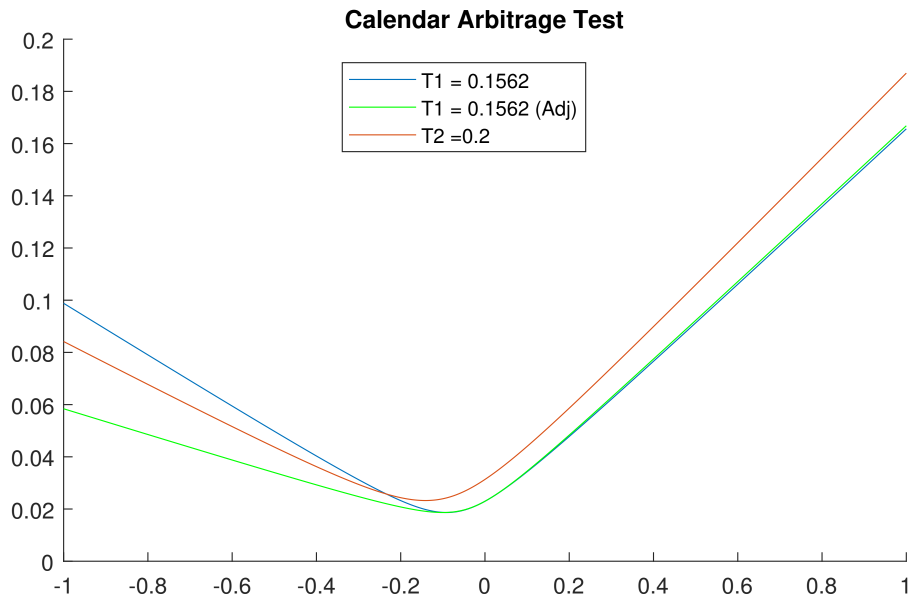

Fig. 21: Example of usage of the calendar arbitrage test that was presented in Definition 3.4. Blue line demonstrates a case of calendar arbitrage. Green line corresponds to the adjusted case when the arbitrage is eliminated.

8.4 Multi Linear Regression Solution Proof

A multiple linear regression model is defined as,

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_k x_k + \epsilon, \tag{44}$$

where $\beta_0, ...\beta_k$ is coefficients, $x_1, ..., x_k$ the models regressors (variables), y the responds and ϵ the random error.

To find the coefficients to eq.(44) we use *Least-Squares*. We assume that we have $n > k$ observations. We also assume that ϵ_i is uncorrelated and have $E[\epsilon] = 0$ and $Var(\epsilon) = \sigma^2$.

We may write eq.(44) in matrix form as,

$$\mathbf{y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon} \tag{45}$$

where

$$\mathbf{y} = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}, \quad \mathbf{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \dots & x_{1k} \\ 1 & x_{21} & x_{22} & \dots & x_{2k} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n1} & x_{n2} & \dots & x_{nk} \end{bmatrix}, \quad \boldsymbol{\beta} = \begin{bmatrix} \beta_0 \\ \beta_1 \\ \vdots \\ \beta_k \end{bmatrix}, \quad \boldsymbol{\epsilon} = \begin{bmatrix} \epsilon_1 \\ \epsilon_2 \\ \vdots \\ \epsilon_n \end{bmatrix}. \tag{46}$$

The least-square function can then be defined as,

$$S(\boldsymbol{\beta}) = \sum_{i=1}^{n} \epsilon_i^2 = \boldsymbol{\epsilon'\epsilon} = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}). \tag{47}$$

By simplifying we get,

$$S(\boldsymbol{\beta}) = \mathbf{y}'\mathbf{y} - \boldsymbol{\beta}'\mathbf{X}'\mathbf{y} - \mathbf{y}'\mathbf{X}\boldsymbol{\beta} + \boldsymbol{\beta}'\mathbf{X}'\mathbf{X}\boldsymbol{\beta}. \tag{48}$$

Since $(\boldsymbol{\beta}'\mathbf{X}'\mathbf{y})' = \mathbf{y}'\mathbf{X}\boldsymbol{\beta}$ and is a scalar we get,

$$S(\boldsymbol{\beta}) = \mathbf{y}'\mathbf{y} - 2\boldsymbol{\beta}'\mathbf{X}'\mathbf{y} + \boldsymbol{\beta}'\mathbf{X}'\mathbf{X}\boldsymbol{\beta}. \tag{49}$$

Now to find the $\boldsymbol{\beta}$ that minimize $S(\boldsymbol{\beta})$, $\hat{\boldsymbol{\beta}}$ need to satisfy,

$$\left.\frac{\partial S}{\partial \boldsymbol{\beta}}\right|_{\hat{\boldsymbol{\beta}}} = -2\mathbf{X}'\mathbf{y} + 2\mathbf{X}'\mathbf{X}\hat{\boldsymbol{\beta}} = \mathbf{0} \tag{50}$$

We move around and then get the final solution,

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{X})^{-1}\mathbf{X}'\mathbf{y}. \tag{51}$$

This solution will always exist if the regressors are linearly independent, i.e there is no column in X that is a linear combination of the other columns.

If we add weights to the least-square method defined as,

$$\mathbf{W} = \begin{bmatrix} w_1 & & & 0 \\ & w_2 & & \\ & & \ddots & \\ 0 & & & w_n \end{bmatrix}, \tag{52}$$

we get the initial least-square function,

$$S(\boldsymbol{\beta}) = \boldsymbol{\epsilon}'\mathbf{W}\boldsymbol{\epsilon} = (\mathbf{y} - \mathbf{X}\boldsymbol{\beta})'\mathbf{W}(\mathbf{y} - \mathbf{X}\boldsymbol{\beta}) \tag{53}$$

If we then follow the same steps as above we end up with the gollowing solution,

$$\hat{\boldsymbol{\beta}} = (\mathbf{X}'\mathbf{W}\mathbf{X})^{-1}\mathbf{X}'\mathbf{W}\mathbf{y} \tag{54}$$

If the reader want more details about this area we recommend strongly to look into [36].

8.5 Experiment 1: Comparing the SVI, SSVI & eSSVI fit

In this subsection we will compare the fitting capacity for the SVI, SSVI and eSSVI model. Each model is fitted against 10 different market data sets. For each data set is the total error, the mean error and error for the shorter maturities calculated. For each fit we have also checked if the surface has introduced any arbitrage. The result for each data set is presented in table 6 and a summary of that result is then presented in table 7.

Dataset	Method	Total Error	Mean Error	Short Maturity Error	Arbitrage
1	SVI	0.0001	0.0000	0.0000	1
	SSVI	0.0091	0.0023	0.0031	0
	eSSVI	0.0079	0.0020	0.0015	0
2	SVI	0.0001	0.0000	0.0000	1
	SSVI	0.0150	0.0038	0.0064	0
	eSSVI	0.0104	0.0026	0.0038	0
3	SVI	0.0000	0.0000	0.0000	1
	SSVI	0.0011	0.0003	0.0003	0
	eSSVI	0.0025	0.0006	0.0008	0
4	SVI	0.0000	0.0000	0.0000	1
	SSVI	0.0042	0.0010	0.0022	0
	eSSVI	0.0055	0.0014	0.0026	0
5	SVI	0.0001	0.0000	0.0000	1
	SSVI	0.0079	0.0020	0.0033	0
	eSSVI	0.0044	0.0011	0.0008	0
6	SVI	0.0000	0.0000	0.0000	1
	SSVI	0.0177	0.0044	0.0047	0
	eSSVI	0.0114	0.0028	0.0029	0
7	SVI	0.0000	0.0000	0.0000	1
	SSVI	0.0053	0.0013	0.0009	0
	eSSVI	0.0061	0.0015	0.0006	0
8	SVI	0.0000	0.0000	0.0000	1
	SSVI	0.0110	0.0027	0.0041	0
	eSSVI	0.0128	0.0032	0.0048	0
9	SVI	0.0001	0.0000	0.0001	1
	SSVI	0.0102	0.0026	0.0020	0
	eSSVI	0.0088	0.0022	0.0015	0
10	SVI	0.0000	0.0000	0.0000	1
	SSVI	0.0023	0.0006	0.0004	0
	eSSVI	0.0045	0.0011	0.0007	0

Table 6: Comparing method SVI, SSVI and eSSVI on 10 different data sets. Total error is the complete error against each data point in that data set for all slices. Mean error is the mean error for each slice. Short maturity error is the total error for the slices that corresponds to the shorter maturities.

Method	Total Error	Mean Error	Short Maturity Error	Arbitrage
SVI	0.00004	0.0000	0.00001	10/10
SSVI	0.0084	0.0021	0.0027	0/10
eSSVI	0.0074	0.0018	0.0020	0/10

Table 7: Summary of the result from table 6 in mean form.

From the result presented in the table above we can directly see that the SVI fit gives much closer fits then the SSVI and the eSSVI fit but on the other hand introduces arbitrage in every fit. This experiment is not enough to conclude that the SVI fit will always introduce arbitrage but it is safe to say that it seems to have a much higher possibility to do so compared to the SSVI and the eSSVI fit.

We can also see that the eSSVI fit will in general give better fits then the SSVI and we can see that the improvement in general is located at the shorter maturities. It is worth metioning that the optimization procedure might not have been perfect in this experiment since in theory should the eSSVI fit never be worse then the SSVI fit. We can see in table 6 that in some data set for example data set 4 that the eSSVI fit performs worse then the SSVI. The drawback with the eSSVI is that we get more parameters to fit and so the optimization becomes a bigger problem, it is therefore not unreasonable that similar problems can occur in practice if we would use the eSSVI.

The difference in performance between the eSSVI and the SSVI is so small so the alternative to have a optimizaztion problem with less parameters might be a argument to use the SSVI over the eSSVI.

To illustrate how the three models in general fit against the data sets we have plotted data set 1 in fig.(22).

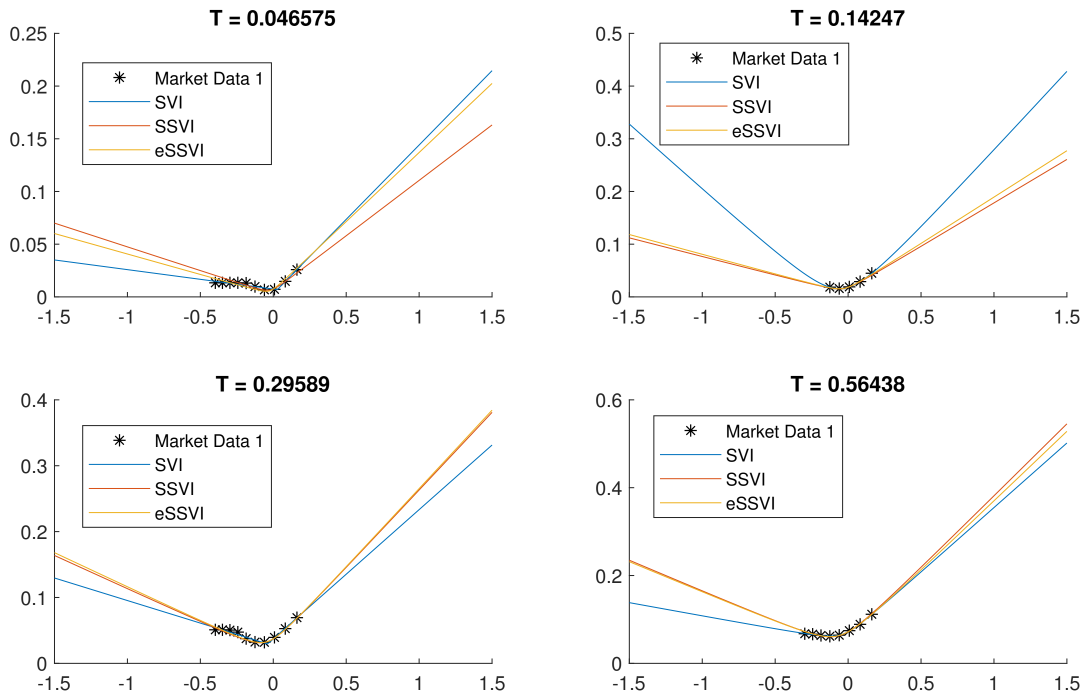

Fig. 22: This is a comparison between the SVI, SSVI and eSSVI fitted against data set 1 in table 6. Each subplot illustrate each slice depending on maturity.

8.6 Experiment 2: Comparing Calibration Strategies

To compare the different calibration strategies presented in section 5.7 we have in this section applied all 3 calibration strategies on one and the same SSVI fit.

In fig.(23) we can see all different complete market fits after applying each calibration strategy, in fig.(24) we can then see each slice compared against each other with corresponding butterfly arbitrage curve, in fig.(25) we compare each slice residuals and in fig.(26) we can see the total summary of these residuals.

From these results it is clear that the global calibration and the backward calibration is the better performing strategies on this market data.

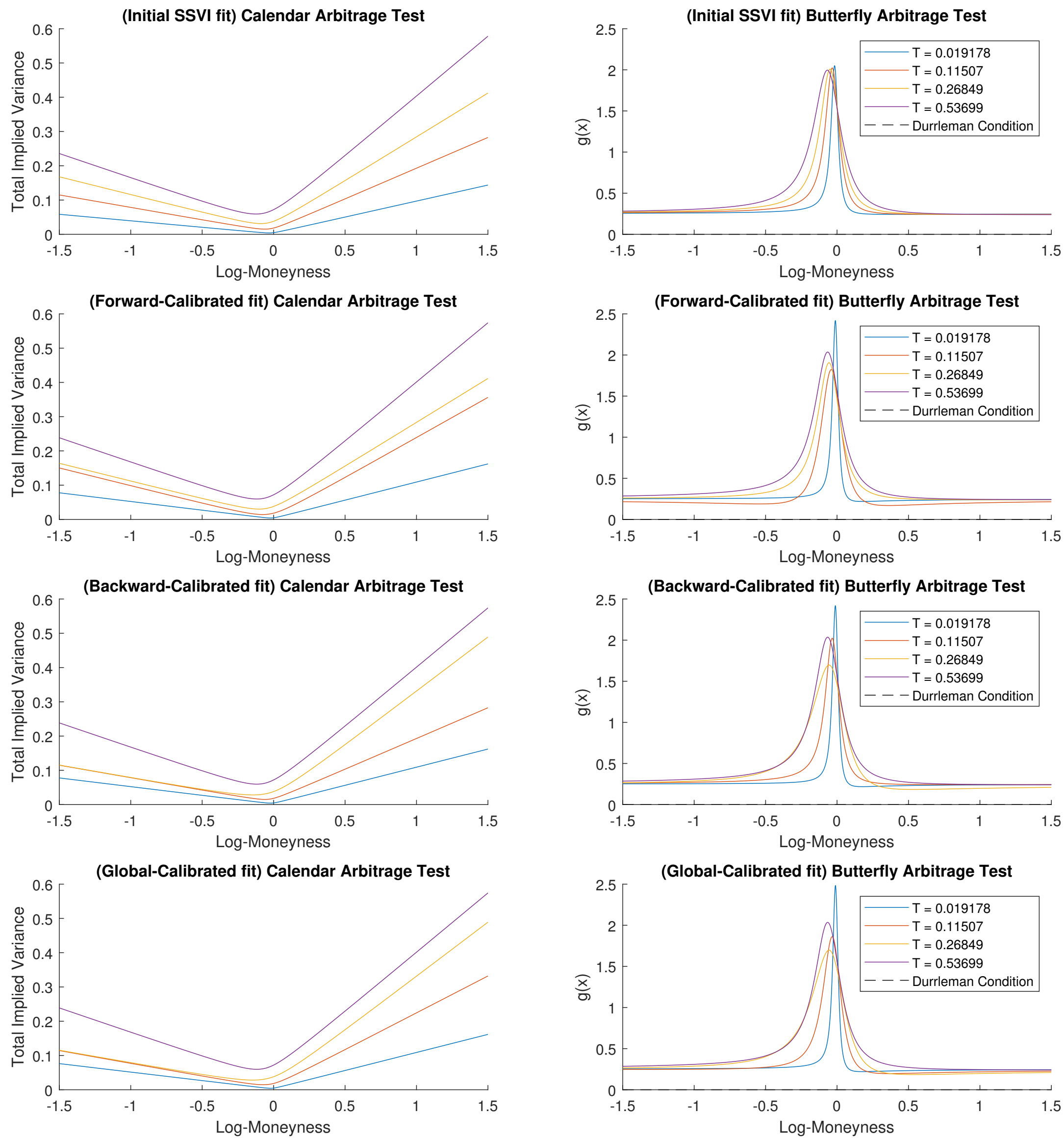

Fig. 23: Collection of all market data fits after applying each calibration strategy on the initial SSVI fit. To the right we have the corresponding butterfly arbitrage curves.

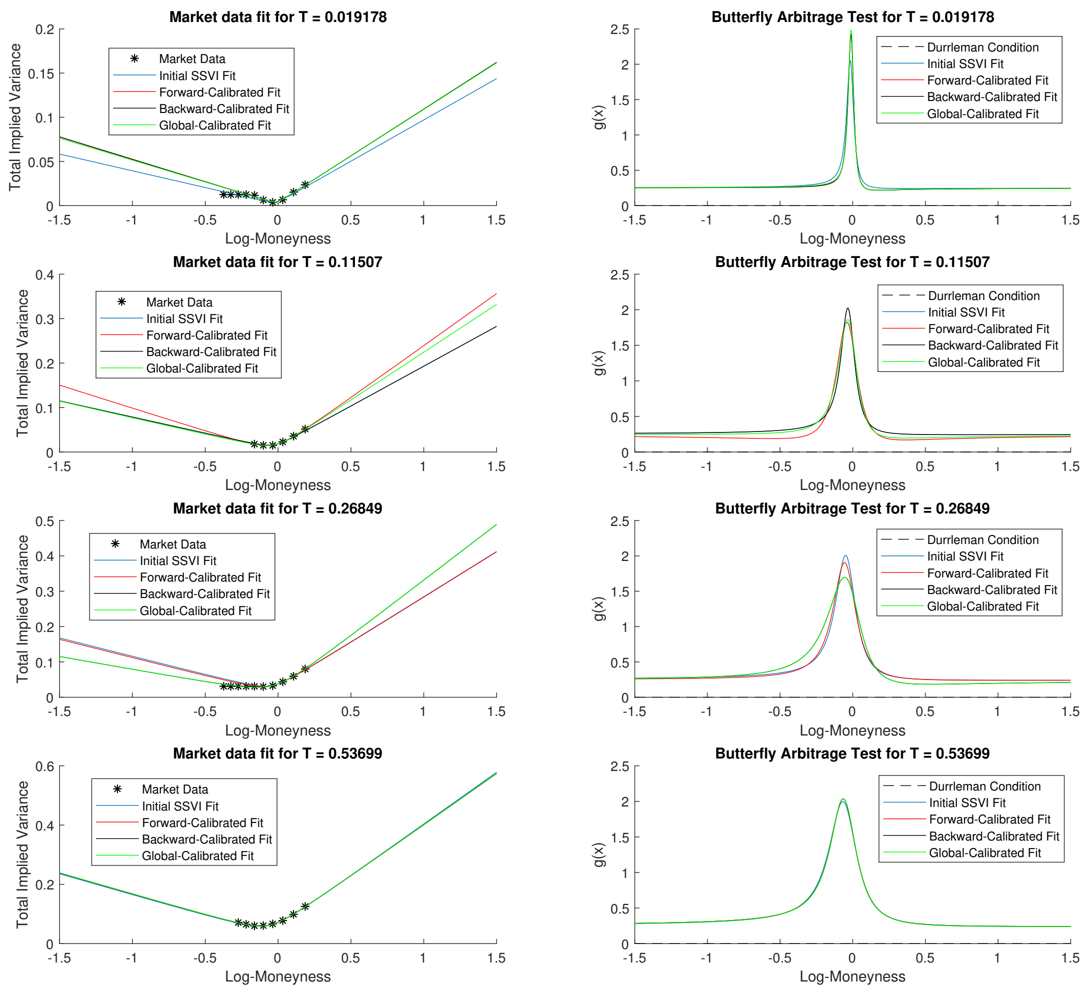

Fig. 24: Each slice from the fits in fig.(23) compared to each other. To the right we have the corresponding butterfly arbitrage curve.

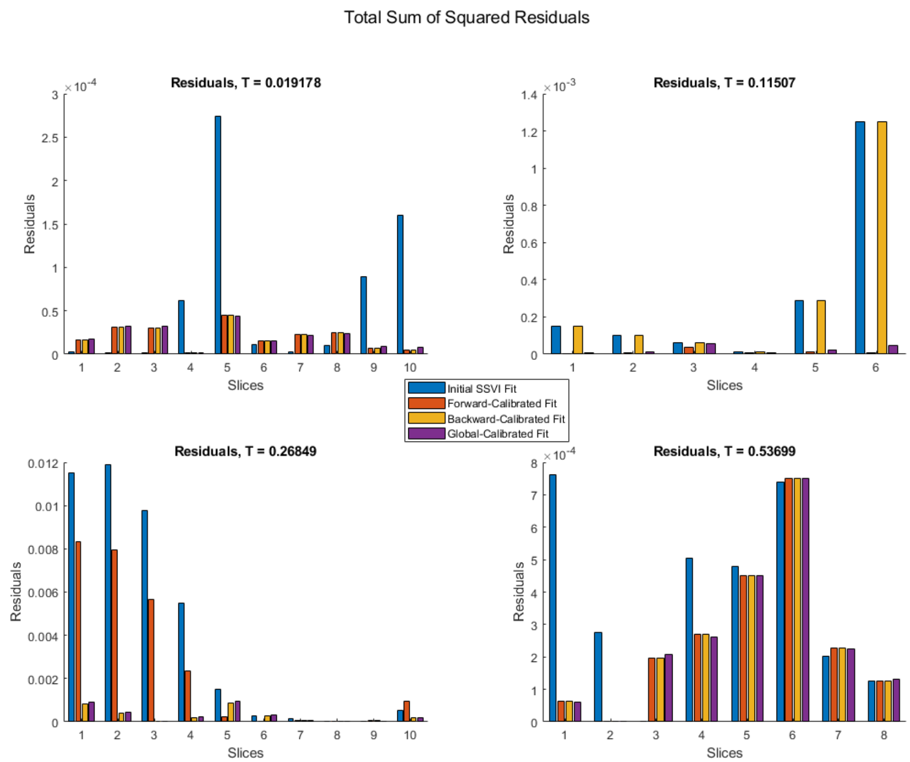

Fig. 25: Comparing residuals for each slice presented in fig.(24).

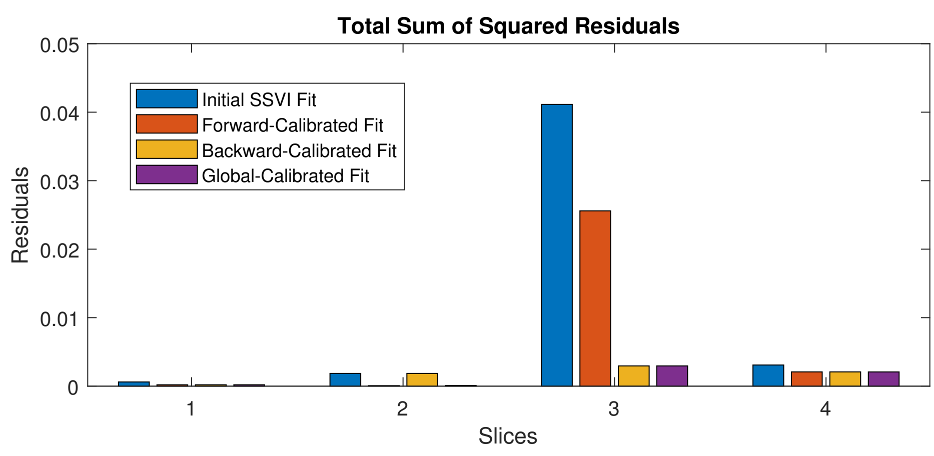

Fig. 26: Summary of the total sum of squared residuals presented in fig.(25).

8.7 Behavior when adjusting JW parameters

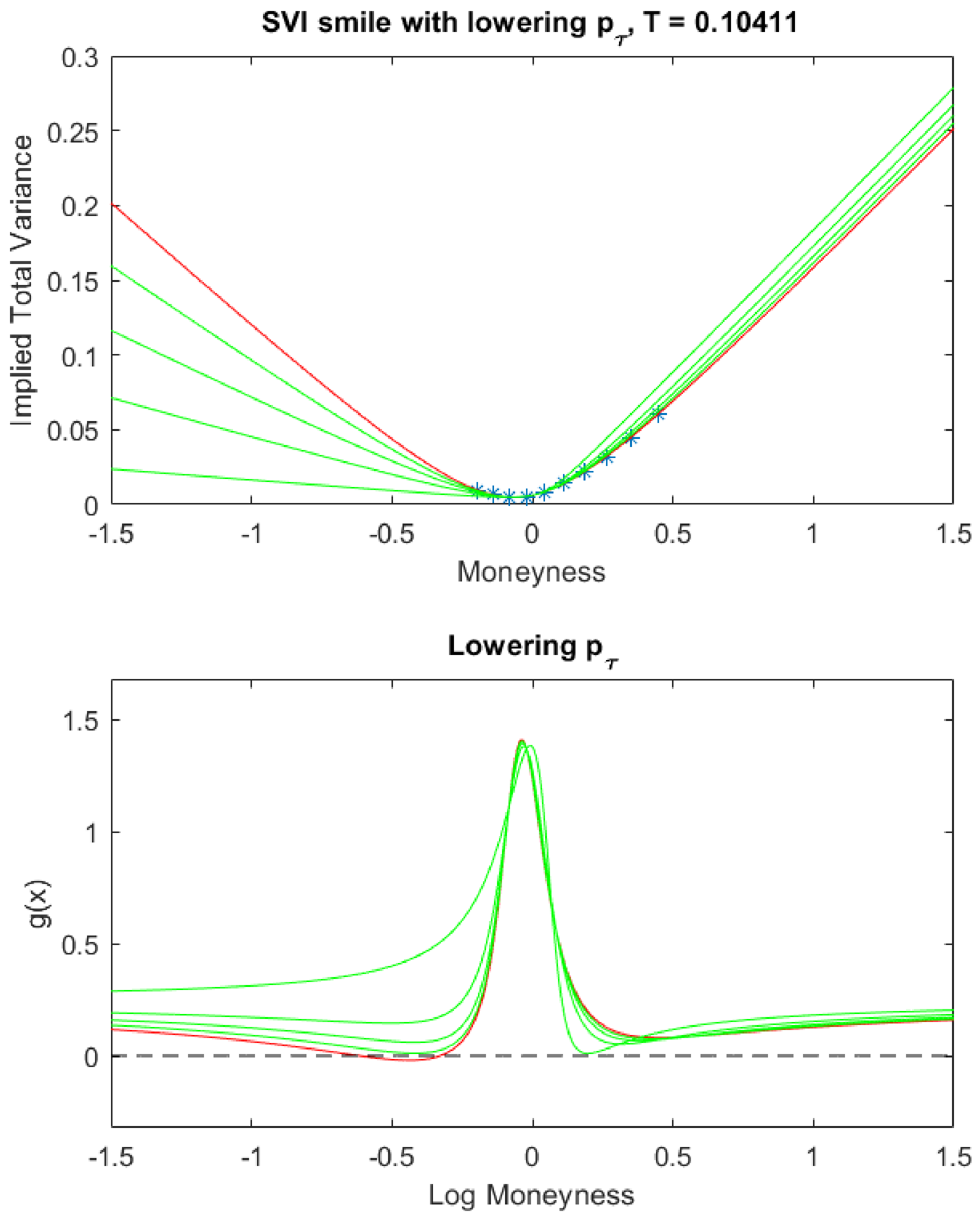

Fig. 27: This is a demonstration on how lowering p_τ affects the arbitrage curve in the bottom and the smile curve on the top.

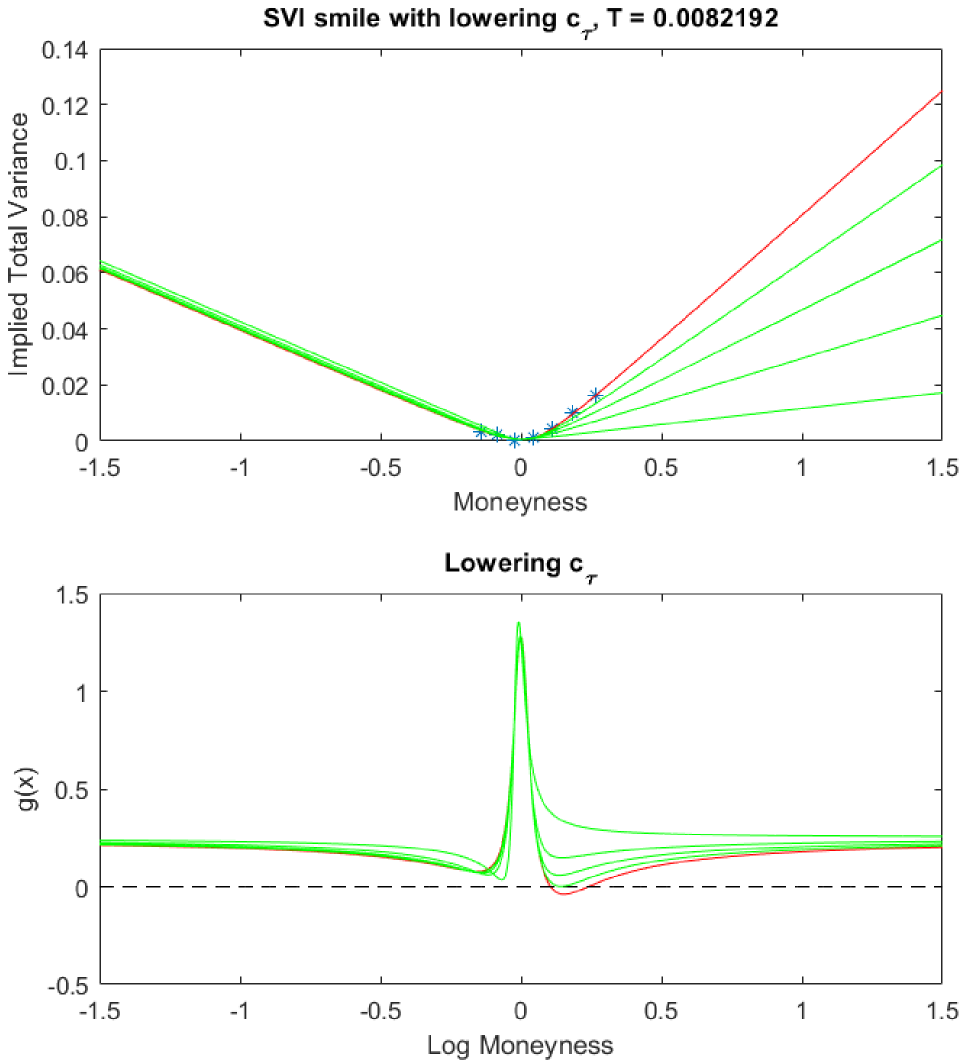

Fig. 28: This is a demonstration on how lowering c_τ affects the arbitrage curve in the bottom and the smile curve on the top.

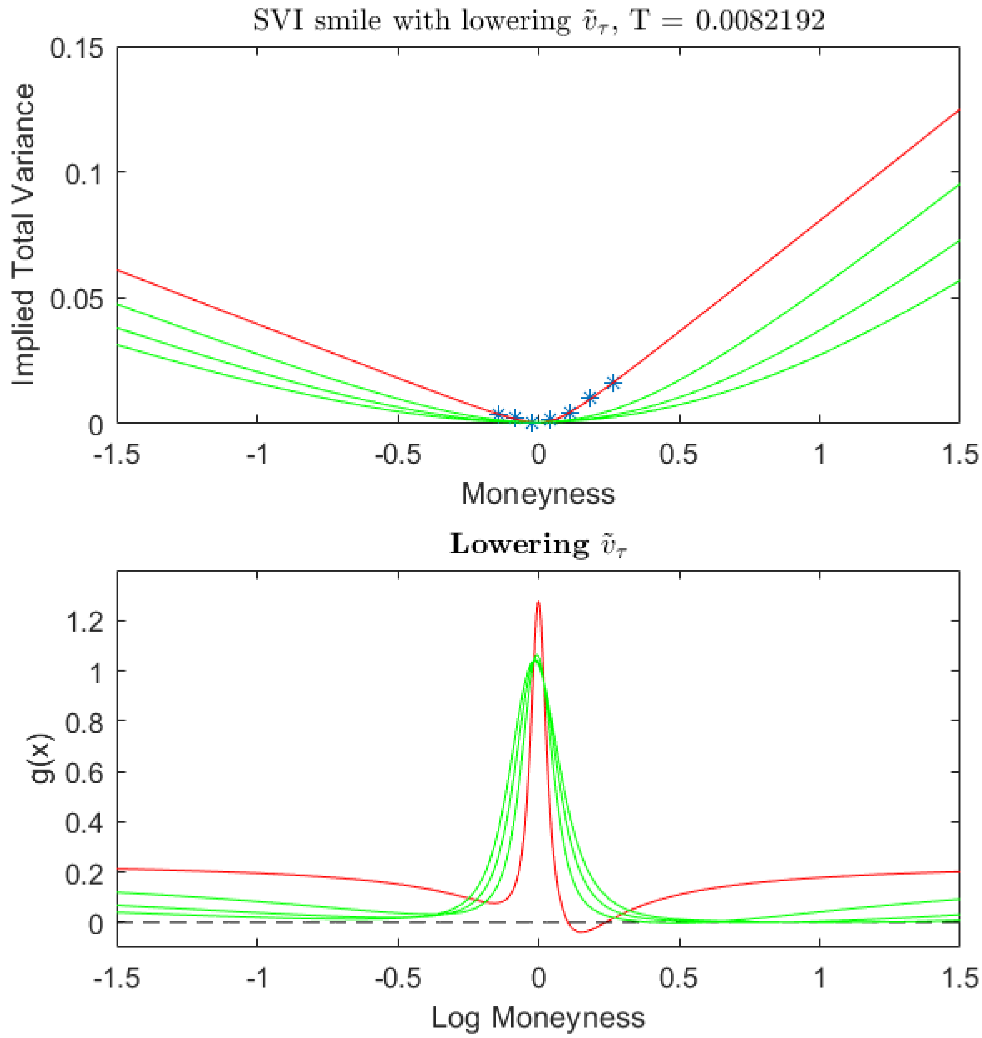

Fig. 29: This is a demonstration on how lowering $\tilde{v}_\tau$ affects the arbitrage curve in the bottom and the smile curve on the top.

8.8 Performance Analysis Development Results

In this subsection we are presenting the development of the performance against the market data. The result is presented by plots and is comparing the SSVI method with the calibrated SSVI method.

Fig. 30: Data set error development for the case of 1330.

Fig. 31: Slice mean error development for the case of 1330.

Fig. 32: Data set error development for the case of 1343.

Fig. 33: Slice mean error development for the case of 1343.

Fig. 34: Data set error development for the case of 6770.

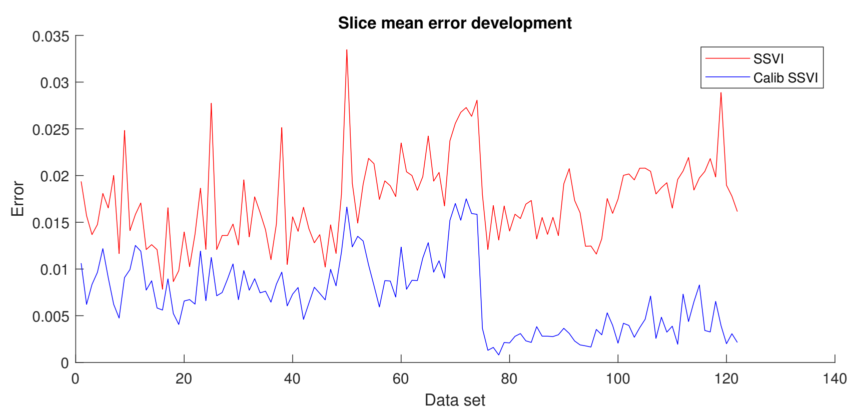

Fig. 35: Slice mean error development for the case of 6770.

Fig. 36: Data set error development for the case of Nikkei.

Fig. 37: Slice mean error development for the case of Nikkei.

Fig. 38: Data set error development for the case of Toppix.

Fig. 39: Slice mean error development for the case of Toppix.

References

[1] Duan, J.,Härdle, W.K. & Gentle, J.E., (2012). Option Data and Modeling BSM Implied Volatility, *Handbook of Compuational Finance.*

[2] Rubinstein, M., (1994). Implied Binomial Trees, *The Journal of Finance.*

[3] Fengler, M.R., Härdle, W. & Mammen, E. (2007). A semiparametric factor model for implied volatility dynamics. *Journal of Financial Econometrics.*

[4] Fengler, M.R, Härdle, W. & Mammen, (2005). A Dynamic Semiparametric Factor Model for Imlied Volatility String Dynamics.

[5] Fengler, M.R., (2005). Arbitrage-Free Smoothing of the Implied Volatility Surface, *Economic Risk Berlin.*

[6] Homescu, C. (2011). Implied volatility surface: construction methodologies and characteristics.

[7] Magnusson, E. (2018). Implied Volatility Surface Construcion.

[8] Cont, R. & Fonseca, J. (2002). Dynamics of implied volatility surfaces, *Quantitative Finance, Vol 2.*

[9] Cont, R. & Fonseca, J. (2002). Deformation of implied volatility surfaces: an empirical analysis, *Empirical Science of Financial Fluctuations.*

[10] Cont, R., Fonseca, J. & Durrleman, V. (2002). Stochastic Models of Implied Volatility Surfaces, *Economic Notes*, Vol 31.

[11] Peña, I, Rubio, G. & Serna, G. (1997), Why do we smile? On the determinants of the implied volatility function.

[12] Hull, J.C., (2012). Options, Futures and Other Derivatives. no. 8, pp. 218-231. Pearson Education.

[13] Björk, T. (2009). Arbitrage Theory in Continuous Time.

[14] Roper, M. (2010). Arbitrage Free Implied Volatility Surfaces, *The University of Sydney*

[15] Gatheral, J., (2004). A parsimonious arbitrage-free implied volatility parameterization with application to the valuation of volatility derivatives, *Presentation at Merril Lynch*

[16] Gatheral, J. & Jacquier, A., (2014). Arbitrage-free SVI volatility surfaces, *Quantitative Finance.*

[17] Kahalé, N., (2003). An Arbitrage-free Interpolation of Volatilities

[18] Kotze, A. & Joseph, A., (2009). Constructing a South African Index Volatility Surface from Exchange Traded Data.

[19] Kotzé, A., Labuschagne, C., Nair, M. & Padayachi, N., (2013). Arbitrage-free implied volatility surfaces for options on single stock futures, *Nort American Journal of Economics and Finance*

[20] Carr, P., (2004). Implied Vol Constraints, *Bloomberg.*

[21] Rebonato, R., (2004). Volatility and Correlation: The Perect Hedger and the Fox.

[22] Niu, Q. (2015). No arbitrage Conditions and Characters of Implied Volatility Surface, *A Review for Implied Volatility Modelers.*

[23] Kim, N. & Lee, J. (2012). No-arbitrage implied volatility functions: Empirical evidence from KOSPI 200 index options, *Journal of Empirical Finance.*

[24] Guo, G., Jacquier, A., Martini, C. & Neufcourt, L., (2016). Generalised Arbitrage-Free SVI Volatility Surfaces.

[25] Merton, C.R, (1973). Theory of Rational Option Pricing, *The Bell Journal of Economics and Management Science.*

[26] Carr, P. & Madan, D.B, (2005). A note on sufficient conditions for no arbitrage.

[27] Carr, P. & Wu, L. (2011), A New Simple Approach for Constructing Implied Volatility Surfaces, *Presentation at Columbia University.*

[28] Föllmer, H. & Schied, A. (2004). Stochastic finance: an introduction in discrete time, second revised and extended ed., Walter de Gruyter & Co., Berlin.

[29] Ioffe, I.D. & Prisman, E.Z., (2013). Arbitrage violations and implied valuations: the option market, *The European Journal of Finance.*

[30] Aurell, A., (2014). The SVI implied volatility model and its calibration.

[31] Rousseau, J-j., (2009). Quasi-Explicit Calibration of Gatheral's SVI model, *Zeliade Systems.*

[32] Benhamou, E., Arbitrage Spreads, *http://www.ericbenhamou.net/documents/Encyclo/Arbitrage%20spread.pdf*

[33] Hendriks, S. & Martini, C., (2017). The extended SSVI volatility surface, *Zeliade Systems.*

[34] Hendriks, S., (2016). Extending the SSVI model with arbitrage-free conditions. *Zeliade Systems.*

[35] Corbetta, J., Cohort, P., Laachir, I. & Martini, C, (2019), Robust calibration and arbitrage-free interpolation of SSVI slices, *Zeliade Systems.*

[36] Montgomery, D.C., Peck, E.A & Vining, G.G., (2012), Introduction to Linear Regression Analysis ,*Fourth Edition*

[37] Kovachev, Y., (2014), Calibration of stochastic volatility models

[38] Kilin, F., (2006), Accelerating the calibration of stochastic volatility models

[39] Mathworks, lsqnonlin, viewed 29 August 2019, https://se.mathworks.com/help/optim/ug/lsqnonlin.html.

[40] El Euch, O., Gatheral, J. & Rosenbaum, M., (2019), Roughening Heston, *Risk.net*.

[41] Goncu, A., (2014), Fitting the Heston Stochastic Volatility Model to Chinese Stocks.

[42] Joubert, P., (2005), Stochastic Volatility Models: Considerations for the Lay Actuary.

[43] Gatheral, J., (2006), The Volatility Surface: A Practitioner's Guide.

[44] Wang, Y., Yin, H. & Qi, L., (2004), No-Arbitrage Interpolation of the Option Price Function and Its Reformulation.

TRITA -SCI-GRU 2019:316