Numerical and optimal control methods for partial differential equations arising in computational finance

Mathematics PhD

James Arthur George Miles

University of Sussex

September 2018

Acknowledgements

I consider the eventual completion of this work to be the result of the unique combination of academic and emotional support I was fortunate enough to receive from my supervisor, family and friends. I thank my loving wife, Zlata, whose patience over the five year period ultimately outlasted my own. I could not have asked any more of you as you gave everything. I thank my supervisor, Dr Bertram Düring, for your generous enthusiasm, clarity of thought and kindness over the duration of this research. I thank my supportive parents, Mum and Dad, who observed the hardest period of this research at the closest range, you continued to support us in every imaginable way as we moved from Brighton to Milton Keynes and thereafter. SN, you provided the highest highs and you bounced me up from the lowest lows, never stop. Aliya, Beata, Craig, Dan and Zoe, I'll always remember your impeccably timed visit at the last hurdle as a work of brilliant friendship. Thanks guys. To the rest of my family and friends, thanks for tolerating my distracted and unfocused demeanour for such a long period of time. Your extra effort to engage with me during these times has always been noted and appreciated. Finally, I am thankful for the times spent with my affectionate family dogs, Charlie and Gnasher (RIP), your companionship was sometimes all I needed to keep typing.

Abstract

The chosen title for my PhD thesis is "Numerical and optimal control methods for partial differential equations arising in computational finance". The body of my research is divided into two parts. The first part is devoted to the application of an alternating direction implicit numerical method for solving stochastic volatility option pricing models. The second part focuses on a partial-integro differential equation constrained optimal control approach to parameter estimation for the forward jump-diffusion option pricing model. The body of the thesis is preceded by an extensive introduction, which seeks to contextualize my work with respect to the field of computational finance, this is followed by a brief conclusion. Finally, the thesis is completed by a list of references.

The first project proposes a new high-order alternating direction implicit (ADI) finite difference scheme for the solution of initial-boundary value problems of convection-diffusion type with mixed derivatives and non-constant coefficients, as they arise from stochastic volatility models in option pricing. The approach combines different high-order spatial discretisations with Hundsdorfer and Verwer's ADI time-stepping method, to obtain an efficient method which is fourth-order accurate in space and second-order accurate in time. Numerical experiments for the European put option pricing problem using Heston's stochastic volatility model confirm the high-order convergence.

The second project proposes to solve a parameter calibration problem for the forward jump-diffusion option pricing model proposed by Andersen and Andreasen. A distributed optimal control approach is employed, with a partial-integro differential equation as our state equation. By approaching the problem from a functional analysis perspective, I investigate the necessary regularity conditions for our parameters of interest. Following this, the existence of optimal solutions is proven under certain analytical conditions. Furthermore, the first-order necessary conditions for optimality are also established. Finally, a projected-gradient optimization method is applied numerically to empirical market data and results are given.

Contents

Chapter 1

Introduction

1.1 Financial derivatives overview

A financial derivative is a contract between two parties extending over some finite period which derives its value from some underlying financial asset. The writer of the contract is responsible for setting the terms of the contract and honouring them, while the buyer pays a premium at the outset of the contract for the privilege of entering a potentially profitable scenario. The way in which the financial derivative derives its value from the underlying financial asset is the fundamental characteristic that distinguishes different types of derivative from one another. The relationship between the value of entering such a contract, or holding an existing contract, and the perceived behaviour of the underlying asset can be quite complex from a mathematical standpoint, thus the field of financial derivatives has become a rich trove of mathematical research.

In practice, financial derivatives are frequently used to hedge or offset the gains and losses incurred through the ownership of particular financial assets and give market participants the opportunity to profit from speculating over the value of particular financial assets without the responsibilities and duties associated with ownership. A prevalent financial derivative known as an option takes the form of a contract, guaranteeing the buyer the right to purchase a certain volume of a particular financial asset at a price which will be determined by the difference between an agreed strike price and the future price of the asset at the contracts time of expiry. Financial derivatives magnify potential returns by drastically reducing the initial outlay for speculators. They are available in various forms that provide buyers with the opportunity to profit from rises or falls in the value of the financial asset. Establishing the value of entering such a contract guaranteeing the "right to buy at a later date" is a fairly non-trivial problem. The seller of the contract must quantify the stochastic behaviour of the underlying financial asset and price the derivative according to the corresponding mathematical model that satisfies a no arbitrage constraint. A classic example is the celebrated Black-Scholes option pricing model [5].

In the Black-Scholes model, it is assumed that the future price of the underlying asset (equity, commodity or currency etc) follows a geometric Brownian motion, while the

future returns of the asset are log-normally distributed. The volatility parameter for this stochastic process (measured as the standard deviation of the log returns on the asset) is taken to be some constant value, prescribed by the writer (seller) of the option:

$$\frac{dS(t)}{S(t-)} = \mu dt + \sigma dW(t), \tag{1.1}$$

where $S(t) \in \mathbb{R}^+$ is the value of some financial asset at time $t \in \mathbb{R}^+$, $\mu \in \mathbb{R}_0^+$ is the expected rate of return of the financial asset, $\sigma \in \mathbb{R}_0^+$ is the standard deviation of the return of the financial asset and $W(t)$ is a stochastic variable that follows a Brownian motion. (1.1) is short hand notation for the integral equation:

$$S(t) - S(0) = \int_0^t \mu S(t) ds + \int_0^t \sigma dW(t), \tag{1.2}$$

where the second integral is a stochastic Ito integral. If we assume that this stochastic equation accurately models the behaviour of the underlying financial asset, then we can use Ito's lemma to model the stochastic behaviour of some arbitrary function, $V(S,t)$, of $S(t)$ and t, as follows:

$$dV = (\mu S \frac{\partial V}{\partial S} + \frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2})dt + \sigma S \frac{\partial V}{\partial S} dW. \tag{1.3}$$

In order to satisfy no-arbitrage assumptions, we consider the value of some portfolio $P(t)$ over some time period $dt = [t, t+\Delta t]$ consisting of one short option $V(S,t)$ (increases in value for decreasing value of asset $S(t)$) and an amount of the financial asset, $S(t)$, specified by the value $\frac{\partial V(S,t)}{\partial S(t)}$:

$$P(t) = -V(S,t) + \frac{\partial V}{\partial S} S(t). \tag{1.4}$$

As the portfolio should be risk-free, the above change in value must be equal to an increase in portfolio value at the riskless rate of interest $r(t) \in \mathbb{R}^+$:

$$\Delta P = -\Delta V + \frac{\partial V}{\partial S}\Delta S = r(t)P(t)dt. \tag{1.5}$$

Substituting equations 1.1 and 1.3 into 1.5 and rearranging yields the Black Scholes PDE:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0. \tag{1.6}$$

This equation holds for $S > 0$ for the extent of the contract $t \in [0, T)$ with a prescribed final payoff at the expiry of the contract $t = T$ and boundary conditions dependent on the type of contract. European (or vanilla) options are simply priced and commonly traded. A European Put option prices the payoff at expiry as $V(S,T) = \max(K - S, 0)$, where $K > 0$ is some predetermined strike price. The corresponding boundary conditions for $V(S,t)$ in space for such a contract are given as $V(0,t) = Ke^{-r(T-t)}$ and $\lim_{S\to\infty} V(S,t) = 0$. Such a contract gains in value when the value of the underlying financial asset drops over time. A European Call option behaves symmetrically in that it gains in value for increases

in the value of the underlying financial asset over the extent of the contract. The final condition prices the payoff at expiry as $V(S,T) = \max(S-K,0)$ where $K > 0$ is some pre-agreed strike price. The corresponding boundary conditions for $V(S,t)$ in space for such a contract are given as $V(0,t) = 0$ and $\lim_{S\to\infty} V(S,t) = S$.

1.2 Option Pricing

If we determine that the underlying financial asset follows Black-Scholes' assumed stochastic behaviour, then the value of either European option $V(S,t)$ for any $t \in [0,T)$ and $S \in [0,\infty)$ can be determined by solving equation (1.6). A closed form solution exists for the Black-Scholes PDE, thus pricing options via Black-Scholes is a computationally trivial task. Despite this, the model is flawed from both an analytical perspective and an empirical one.

The assumption that volatility is an independent parameter, is contradicted by empirical market data. By calculating the implied volatility from many Calls or Puts for the same underlying financial asset, we observe higher volatility values for more extreme strike values. The Black-Scholes model fails to take into account the non-negligible risk of extreme movements and systemic risk in markets through its normally distributed returns, therefore it would appear that writers of options are prescribing higher volatility values for extreme strike values, resulting in higher option prices to hedge against extreme movements in the underlying financial asset. Another issue stemming from accurate parameterization of volatility is that the value of volatility is not necessarily constant. The adoption of a stationary stochastic model (in that its parameters are constant in time) is a serious oversimplification of the behaviour of some financial assets. It is possible that the stochastic behaviour of the underlying financial asset could be drastically altered during some intermediate period of the contract, thus the prior prescription of a constant value for volatility based on past behaviour would fail to reflect the new reality of the financial asset's dynamics.

For the above reasons, alternative pricing models have been proposed and adopted over the years. Many of these models follow a similar derivation process, with the key distinctions being made in the assumed stochastic behaviour of the underlying financial asset. These models do not typically admit closed form solutions. As a result, it is quite common for a model's associated partial differential equations, along with their boundary and final conditions, to be solved numerically. In order to price options robustly according to these models, we must be able to solve the corresponding equations efficiently and with sufficient accuracy. This requirement has led to an expansion in the development and implementation of numerical methods specifically aimed at solving option pricing problems accurately while attempting to minimize computational effort.

In financial option pricing, stochastic volatility models such as the Heston model [32] have become one of the standard approaches. The underlying asset is modelled by the following processes, where asset price $0 \leq S(t) < \infty$ and volatility $0 \leq \sigma(t) < \infty$ for

$t \in [0, T]$:

$$dS(t) = \bar{\mu} S(t) dt + \sqrt{\sigma(t)} S(t) dW^{(1)}(t), \tag{1.7}$$

$$d\sigma(t) = \kappa(\theta - \sigma(t))dt + v\sqrt{\sigma(t)} dW^{(2)}(t), \tag{1.8}$$

where $dW^{(1)}(t)$ and $dW^{(2)}(t)$ are correlated Brownian motions with constant correlation parameter $dW^{(1)}(t)dW^{(2)}(t) = \rho dt$. The mean return of the asset is given by $\bar{\mu} \in \mathbb{R}$, $\kappa \geq 0$ and $\theta \geq 0$ are the mean reversion speed of $\sigma(t)$ and the long run mean of $\sigma(t)$, respectively and $v \geq 0$ is the volatility of volatility. Unlike the classical Black-Scholes model [5] the volatility of the option's underlying asset is not assumed to be constant, but is modelled as a second, correlated stochastic diffusive process and therefore the option price is modelled as a function of asset price, volatility and time. This additional source of randomness allows us to model option prices more accurately and to fit higher moments of the asset return distribution. Using Ito's lemma and standard arbitrage arguments, a partial differential equation of convection-diffusion type with mixed second-order derivatives is derived for pricing options, solved by $V(S, t) \geq 0$:

$$V_t + \frac{S^2\sigma}{2} V_{SS} + \rho v \sigma S V_{S\sigma} + \frac{v^2\sigma}{2} V_{\sigma\sigma} + rSV_s + [\kappa(\theta - \sigma) - \lambda_0 \sigma] V_\sigma - rV = 0, \tag{1.9}$$

where $\lambda_0 \sigma(t)$ is the market price of volatility risk and $\lambda_0 \in \mathbb{R}$. For some stochastic volatility models and under additional restrictions, closed-form solutions can be obtained by Fourier methods (e.g. [32], [22]). Another approach is to derive approximate analytic expressions, see e.g. [4] and the literature cited therein. In general, however, —even in the Heston model [32] when the parameters in it are non constant— the partial differential equations arising from stochastic volatility models have to be solved numerically. Moreover, many (so-called American) options feature an additional early exercise right. Then one has to solve a free boundary problem which consists of the partial differential equation and an early exercise constraint for the option price. Also for this problem one typically has to resort to numerical approximations.

In the mathematical literature, there are many papers on numerical methods for option pricing, mostly addressing the one-dimensional case of a single risk factor and using standard, second order finite difference methods (see, e.g., [59] and the references therein). More recently, high-order finite difference schemes (fourth order in space) were proposed [28, 52, 57] that use a compact stencil (three points in space). In the option pricing context, see e.g. [20, 19, 47].

There are less works considering numerical methods for option pricing in stochastic volatility models, i.e., for two spatial dimensions. Finite difference approaches that are used are often standard, second order methods, e.g. in [38] where different efficient methods for solving the American option pricing problem for the Heston model are proposed. Other approaches include finite element-finite volume [63], multigrid [10], sparse wavelet [33], or spectral methods [62].

The classical alternating direction implicit (ADI) method, introduced by Peaceman and Rachford [51], Douglas [15, 16], Fairweather and Mitchell [50], is a very powerful

method that is especially useful for solving parabolic equations (*without* mixed derivative terms) on rectangular domains. Beam and Warming [3], however, have shown that no simple ADI scheme involving only discrete solutions at time levels n and $n+1$ can be second-order accurate in time in the presence of mixed derivatives. To overcome this limitation and construct an unconditionally stable ADI scheme of second order in time, a number of results have been given by Hundsdorfer and Verwer [36, 37] and more recently by in't Hout and Welfert [35]. These schemes are second-order accurate in time and space. In [34] different second-order ADI schemes of this type are applied to the Heston model. In [24] this approach was combined with different high-order discretisations in space, using high-order compact schemes for two-dimensional convection-diffusion problems *with mixed derivatives and constant coefficients*. In [31] this approach was combined with sparse grids and applied to a multi-dimensional Black-Scholes equation, again *with constant coefficients*. A high order compact computational stencil in space combined with a Crank Nicolson time stepping method is considered for solving stochastic volatility option pricing models in [23], achieving fourth-order accuracy in space and second-order accuracy in time. [25] extends the application of a high order compact computational stencil in space for solving stochastic volatility option pricing models by implementing a Rannacher smoothed finite difference method in time for non-uniform meshes. In both cases we observe that, in general, the resulting linear systems cannot be solved with high efficiency due to the implicit time-stepping methods employed.

In the second chapter of this thesis I have contributed a high-order ADI method for option pricing in stochastic volatility models, combining and extending the ideas presented in [23, 24, 25]. This involves two-dimensional convection-diffusion equations *with mixed derivative terms and space-dependent coefficients* which adds substantial algebraic complexity in the derivation of the scheme. The new scheme is second-order accurate in time and fourth-order accurate in space.

1.3 Calibrating Parameters to market data

Due to the public availability of option prices over a wide variety of strikes and expiries, it is possible to calibrate the parameters of a given option pricing model to the market data. Parameter estimation and model inversion has existed and developed as a topic within financial mathematics for over forty years, with the first noted case given by [45]. In this paper, the authors seek to estimate the volatility parameter of the Black-Scholes model using market prices and directly observable information by employing a simple numerical search method to minimize the difference between the price generated by Black-Scholes' model and an empirical European Call price. Over time, the Newton-Raphson iterative numerical method, as shown by [48], has remained a popular choice among practitioners due to its quadratic convergence. The Brent Dekker [7] algorithm has also been employed successfully due to its unconditional convergence. Alternatively, closed form approximations provide practitioners with speed of computation but lack accuracy. A combined

approach, choosing some closed-form approximation as a starting point for an iterative method, is also common in practice.

Due to the observed dependence of volatility on option strike and expiry, it would be desirable to imply a volatility surface as a function of these strikes and expiriess from a collection of option prices on a particular financial asset. Dupire [18] presented a forward diffusion equation, satisfied by the option price as a function of strike, $0 \leq K \leq K_{max}$, and expiry, $0 \leq T \leq T_{max}$, while keeping the underlying assets value and time fixed to the present:

$$\frac{\partial V}{\partial T} = \frac{1}{2}\sigma^2(K,T)K^2\frac{\partial^2 V}{\partial K^2} - rK\frac{\partial V}{\partial K}, \tag{1.10}$$

where the initial condition and boundary conditions are analogous to their Black-Scholes counterparts and correspond to whether the option is a put or call. In this model, the volatility parameter for asset returns is modelled as a deterministic 'local volatility' surface as a function of the option's strike price and expiry. Much work has been done in this sphere for implying such a volatility surface using various methods. While it is possible to invert the model for an explicit expression for the volatility surface, the scarcity and noisiness of market data prevents us from evaluating this expression directly. Approaches to treating this issue include residual minimization with interpolation of market data or interpolation of discrete implied volatilities [6, 55], and minimum relative entropy methods which treat the volatility surface probabilistically [2]. Another solution is to apply Tikhonov regularization to stabilize the otherwise ill-posed inverse problem of residual minimization. Minimizing a regularized cost functional, where the gradient of the local volatility is a contributing element, controls the smoothness of the calibrated result. An introduction to Tikhonov regularization methods for solving generalized inverse problems is given by Tikhonov et. al. in [60, 69], while a survey of the modern theory of regularized inverse problems, in general, is given in [73]. In [43], Lagnado and Osher adopt a Tikhonov regularized approach to identify the volatility parameter from Dupire's forward equation via gradient descent method. The computation of the directional derivative of the cost functional with respect to a change in the volatility surface requires numerically solving a PDE at every point on the discrete domain of the volatility surface for every iteration of the gradient descent procedure. [64] supplies convergence and stability results from the theory of Tikhonov regularization for the same problem and conducts convergence rate analysis under restrictive assumptions. In [21], the authors have produced a thorough analysis of an adjoint based regularized optimal control approach for implying the volatility surface, providing results for first and second order conditions of optimality. Alternatively, [68] chooses a spline representation for the implied volatility surface, thus restricting the calibrated local volatility surface to smooth representations, where the weights of the spline are optimized to minimize a Tikhonov regularized cost functional. In [66, 67], Achdou et. al. extend the use of Tikhonov regularization to calibrate local volatility to the case of American vanilla options.

We have adopted the adjoint-based gradient descent method for minimizing a regularized cost functional in the third chapter for calibrating a different option pricing model: the jump-diffusion model initially introduced by Merton [49]. The jump-diffusion model includes the modelling of possible jumps via a Poisson process with a jump-size modelled by either a log-normal distribution or double exponential distribution, as suggested by [39]. This jump term intends to take account of the suddenly discontinuous behaviour that we expect to observe in markets over long periods of time. A primer on jump-diffusion models is provided by [58]. From these dynamics Andersen and Andreasen [1] extended the work of Dupires approach by deriving a forward partial integro-differential equation based on the original jump-diffusion pricing model. This models the option price and corresponding parameters as functions of strike and expiry, while the underlying asset value and time is fixed at their present values.
As in Dupire's work, the advantage of calibrating parameters, for a domain of strikes and expiries, to the forward variant of the jump-diffusion model is that we repeatedly solve a single PIDE over the required domain. Attempting to calibrate to the traditional pricing model would require us to repeatedly solve a PIDE for every point in our domain, a much more computationally expensive task. [30] considers a discrete approach to calibrating a local volatility surface to the forward jump-diffusion model. [8] considers the transformation of the jump-diffusion model and the calibration of model parameters through closed form approximations followed by interpolation of results. R. Cont et. al. explore the application of relative-entropy methods for jump-diffusion parameter calibration and calibrate an exponential Lévy model which minimizes a regularized price residual, where the relative entropy with respect to some prior model performs the necessary regularization in [12, 11]. [54] outlines an optimal control approach involving an adjoint equation for the computation of the gradient for calibrating the volatility surface, Poisson parameter and the jump size distribution parameters to market prices, although the adjoint equation itself is not specified. We have aimed to approximate the local volatility function and the jump term's Poisson parameter simultaneously by reformulating the problem as a distributed optimal control problem, treating the forward PIDE as our state equation and applying box constraints on the controls. We have produced a result for the existence of a solution to the forward jump-diffusion equation, via a Galerkin approximation. We have thoroughly investigated the necessary regularity conditions for our parameters of interest. We also prove the existence of unique optimal solutions under certain analytic conditions. Furthermore, we have established the first-order necessary conditions for optimality and explicitly derived the adjoint equation.

We have applied a projected gradient descent technique to alternately update the volatility surface or expiry dependent Poisson parameter at the end of each iteration. In order to calculate the gradient of the cost functional with respect to the controls, a particularly efficient approach is the adjoint state method. This method requires us to derive a Lagrangian functional. The Lagrangian multipliers themselves are are solutions to the

adjoint equation, which closely resembles the state equation. To calculate the direction of steepest descent for the regularized cost functional, we must numerically solve the state PIDE and the adjoint PIDE as a coupled system. Alternately, one of our parameters is updated at the end of each iteration of the gradient descent algorithm.

The numerical experiments consist of two parts. First, we apply the adjoint-state gradient descent method by calibrating our parameters to self-produced, model-abiding option prices. The de-noised option prices are generated by prescribing a volatility surface and a Poisson parameter to the forward jump-diffusion model and numerically solving this model over the domain of some finite time-space cylinder. Second, we calibrate our parameters to historical European Call option prices over various strikes and expiries for a FTSE 100 index equity. The rate and magnitude of reductions in the residual sufficiently demonstrate that our theorized optimal control approach is successful in implying the volatility surface and Poisson parameter to an acceptable degree of accuracy. Finally, we discuss the possibility of extending the scheme to include calibrating to price data for an additional class of option.

1.4 Thesis overview

The second chapter presents my work in implementing a high order compact alternating direction implicit numerical method for solving stochastic volatility option pricing models. By combining a high order compact computational stencil with Hundsdorfer's alternating direction implicit numerical method, I have produced a new efficient solving method for stochastic volatility option pricing models with an accuracy that is of fourth order convergence in space and of second order convergence in time. The results of numerical experiments for solving Heston's option pricing model show strong evidence for this. The third chapter presents my work in implying a volatility surface and Poisson parameter for the forward jump-diffusion option pricing model, presented in [1], using an optimal control approach with a gradient descent method. The existence of a solution, under specific regularity constraints, has been provided for the forward jump-diffusion option pricing model. By employing a Tikhonov regularized cost functional, I have proven that an optimal set of controls exist and derived first-order necessary conditions of optimality, under specific regularity constraints. Through the application of a gradient descent method, numerical experiments demonstrate the methodology for calibrating a transformed local volatility and expiry dependent Poisson parameter to artificial and empirical price data.

Chapter 2

High order compact alternating direction implicit finite difference method for option pricing in stochastic volatility models

We propose a new high-order alternating direction implicit (ADI) finite difference scheme for the solution of initial-boundary value problems of convection-diffusion type with mixed derivatives and non-constant coefficients, as they arise from stochastic volatility models in option pricing. The approach combines different high-order spatial discretizations with Hundsdorfer and Verwer's ADI time-stepping method, to obtain an efficient method which is fourth-order accurate in space and second-order accurate in time. In section 2.1, we introduce stochastic volatility models for option pricing and the related pricing partial differential equation. In Section 2.2 we recall the Hundsdorfer-Verwer ADI splitting in time. For the spatial discretization, we introduce different high-order methods, in 2.3 for the implicit steps, and in 2.4 for the explicit steps. The solution of the resulting scheme and numerical boundary conditions are discussed in 2.5 and 2.6. Finally, numerical convergence and stability results are presented in 2.7.

2.1 Stochastic volatility models

Consider the following class of stochastic volatility models: asume that asset spot price $0 \leq S(t) < \infty$ and variance $0 \leq \sigma(t) < \infty$ follow two stochastic diffusive processes for $t \in [0, T]$,

$$dS(t) = \bar{\mu} S(t) dt + \sqrt{\sigma(t)} S(t) dW^{(1)}(t), \tag{2.1a}$$

$$d\sigma(t) = \kappa(\sigma(t))^{\alpha}(\theta - \sigma(t))dt + v(\sigma(t))^{\beta} dW^{(2)}(t), \tag{2.1b}$$

which are characterised by two Brownian motions, $dW^{(1)}(t)$ and $dW^{(2)}(t)$, with constant correlation parameter $dW^{(1)}(t)dW^{(2)}(t) = \rho dt$. The drift coefficient for stochastic asset

returns is given by the mean return of the asset where $\bar{\mu} \in \mathbb{R}$ and the diffusion coefficient is given by $\sqrt{\sigma(t)}S(t)$.

The drift coefficient of the asset variance is given by $\kappa(\sigma(t))^{\alpha}(\widetilde{\theta}-\sigma(t))$, where constants $\kappa \geq 0$ and $\theta \geq 0$ are the mean reversion speed of $\sigma(t)$ and the long run mean of $\sigma(t)$, respectively. The diffusion coefficient is given by $v(\sigma(t))^{\beta}$ where constant $v \geq 0$ is the volatility of volatility. The constant riskless interest rate is denoted by $r \geq 0$. The constants α, β determine the stochastic volatility model used.

The class of stochastic volatility models (2.1) includes a number of known stochastic volatility models: The most prominent stochastic volatility model, the *Heston model* [32] (also called *square root (SQR) model*) specifies the variance by

$$d\sigma(t) = \kappa\left(\theta - \sigma(t)\right)\mathrm{d}t + v\sqrt{\sigma(t)}\mathrm{d}W^{(2)}(t).$$

Other known stochastic volatility models include the *GARCH* (or *VAR model*) model, see [17], where the stochastic variance is modelled by

$$d\sigma(t) = \kappa\left(\theta - \sigma(t)\right)\mathrm{d}t + v\sigma(t)\mathrm{d}W^{(2)}(t),$$

and the *3/2 model* [46] in which the variance follows the process

$$d\sigma(t) = \kappa\left(\theta - \sigma(t)\right)\mathrm{d}t + v\sigma^{\frac{3}{2}}(t)\mathrm{d}W^{(2)}(t).$$

All of the three stochastic volatility models mentioned above use a linear mean-reverting drift for the stochastic process of the variance $v(t)$, but there are also models, in which the drift is mean reverting in a non-linear fashion. Following [9], we denote these models with an additional "N": in the *SQRN model* the stochastic variance follows

$$d\sigma(t) = \kappa\sigma(t)\left(\theta - \sigma(t)\right)\mathrm{d}t + v\sqrt{\sigma(t)}\mathrm{d}W^{(2)}(t),$$

in the *VARN model*

$$d\sigma(t) = \kappa\sigma(t)\left(\theta - \sigma(t)\right)\mathrm{d}t + v\sigma(t)\mathrm{d}W^{(2)}(t),$$

and in the 3/2-*N model*

$$d\sigma(t) = \kappa\sigma(t)\left(\theta - \sigma(t)\right)\mathrm{d}t + v\sigma^{\frac{3}{2}}(t)\mathrm{d}W^{(2)}(t),$$

see [9].

Applying standards arbitrage arguments and Ito's lemma to the class of stochastic volatility models (2.1), we can derive the following second order partial differential equation for any financial derivative $V(S, \sigma, t)$, to be solved backwards in time with $0 < S < \infty$, $0 < \sigma < \infty$, $t \in [0, T)$:

$$V_t + \frac{S^2\sigma}{2}V_{SS} + \rho v\sigma^{\beta+\frac{1}{2}}SV_{S\sigma} + \frac{v^2\sigma^{2\beta}}{2}V_{\sigma\sigma} + rSV_s + [\kappa\sigma^{\alpha}(\theta - \sigma) - \lambda_0\sigma]V_{\sigma} - rV = 0. \quad (2.2)$$

Here, $\lambda_0\sigma(t)$ is the market price of volatility risk, where $\lambda_0 \in \mathbb{R}$, which is usually assumed to be proportional to the variance. In the following we assume $\lambda_0 = 0$ for streamlining

the presentation. The generalisation to the case $\lambda_0 \neq 0$ is straightforward by consistently adding in the additional term in the coefficient of V_σ. The boundary conditions and final condition are determined by the type of financial derivative $V(S, \sigma, t)$ we are solving for. The boundary conditions of any European option will depend on a prescribed exercise price, denoted here by $E > 0$. For example, in the case of the European Put Option:

$$\begin{aligned}
V(S, \sigma, T) &= \max(E - S, 0), && 0 < S < \infty,\ 0 < \sigma < \infty, \\
\lim_{S\to\infty} V(S, \sigma, t) &= 0, && 0 < \sigma < \infty,\ 0 < t < T, \\
V(0, \sigma, t) &= E\exp(-r(T - t)), && 0 < \sigma < \infty,\ 0 < t < T, \\
\lim_{\sigma\to\infty} V_\sigma(S, \sigma, t) &= 0, && 0 < S < \infty,\ 0 < t < T,
\end{aligned}$$

The remaining boundary condition at $\sigma = 0$ can be obtained by looking at the formal limit $\sigma \to 0$ in (2.2), i.e.,

$$V_t + rSV_S + \kappa\theta V_\sigma - rV = 0, \quad T > t \geq 0,\ S > 0, \text{ as } \sigma \to 0. \tag{2.3}$$

This boundary condition is used frequently, e.g. in [38, 63]. Alternatively, one can use a homogeneous Neumann condition [10], i.e.,

$$V_\sigma(S, 0, t) = 0, \quad 0 < S < \infty,\ 0 < t < T. \tag{2.4}$$

By using a change of variables:

$$x = \ln\frac{S}{E}, \quad y = \frac{\sigma}{v}, \quad \tau = T - t, \quad u = \exp(r\tau)\frac{V}{E}$$

we transform the partial differential equation to an convection-diffusion equation in two spatial dimensions with a mixed derivative term. The transformed partial differential equation and boundary/initial conditions are now satisfied by $u(x, y, \tau)$, where $x \in \mathbb{R}$, $y > 0$, $\tau \in (0, T]$:

$$u_\tau = \frac{vy}{2}u_{xx} + \frac{(vy)^{2\beta}}{2}u_{yy} + \rho(vy)^{\beta+\frac{1}{2}}u_{xy} + \left(r - \frac{vy}{2}\right)u_x + \left(\kappa\,(vy)^\alpha\,\frac{\theta - vy}{v}\right)u_y, \tag{2.5}$$

$$\begin{aligned}
u(x, y, 0) &= \max(1 - \exp(x), 0), && -\infty < x < \infty, 0 < y < \infty, && \text{(2.6a)} \\
\lim_{x\to\infty} u(x, y, \tau) &= 0, && 0 < y < \infty, 0 \leq \tau < T, && \text{(2.6b)} \\
\lim_{x\to-\infty} u(x, y, \tau) &= 1, && 0 < y < \infty, 0 \leq \tau < T, && \text{(2.6c)} \\
\lim_{y\to\infty} u_y(x, y, \tau) &= 0, && -\infty < x < \infty, 0 < \tau \leq T, && \text{(2.6d)} \\
\lim_{y\to 0} u_y(x, y, \tau) &= 0, && -\infty < x < \infty, 0 < \tau \leq T. && \text{(2.6e)}
\end{aligned}$$

In order to discretise the problem and solve numerically, we truncate our spatial boundaries to finite values. Take $L_1 \leq x \leq K_1$, where $L_1 < K_1$, and $L_2 \leq y \leq K_2$, where $0 < L_2 < K_2$,

so that the spatial domain forms a closed rectangle in $\mathbb{R}^2$ of $M \times N$ points with uniform spacing of Δ_x in the x-direction and Δ_y in the y-direction:

$$x_i = L_1 + (i-1)\Delta_x,\ i = 1, 2, \ldots, M, \qquad y_j = L_2 + (j-1)\Delta_y,\ j = 1, 2, \ldots, N.$$

The lower y-boundary is truncated to $L_2 > 0$ to ensure non-degeneracy of the partial differential equation for all values of y. We assume moderate cell aspect ratios, often we choose a square mesh with $h = \Delta_x = \Delta_y$ in our numerical experiments. We also take a uniform partition of $\tau \in [0, T]$ into P points such that $\tau_k = (k-1)\Delta_\tau$, where $k = 1, 2, \ldots, P$. We denote the discrete approximation of $u((i-1)\Delta_x, (j-1)\Delta_y, (k-1)\Delta_\tau)$ by $u^k_{i,j}$ and $U^n = (u^n_{i,j})_{i,j}$.

2.2 Hundsdorfer-Verwer ADI splitting scheme

We consider the Alternating Direction Implicit (ADI) time-stepping numerical method proposed by Hundsdorfer and Verwer [36, 37]. Our partial differential equation (2.5) takes the form $u_\tau = F(u)$. We employ the splitting $F(u) = F_0(u) + F_1(u) + F_2(u)$ where unidirectional and mixed derivative differential operators are given by:

$$F_0(u) = \rho(vy)^{\beta+\frac{1}{2}} u_{xy}, \tag{2.7a}$$

$$F_1(u) = \frac{vy}{2} u_{xx} + \left(r - \frac{vy}{2}\right) u_x, \tag{2.7b}$$

$$F_2(u) = \frac{(vy)^{2\beta}}{2} u_{yy} + \left(\kappa\,(vy)^{\alpha}\,\frac{\theta - vy}{v}\right) u_y. \tag{2.7c}$$

We consider (2.5) with the splitting (2.7) and look for a semi-discrete approximation $U^n \approx u(\tau_n)$ at time $n\Delta_\tau$. Given an approximation U^{n-1}we can calculate an approximation for U^n at time $n\Delta_\tau$ using the differential operators from (2.7):

$$Y_0 = U^{n-1} + \Delta_t F(U^{n-1}), \tag{2.8a}$$

$$Y_1 = Y_0 + \phi\Delta_t(F_1(Y_1) - F_1(U^{n-1})), \tag{2.8b}$$

$$Y_2 = Y_1 + \phi\Delta_t(F_2(Y_2) - F_2(U^{n-1})), \tag{2.8c}$$

$$\widetilde{Y}_0 = Y_0 + \psi\Delta_t(F(Y_2) - F(U^{n-1})), \tag{2.8d}$$

$$\widetilde{Y}_1 = \widetilde{Y}_0 + \phi\Delta_t(F_1(\widetilde{Y}_1) - F_1(Y_2)), \tag{2.8e}$$

$$\widetilde{Y}_2 = \widetilde{Y}_1 + \phi\Delta_t(F_2(\widetilde{Y}_2) - F_2(Y_2)), \tag{2.8f}$$

$$U^n = \widetilde{Y}_2. \tag{2.8g}$$

The parameter ψ is taken to be $\psi = 1/2$ to ensure second-order accuracy in time. The choice of ϕ is discussed in [36]. Typically it is fixed to $\phi = 1/2$. Larger values give stronger damping of the implicit terms while lower values return better accuracy. We investigate the role of ϕ further numerically in Section 2.7.

The first and fourth step in (2.8) can be solved explicitly, while the remaining steps are solved implicitly. Our aim is to derive high-order spatial discretisations of the differential

operators. Following [24] we combine high-order compact finite difference methods for the implicit steps with a (classical, non-compact) high-order stencil for the explicit steps.

The benefit of using compact stencils for the implicit steps is that the computational cost is significantly reduced when compared with the use of standard fourth-order approximations. To elaborate, the use of compact stencils yields a tri-diagonal matrix and a block tri-diagonal matrix (composed of diagonal sub-matrices) for the implicit steps solved for the x and y directions, respectively, of the ADI scheme. Inverting these relatively sparse matrices, each comprised of no more than three non-zero diagonals, can be accomplished far more efficiently than inverting the denser coefficient matrices yielded by standard fourth-order central difference approximations in space.

We are not concerned about resorting to standard fourth-order central difference approximations in space for the explicit steps of the ADI scheme, as the increased density of the coefficient matrix does not add significant computational complexity to solving the linear system explicitly.

2.3 High-order compact scheme for implicit steps

For $F_1(u)$, consider

$$F_1(u) = \frac{vy}{2}u_{xx} + \Big(r - \frac{vy}{2}\Big)u_x = g \tag{2.9}$$

with arbitrary right hand side g. We wish to derive a fourth-order accurate in space approximation for (2.9) which can be used to solve the implicit second and fifth step in (2.8). Using standard second-order central difference operators and Taylor's expansion, we have:

$$u_x(x_i, y_j) = \delta_{x0}u_{i,j} - \frac{\Delta_x^2}{6}u_{xxx}(x_i, y_j) + \mathcal{O}(\Delta_x^4) \tag{2.10}$$

$$u_{xx}(x_i, y_j) = \delta_x^2 u_{i,j} - \frac{\Delta_x^2}{12}u_{xxxx}(x_i, y_j) + \mathcal{O}(\Delta_x^4) \tag{2.11}$$

where

$$\delta_{x0}u_{i,j} = \frac{u_{i+1,j} - u_{i-1,j}}{2\Delta_x} \text{ and } \delta_x^2 u_{i,j} = \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta_x^2}.$$

If we can find second-order accurate expressions for u_{xxx} and u_{xxxx} using only information on the compact stencil, then it will be possible to approximate u_x and u_{xx} with fourth order accuracy on the compact stencil. By differentiating (2.9) once and twice with respect to x, respectively, it is possible to express u_{xxx} and u_{xxxx} in terms of first- and second-order derivatives of u and g with respect to x:

$$u_{xxx} = \frac{2}{vy}g_x + \Big(1 - \frac{2r}{vy}\Big)u_{xx}, \tag{2.12}$$

$$u_{xxxx} = \frac{2}{vy}g_{xx} + \Big(1 - \frac{2r}{vy}\Big)\Big[\frac{2}{vy}g_x + \Big(1 - \frac{2r}{vy}\Big)u_{xx}\Big]. \tag{2.13}$$

By substituting standard second-order central difference operators into (2.12) and

(2.13) we obtain second-order accurate in space approximations for u_{xxx} and u_{xxxx}:

$$u_{xxx}(x_i, y_j) = \frac{2}{vy_j}\delta_{x0}g_{i,j} + \Big(1 - \frac{2r}{vy_j}\Big)\delta_x^2 u_{i,j} + O(\Delta_x^2), \tag{2.14}$$

$$u_{xxxx}(x_i, y_j) = \frac{2}{vy_j}\delta_x^2 g_{i,j} + \Big(1 - \frac{2r}{vy_j}\Big)\Big[\frac{2}{vy_j}\delta_{x0}g_{i,j} + \Big(1 - \frac{2r}{vy_j}\Big)\delta_x^2 u_{i,j}\Big] + \mathcal{O}(\Delta_x^2). \tag{2.15}$$

Substituting (2.14) and (2.15) into (2.10) and (2.11), respectively, yields:

$$u_x(x_i, y_j) = \delta_{x0}u_{i,j} - \frac{\Delta_x^2}{6}\Big[\frac{2}{vy_j}\delta_{x0}g_{i,j} + \Big(1 - \frac{2r}{vy_j}\Big)\delta_x^2 u_{i,j}\Big] + \mathcal{O}(\Delta_x^4), \tag{2.16}$$

$$u_{xx}(x_i, y_j) = \delta_x^2 u_{i,j} - \frac{\Delta_x^2}{12}\left[\frac{2}{vy_j}\delta_x^2 g_{i,j} + \Big(1 - \frac{2r}{vy_j}\Big)\Big[\frac{2}{vy_j}\delta_{x0}g_{i,j} + \Big(1 - \frac{2r}{vy_j}\Big)\delta_x^2 u_{i,j}\Big]\right] + \mathcal{O}(\Delta_x^4). \tag{2.17}$$

Substituting these fourth-order approximations for u_x and u_{xx} into (2.9) and rearranging the equation such that all derivatives of u with respect to x are on the left hand side and all derivatives of g with respect to x are on the right hand side we obtain a fourth-order compact scheme for (2.9):

$$\frac{vy_j}{2}\left(1 - \frac{\Delta_x^2}{12}\left(\frac{2r}{vy_j} - 1\right)^2\right)\delta_x^2 u_{i,j} + \Big(r - \frac{vy_j}{2}\Big)\delta_{x0}u_{i,j} = g_{i,j} + \frac{\Delta_x^2}{12}\left(\frac{2r}{vy_j} - 1\right)\delta_{x0}g_{i,j} + \frac{\Delta_x^2}{12}\delta_x^2 g_{i,j}. \tag{2.18}$$

Finally, substituting the expressions for the difference operators δ_{x0}, δ_x^2 into (2.18) and separating the terms into values of u and g at the three horizontally adjacent nodal points in space, we get:

$$\begin{aligned}
&\frac{v^2y_j^2\Delta_x^2 - 4rvy_j\Delta_x^2 - 6v^2y_j^2\Delta_x + 4r^2\Delta_x^2 + 12rvy_j\Delta_x + 12v^2y_j^2}{24vy_j\Delta_x^2}u_{i+1,j} \\
&\qquad - \frac{v^2y_j^2\Delta_x^2 - 4rvy_j\Delta_x^2 + 4r^2\Delta_x^2 + 12v^2y_j^2}{12vy_j\Delta_x^2}u_{i,j} \\
&\quad + \frac{v^2y_j^2\Delta_x^2 - 4rvy_j\Delta_x^2 + 6v^2y_j^2\delta_x + 4r^2\Delta_x^2 - 12rvy_j\Delta_x + 12v^2y_j^2}{24vy_j\Delta_x^2}u_{i-1,j} \\
&= \frac{-vy_j\Delta_x + 2r\Delta_x + 2vy_j}{24vy_j}g_{i+1,j} + \frac{5}{6}g_{i,j} - \frac{-vy_j\Delta_x + 2r\Delta_x - 2vy_j}{24vy_j}g_{i-1,j}
\end{aligned} \tag{2.19}$$

The above derivation can be presented systematically also in the following concise form. Considering the convection diffusion equation

$$u_{xx} + c_1 u_x = c_2 g \tag{2.20}$$

with constants c_1 and c_2, the necessary relations derived explicitly above can be concisely written in matrix form

$$\begin{bmatrix} 1 & 0 & \frac{1}{6} & 0 \\ 0 & 1 & 0 & \frac{1}{12} \\ 0 & c_1\Delta_x^2 & 1 & 0 \\ 0 & 0 & c_1 & 1 \end{bmatrix} \cdot \begin{bmatrix} u_x \\ u_{xx} \\ \Delta_x^2 u_{xxx} \\ \Delta_x^2 u_{xxxx} \end{bmatrix} = \begin{bmatrix} \delta_{x0}u_{i,j} \\ \delta_x^2 u_{i,j} \\ c_2\Delta_x^2 g_x \\ c_2\Delta_x^2 g_{xx} \end{bmatrix} + \mathcal{O}(\Delta_x^4) \begin{bmatrix} 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} \delta_{x0}u_{i,j} \\ \delta_x^2 u_{i,j} \\ c_2\Delta_x^2\delta_{x0}g_{i,j} \\ c_2\Delta_x^2\delta_x^2 g_{i,j} \end{bmatrix} + \mathcal{O}(\Delta_x^4),$$

where the first two lines of the system correspond to (2.10) and (2.11), while the third and fourth are obtained from the repeated differentiation of (2.20). This also shows that only second-order approximations for u_x, u_{xx}, g_x and g_{xx} are needed. We are grateful to an anonymous referee for drawing our attention to this form of presenting the derivation of the scheme. Equation (2.19) defines a fourth-order compact approximation for (2.9). In other words, we have a system of equations which defines a fourth-order accurate approximation for (2.9) at any point on the inner grid of the spatial domain (all points of the spatial domain except those that lie on the x and y boundaries). To approximate (2.9) at points along the x boundaries of the inner grid of the spatial domain, we will require a contribution from the Dirichlet values at the x-boundaries of the spatial domain. We collect these separately in a vector d. Details on the boundary conditions are given in Section 2.6. The linear system to be solved can be written in matrix form:

$$A_x u = B_x g + d,$$

where $u = (u_{2,2}, u_{2,3}, \ldots, u_{M-1,N-1})$, $g = (g_{2,2}, g_{2,3}, \ldots, g_{M-1,N-1})$. The coefficient matrices A_x and B_x are block diagonal matrices, with the following structure:

$$A_x = \begin{bmatrix} A_x^{1,1} & 0 & 0 & 0 \\ 0 & A_x^{2,2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & A_x^{N-2,N-2} \end{bmatrix}, \quad B_x = \begin{bmatrix} B_x^{1,1} & 0 & 0 & 0 \\ 0 & B_x^{2,2} & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & B_x^{N-2,N-2} \end{bmatrix},$$

where each $A_x^{j,j} = \mathrm{diag}[a_{-1}^{j,j}, a_0^{j,j}, a_1^{j,j}]$ and $B_x^{j,j} = \mathrm{diag}[b_{-1}^{j,j}, b_0^{j,j}, b_1^{j,j}]$ are tri-diagonal matrices.

Let us consider now the case of F_2:

$$F_2(u) = \frac{(vy)^{2\beta}}{2} u_{yy} + \left(\kappa \, (vy)^{\alpha} \, \frac{\theta - vy}{v} \right) u_y = g. \tag{2.21}$$

Due to the appearance of y terms in the coefficients of $F_2(u)$, the algebraic complexity in deriving a fourth-order accurate scheme in space is much greater. By Taylor's expansions we obtain:

$$u_y \, (x_i, y_j) = \delta_{y_0} u_{i,j} - \frac{{\Delta_y}^2}{6} u_{yyy} \, (x_i, y_j) + \mathcal{O}(\Delta_y^4), \tag{2.22}$$

$$u_{yy} \, (x_i, y_j) = \delta_y^2 u_{i,j} - \frac{{\Delta_y}^2}{12} u_{yyyy} \, (x_i, y_j) + \mathcal{O}(\Delta_y^4). \tag{2.23}$$

We wish to find second order accurate approximations for u_{yyy} and u_{yyyy} on the compact stencil in order to find fourth-order accurate expressions for u_y and u_{yy}. Re-arranging (2.21), we get:

$$u_{yy} = \frac{2}{(vy)^{2\beta}} \left(- \left(\kappa \, (vy)^{\alpha} \, \frac{\theta - vy}{v} \right) u_y + g \right).$$

Via repeated applications of the chain rule, second-order accurate approximations for

$u_{yyy}(x_i, y_j)$ and $u_{yyyy}(x_i, y_j)$ are given by:

$$u_{yyy}(x_i, y_j) = \frac{(2\,(vy_j)^{\alpha}\,\alpha\kappa v y_j - 2\,(vy_j)^{\alpha}\,\theta\alpha\kappa + 2\,(vy_j)^{\alpha}\,\kappa v y_j)}{(vy_j)^{2\beta}\,vy_j}\delta_{y0}u_{i,j}$$
$$+ \frac{(2\,(vy_j)^{\alpha}\,\kappa v {y_j}^2 - 2\,(vy_j)^{2\beta}\,\beta v - 2\,(vy_j)^{\alpha}\,\theta\kappa y_j)}{(vy_j)^{2\beta}\,vy_j}\delta_y^2 u_{i,j} + \frac{2}{(vy_j)^{2\beta}}\delta_{y0}g_{i,j} + \mathcal{O}(\Delta_y^2), \quad (2.24)$$

$$u_{yyyy}(x_i, y_j) = \left(\frac{2(2\,(vy_j)^{\alpha}\,\kappa v {y_j}^2 - 2\,(vy_j)^{2\beta}\,\beta v - 2\,(vy_j)^{\alpha}\,\theta\kappa y_j)}{(vy_j)^{4\beta}\,vy_j} - \frac{4\beta}{(vy_j)^{2\beta}\,y_j}\right)\delta_{y0}g_{i,j}$$
$$+ \left(\frac{1}{(vy_j)^{2\beta}\,vy_j}(2\,(vy_j)^{\alpha}\,\alpha^2\kappa v + 4\,(vy_j)^{\alpha}\,\alpha\kappa v - \frac{2\,(vy_j)^{\alpha}\,\alpha^2\theta\kappa}{y_j} + 2\,(vy_j)^{\alpha}\,\kappa v)\right.$$
$$- \frac{2\beta\,(2\,(vy_j)^{\alpha}\,\alpha\kappa v y_j - 2\,(vy_j)^{\alpha}\,\theta\alpha\kappa + 2\,(vy_j)^{\alpha}\,\kappa v y_j)}{(vy_j)^{2\beta}\,v{y_j}^2}$$
$$- \frac{2\,(vy_j)^{\alpha}\,\alpha\kappa v y_j - 2\,(vy_j)^{\alpha}\,\theta\alpha\kappa + 2\,(vy_j)^{\alpha}\,\kappa v y_j}{(vy_j)^{2\beta}\,v{y_j}^2}$$
$$+ \frac{(2\,(vy_j)^{\alpha}\,\kappa v {y_j}^2 - 2\,(vy_j)^{2\beta}\,\beta v - 2\,(vy_j)^{\alpha}\,\theta\kappa y_j)}{((vy_j)^{4\beta}\,v^2{y_j}^2)^{\frac{1}{2}}}$$
$$\left.\frac{(2\,(vy_j)^{\alpha}\,\alpha\kappa v y_j - 2\,(vy_j)^{\alpha}\,\theta\alpha\kappa + 2\,(vy_j)^{\alpha}\,\kappa v y_j)}{((vy_j)^{4\beta}\,v^2{y_j}^2)^{\frac{1}{2}}}\right)\delta_{y0}u_{i,j}$$
$$+ \left(\frac{2\,(vy_j)^{\alpha}\,\alpha\kappa v y_j - 2\,(vy_j)^{\alpha}\,\theta\alpha\kappa + 2\,(vy_j)^{\alpha}\,\kappa v y_j}{(vy_j)^{2\beta}\,vy_j}\right.$$
$$+ \frac{1}{(vy_j)^{2\beta}\,vy_j}(2\,(vy_j)^{\alpha}\,\alpha\kappa v y_j + 4\,(vy_j)^{\alpha}\,\kappa v y_j - 4\frac{(vy_j)^{2\beta}\,\beta^2 v}{y_j} - 2\,(vy_j)^{\alpha}\,\theta\alpha\kappa$$
$$- 2\,(vy_j)^{\alpha}\,\theta\kappa) - \frac{2\beta(2\,(vy_j)^{\alpha}\,\kappa v {y_j}^2 - 2\,(vy_j)^{2\beta}\,\beta v - 2\,(vy_j)^{\alpha}\,\theta\kappa y_j)}{(vy_j)^{2\beta}\,v{y_j}^2}$$
$$- \frac{2\,(vy_j)^{\alpha}\,\kappa v {y_j}^2 - 2\,(vy_j)^{2\beta}\,\beta v - 2\,(vy_j)^{\alpha}\,\theta\kappa y_j}{(vy_j)^{2\beta}\,v{y_j}^2}$$
$$\left.+ \frac{(2\,(vy_j)^{\alpha}\,\kappa v {y_j}^2 - 2\,(vy_j)^{2\beta}\,\beta v - 2\,(vy_j)^{\alpha}\,\theta\kappa y_j)^2}{(vy_j)^{4\beta}\,v^2{y_j}^2}\right)\delta_y^2 u_{i,j}$$
$$+ \frac{2}{(vy_j)^{2\beta}}\delta_y^2 g_{i,j} + \mathcal{O}(\Delta_y^2). \quad (2.25)$$

where δ_{y0} and δ_y^2 denote the standard second-order central difference operators.

Substituting (2.24) and (2.25) into (2.22) and (2.23), respectively, yields fourth-order accurate approximations (not given here) for $u_y(x_i, y_j)$ and $u_{yy}(x_i, y_j)$ on the compact stencil. By substituting these fourth-order accurate approximations into (2.21) and separating the u and g terms onto the left and right hand sides, respectively, we obtain a linear system which can be represented in matrix form:

$$A_y u = B_y g$$

where $u = (u_{2,2}, u_{2,3}, \ldots, u_{M-1,N-1})$, $g = (g_{2,2}, g_{2,3}, \ldots, g_{M-1,N-1})$. We do not impose any boundary conditions in y-direction, but discretise the boundary grid points with the same

scheme, and handle resulting ghost points via extrapolation; details on the boundary conditions are given in Section 2.6. The coefficient matrices A_y and B_y are block tri-diagonal matrices with the following structures:

$$\mathbf{A_y} = \begin{bmatrix} A_y^{1,1} & A_y^{1,2} & 0 & 0 & 0 \\ A_y^{2,1} & A_y^{2,2} & A_y^{2,3} & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & A_y^{N-3,N-4} & A_y^{N-3,N-3} & A_y^{N-3,N-2} \\ 0 & 0 & 0 & A_y^{N-2,N-3} & A_y^{N-2,N-2} \end{bmatrix},$$

$$\mathbf{B_y} = \begin{bmatrix} B_y^{1,1} & B_y^{1,2} & 0 & 0 & 0 \\ B_y^{2,1} & B_y^{2,2} & B_y^{2,3} & 0 & 0 \\ 0 & \ddots & \ddots & \ddots & 0 \\ 0 & 0 & B_y^{N-3,N-4} & B_y^{N-3,N-3} & B_y^{N-3,N-2} \\ 0 & 0 & 0 & B_y^{N-2,N-3} & B_y^{N-2,N-2} \end{bmatrix},$$

where each $A_y^{j,j} = \mathrm{diag}[a^{i,j}]$ and $B_y^{j,j} = \mathrm{diag}[b^{i,j}]$ are diagonal matrices, with values on these diagonals given as follows:

$$\begin{aligned} a^{i,j\pm1} = {} & \frac{1}{2{\Delta_y}^2}(vy_j)^{2\beta} - \frac{1}{12(vy_j)^{2\beta}v^2{y_j}^2}\Big(- 2(vy_j)^{2\alpha}\kappa^2v^2{y_j}^4 + 2(vy_j)^{2\beta+\alpha}\alpha\kappa v^2{y_j}^2 \\ & - 2(vy_j)^{2\beta+\alpha}\beta\kappa v^2{y_j}^2 + 4(vy_j)^{2\alpha}\theta k^2v{y_j}^3 + 2(vy_j)^{4\beta}\beta^2v^2 - 2(vy_j)^{2\beta+\alpha}\theta\alpha\kappa vy_j \\ & + 2(vy_j)^{2\beta+\alpha}\theta\beta\kappa vy_j + 2(vy_j)^{2\beta+\alpha}\kappa v^2{y_j}^2 - 2(vy_j)^{2\alpha}\theta^2\kappa^2{y_j}^2 + (vy_j)^{4\beta}\beta v^2\Big) \\ & \pm \Bigg(\frac{-(vy_j)^{\alpha}\kappa vy_j + (vy_j)^{\alpha}\theta\kappa}{2v\Delta_y} - \frac{1}{24\Delta_y\beta^2(vy_j)^4}\Big(- 2(vy_j)^{2\alpha}\alpha\kappa^2v^2{y_j}^3{\Delta_y}^2 \\ & + (vy_j)^{2\beta}(vy_j)^{\alpha}\alpha^2\kappa v^2y_j{\Delta_y}^2 - 4(vy_j)^{2\beta+\alpha}\alpha\beta\kappa v^2y_j{\Delta_y}^2 + 4(vy_j)^{2\alpha}\theta\alpha\kappa^2v{y_j}^2{\Delta_y}^2 \\ & - 2(vy_j)^{2\alpha}\kappa^2v^2{y_j}^3{\Delta_y}^2 - (vy_j)^{2\beta+\alpha}\theta\alpha^2\kappa v{\Delta_y}^2 + 4(vy_j)^{2\beta+\alpha}\theta\alpha\beta\kappa v{\Delta_y}^2 \\ & + (vy_j)^{2\beta+\alpha}\alpha\kappa v^2y_j{\Delta_y}^2 - 4(vy_j)^{2\beta+\alpha}\beta\kappa v^2y_j{\Delta_y}^2 - 2(vy_j)^{2\alpha}\theta^2\alpha\kappa^2y_j{\Delta_y}^2 \\ & + 2(vy_j)^{2\alpha}\theta\kappa^2v{y_j}^2{\Delta_y}^2 + (vy_j)^{2\beta+\alpha}\theta\alpha\kappa v{\Delta_y}^2\Big)\Bigg), \end{aligned} \tag{2.26}$$

$$\begin{aligned} a^{i,j} = {} & \frac{1}{6(vy_j)^{2\beta+2}}\Big(- 2(vy_j)^{2\alpha}k^2v^2{y_j}^4 + 2(vy_j)^{2\beta+\alpha}\alpha kv^2{y_j}^2 + 2(vy_j)^{4\beta}\beta^2v^2 \\ & - 2(vy_j)^{2\beta+\alpha}\beta kv^2{y_j}^2 + 4(vy_j)^{2\alpha}\theta k^2v{y_j}^3 - 2(vy_j)^{2\beta+\alpha}\theta\alpha kvy_j + (vy_j)^{4\beta}\beta v^2 \\ & + 2(vy_j)^{2\beta+\alpha}\theta\beta kvy_j + 2(vy_j)^{2\beta+\alpha}kv^2{y_j}^2 - 2(vy_j)^{2\alpha}\theta^2k^2{y_j}^2 - 6(vy_j)^{4\beta+2}\Big), \end{aligned} \tag{2.27}$$

$$b^{i,j\pm1} = \pm\frac{-2(vy_j)^{\alpha}kv^2{y_j}^3{\Delta_y}^2 - 4(vy_j)^{2\beta}\beta v^2y_j{\Delta_y}^2 + 2(vy_j)^{\alpha}\theta kvy_j{}^2{\Delta_y}^2}{24(vy_j)^{2\beta+2}\Delta_y} + \frac{1}{12}, \quad b^{i,j} = \frac{5}{6}. \tag{2.28}$$

For F_2 the derivation can be presented in a concise form, similar as in Remark 2.3 for F_1. Considering the convection diffusion equation

$$u_{yy} + c_1u_y = c_2g \tag{2.29}$$

with $c_1 = c_1(y)$ and $c_2 = c_2(y)$, the necessary relations can be concisely written in matrix form

$$\begin{bmatrix} 1 & 0 & \frac{1}{6} & 0 \\ 0 & 1 & 0 & \frac{1}{12} \\ c_1'\Delta_y^2 & c_1\Delta_y^2 & 1 & 0 \\ c_1''\Delta_y^2 & 2c_1'\Delta_y^2 & c_1 & 1 \end{bmatrix} \cdot \begin{bmatrix} u_y \\ u_{yy} \\ \Delta_y^2 u_{yyy} \\ \Delta_y^2 u_{yyyy} \end{bmatrix} = \begin{bmatrix} \delta_{y0} u_{i,j} \\ \delta_y^2 u_{i,j} \\ \Delta_y^2(\delta_{y0} c_{2,j} g_{i,j} + c_{2,j}\delta_{y0} g_{i,j}) \\ \Delta_y^2(\delta_y^2 c_{2,j} g_{i,j} + 2\delta_{y0} c_{2,j}\delta_{y0} g_{i,j} + c_{2,j}\delta_y^2 g_{i,j}) \end{bmatrix} + \mathcal{O}(\Delta_y^4), \quad (2.30)$$

where the first two lines of the system correspond to (2.22) and (2.23), while the third and fourth are obtained from the repeated differentiation of (2.29).

2.4 High-order scheme for explicit steps

The first and fourth steps of the ADI scheme (2.8) operate only on previous approximations to explicitly calculate an updated approximation. The differential operator in these steps takes the form of the right hand side of (2.5). For the mixed derivative term it seems not to be possible to exploit the structure of the differential operator to obtain a fourth-order approximation on a compact computational stencil. Hence, in order to maintain fourth-order accuracy of the scheme in the explicit steps of (2.8), the derivatives in each differential operator F_0, F_1 and F_2 are approximated using classical, fourth-order central difference operators which operate on a larger 5×5-stencil in the spatial domain. Here we use the shift operator defined by:

$$s_x = e^{\Delta_x \delta_x} \text{ where } (s_x u)_{i,j} = u_{i+1,j}, \quad s_y = e^{\Delta_y \delta_y} \text{ where } (s_y u)_{i,j} = u_{i,j+1}.$$

For $F_1(u) = \frac{vy}{2} u_{xx} - (\frac{vy}{2} - r) u_x$, we have the following scheme:

$$\Big[\frac{vy}{2} u_{xx} + \Big(r - \frac{vy}{2}\Big) u_x\Big]_{i,j} = \frac{vy_j}{2} \left(\frac{-s_x^{-2} + 16 s_x^{-1} - 30 + 16 s_x - s_x^2}{12\Delta_x^2}\right) u_{i,j} + \Big(r - \frac{vy_j}{2}\Big) \left(\frac{s_x^{-2} - 8 s_x^{-1} + 8 s_x - s_x^2}{12\Delta_x}\right) u_{i,j} + \mathcal{O}(\Delta_x^4).$$

For $F_2(u) = \frac{(vy)^{2\beta}}{2} u_{yy} + \frac{\kappa (vy)^\alpha (\theta - vy)}{v} u_y$, we have:

$$\Big[\frac{(vy)^{2\beta}}{2} u_{yy} + \frac{\kappa (vy)^\alpha (\theta - vy)}{v} u_y\Big]_{i,j} = \frac{(vy_j)^{2\beta}}{2} \left(\frac{-s_y^{-2} + 16 s_y^{-1} - 30 + 16 s_y - s_y^2}{12\Delta_y^2}\right) u_{i,j} + \frac{\kappa (vy_j)^\alpha (\theta - vy_j)}{v} \left(\frac{s_y^{-2} - 8 s_y^{-1} + 8 s_y - s_y^2}{12\Delta_y}\right) u_{i,j} + \mathcal{O}(\Delta_y^4).$$

Finally, for the mixed derivative term $F_0 = \rho (vy)^{\beta + \frac{1}{2}} u_{xy}$, the following computational stencil is used:

$$\Big[\rho (vy)^{\beta+\frac{1}{2}} u_{xy}\Big]_{i,j} = \rho (vy_j)^{\beta+\frac{1}{2}} \left(\frac{s_x^{-2} - 8 s_x^{-1} + 8 s_x - s_x^2}{12\Delta_x}\right) \left(\frac{s_y^{-2} - 8 s_y^{-1} + 8 s_y - s_y^2}{12\Delta_y}\right) u_{i,j} + \mathcal{O}(\Delta_x^4 \Delta_y^4) + \mathcal{O}(\Delta_x^4) + \mathcal{O}(\Delta_y^4).$$

•	$u_{4,1}$	$u_{4,2}$	$u_{4,3}$	$u_{4,4}$
•	$u_{3,1}$	$u_{3,2}$	$u_{3,3}$	$u_{3,4}$
•	$u_{2,1}$	$u_{2,2}$	$u_{2,3}$	$u_{2,4}$
•	$u_{1,1}$	$u_{1,2}$	$u_{1,3}$	$u_{1,4}$
⊙	○	○	○	○

Figure 2.1: Example: evaluation of $F(u_{2,2})$ using the 5×5-point computational stencil in the lower left corner of the computational domain; ghost points outside the computational domain at which values are extrapolated from the interior of the domain are marked by bullets (•,○,⊙), grid points on the boundary are set in Roman.

Using these fourth-order approximations, the first and fourth step in (2.8) can be computed directly. The values at the spatial boundaries for each solution of the ADI scheme are determined by the boundary conditions, the computational stencil is required for all remaining points in the spatial domain. For the explicit steps, the 5×5-point computational stencil exceeds the spatial boundary when we wish to approximate differential operator $F(u)$ at any point along the boundary of the spatial domain's inner grid. For example if we wish to evaluate $F(u_{2,2})$, we will require contributions from ghost points which fall outside the spatial domain, as marked by bullet points in Figure 2.1. We extrapolate information from grid points $u(x_i, y_j)$, where $i = 1, \ldots, M-1$, $j = 1, \ldots, N-1$ to establish values at these ghost points for the purpose of evaluating the differential operator $F(u)$ at any point along the boundary of the inner grid of the spatial domain. To calculate the values at these ghost points, we use the following five-point extrapolation formulae for three cases:

$$
\begin{aligned}
x = L_1 \text{ boundary } (\bullet): \quad & u_{i,0} = 5u_{i,1} - 10u_{i,2} + 10u_{i,3} - 5u_{i,4} + u_{i,5} + \mathcal{O}(\Delta_x^5), \\
y = L_2 \text{ boundary } (\circ): \quad & u_{0,j} = 5u_{1,j} - 10u_{2,j} + 10u_{3,j} - 5u_{4,j} + u_{5,j} + \mathcal{O}(\Delta_y^5), \\
x = L_1, y = L_2 \text{ corner } (\odot): \quad & u_{0,0} = 5u_{1,1} - 10u_{2,2} + 10u_{3,3} - 5u_{4,4} + u_{5,5} + \mathcal{O}(\Delta_x^5) \\
& \quad + \mathcal{O}(\Delta_x^4 \Delta_y) + \mathcal{O}(\Delta_x^3 \Delta_y^2) + \mathcal{O}(\Delta_x^2 \Delta_y^3) + \mathcal{O}(\Delta_x \Delta_y^4) \\
& \quad + \mathcal{O}(\Delta_y^5).
\end{aligned}
$$

The extrapolation at the $x = K_1$ and $y = K_2$ boundaries and the remaining three corners is handled analogously.

2.5 Solving the high-order ADI scheme

Starting from a given U^{n-1}, the ADI scheme (2.8) involves six approximation steps to obtain U^n, the solution at the next time level. The first approximation Y_0 can be solved for explicitly using the 5×5-point computational stencil derived in Section 2.4. The second approximation for our solution, denoted by Y_1, has to be solved for implicitly:

$$
Y_1 = Y_0 + \phi \Delta_t (F_1(Y_1) - F_1(U^{n-1})) \quad \Longleftrightarrow \quad F_1(Y_1 - U^{n-1}) = \frac{1}{\phi \Delta_t}(Y_1 - Y_0). \tag{2.31}
$$

We apply the fourth-order compact scheme established in Section 2.3 to solve (2.31). In matrix form we obtain

$$A_x(Y_1 - U^{n-1}) = B_x\Big(\frac{1}{\phi\Delta_t}(Y_1 - Y_0)\Big) + d.$$

Collecting unknown Y_1 terms on the left hand side and known terms Y_0, U^{n-1} and d on the right hand side we get

$$(B_x - \phi\Delta_t A_x)\, Y_1 = B_x Y_0 - \phi\Delta_t A_x U^{n-1} - \phi\Delta_t d.$$

To solve, we invert the tri-diagonal matrix $(B_x - \phi\Delta_t A_x)$. For the third step of the ADI scheme, we proceed analogously, and use the the high-order compact scheme presented in Section 2.3 to solve for Y_2 implicitly. The fourth, fifth and sixth step of the ADI scheme are performed analogously as the first, second and third steps, respectively.

Note that the matrix $(B_x - \phi\Delta_t A_x)$ appears twice in the scheme (2.8), in the second and fifth step. Similarly, $(B_y - \phi\Delta_t A_y)$ appears in the third and the sixth step. Hence, using LU-factorisation, only two matrix inversions are necessary in each time step of scheme (2.8). Moreover, since the coefficients in the partial differential equation (2.5) do not depend on time, and the matrices are therefore constant, they can be LU-factorised before iterating in time to obtain a highly efficient algorithm.

The combination of the fourth-order spatial discretisation presented in Section 2.3 and 2.4 with the second-order time splitting (2.8) yields a high-order ADI scheme with order of consistency two in time and four in space.

2.6 Boundary conditions

For the case of the Dirichlet conditions at $x = L_1$ and $x = K_1$ we impose

$$\begin{aligned} u(L_1, y_j, \tau_k) &= 1 - e^{r\tau + L_1}, && j = 1, 2, \ldots, N,\ k = 1, 2, \ldots, \\ u(K_1, y_j, \tau_k) &= 0, && j = 1, 2, \ldots, N,\ k = 1, 2, \ldots. \end{aligned}$$

Using the homogeneous Neumann conditions (2.6d) and (2.6e) which are correct in the limit $y \to \infty$ and $y \to 0$, respectively, at the (finite) boundaries $y = L_2 > 0$ and $y = K_2$ would result in a dominant error along these boundaries. Hence, we do not impose any boundary condition at these two boundaries but discretise the partial differential equation using the computational stencil from the interior. The values of the unknown on the boundaries are set by extrapolation from values in the interior. This introduces a numerical error, and it needs to be considered that the order of extrapolation should be high enough not to affect the overall order of accuracy. We refer to Gustafsson [29] to discuss the influence of the order of the approximation on the global convergence rate. We use the following extrapolation formulae:

$$\begin{aligned} u^k_{i,1} &= 5u^k_{i,2} - 10u^k_{i,3} + 10u^k_{i,4} - 5u^k_{i,5} + u^k_{i,6} + \mathcal{O}(\Delta_y^6), \\ u^k_{i,N} &= 5u^k_{i,N-1} - 10u^k_{i,N-2} + 10u^k_{i,N-3} - 5u^k_{i,N-4} + u^k_{i,N-5} + \mathcal{O}(\Delta_y^6). \end{aligned}$$

2.7 Numerical experiments

In this section the results of numerical experiments are discussed. The numerical convergence order of the high-order ADI scheme is covered as well as additional experiments aimed at validating its stability as well as investigating the role of the parameter ϕ.

Numerical convergence

This section of results covers the numerical study to compute the order of convergence of the high-order ADI scheme. For convenience, an equally sized space step $h = \Delta_x = \Delta_y$, has been chosen, creating an evenly-spaced mesh both horizontally and vertically. The parameter-value $\phi = 0.5$ in (2.8) is chosen for the numerical experiments unless mentioned otherwise.

Since the initial condition for the option pricing problem, the payoff function $V(S, \sigma, T)$, is non-smooth at $S = E$, we cannot in general expect to observe high-order convergence [40]. Some form of smoothing has to be applied to the initial condition.

A practical approach would be to carefully choose a mesh where no node coincides with the point in the domain where the non-smooth point exists, [41] investigates the effect of non-smooth payoffs in option pricing and empirically studies the approach of shifting the mesh to preserve the order of convergence of second order implicit finite difference schemes. Employing this method requires constructing the mesh such that the distance between the non-smooth point in the payoff function and the nearest adjacent nodes on the mesh should be equal. This placement of the mesh serves to artificially smooth the payoff, thus significantly reducing the associated error around this point. The advantage of this approach is the ease with which it can be implemented for any given mesh. The disadvantage of this approach is that for various refinements of the mesh, the mesh should be re-aligned such that the distance between the point on the domain where there is a non-smooth point in the payoff, and the two adjacent nodes should be equal. This lack of flexibility means that we are unable to price options consistently at given points on the mesh at various refinements without additional interpolation.

An alternative approach, which allows us to choose our mesh with a greater flexibility, is proposed in [40] where suitable smoothing operators are identified in Fourier space. Since the order of convergence of our high-order compact scheme is four, we could use the smoothing operator Φ_4 as in [26], given by its Fourier transformation

$$\hat{\Phi}_4(\omega) = \left(\frac{\sin\left(\frac{\omega}{2}\right)}{\frac{\omega}{2}} \right)^4 \left[1 + \frac{2}{3} \sin^2\left(\frac{\omega}{2}\right) \right].$$

This leads to the smooth initial condition determined by

$$\tilde{u}_0(x, y) = \int_{-3h}^{3h} \int_{-3h}^{3h} \Phi_4\left(\frac{\tilde{x}}{h}\right) \Phi_4\left(\frac{\tilde{y}}{h}\right) u_0(x - \tilde{x}, y - \tilde{y}) \; d\tilde{x}\, d\tilde{y}$$

for any stepsize $h > 0$, where u_0 is the original initial condition and $\Phi_4(x)$ denotes the Fourier inverse of $\hat{\Phi}_4(\omega)$, see [40]. As $h \to 0$, the smooth initial condition $\tilde{u}_0$ tends towards

the original initial condition u_0 and the approximation of the smoothed problem tends towards the true solution. We have chosen to adopt this smoothing operator for our numerical experiments.

In order to investigate the numerical convergence of the scheme, we fix the parabolic mesh ratio to $\gamma = \Delta_t/h^2$, which is a natural value for parabolic partial differential equations such as (2.5). Given that we have chosen a parabolic mesh ratio where the time step is of the same order as the square of the space step, we might have considered an implementation of Euler's second-order in space explicit finite difference method for solving the PDE. The stringent stability constraint of the explicit scheme requires the time step to be of the order of the square of the spatial step or less. While an explicit method would be simpler to implement and less computationally expensive, the ADI scheme we have employed is unconditionally stable by comparison and achieves a higher order of accuracy in time. Implementing a stable explicit scheme with the same degree of accuracy would requires us to solve the PDE over a significantly finer mesh.

Figure 2.2 shows the numerical solution for the European option price at time $T = 0.5$ using the parameters from Table 2.1. In order to evaluate the high order of convergence

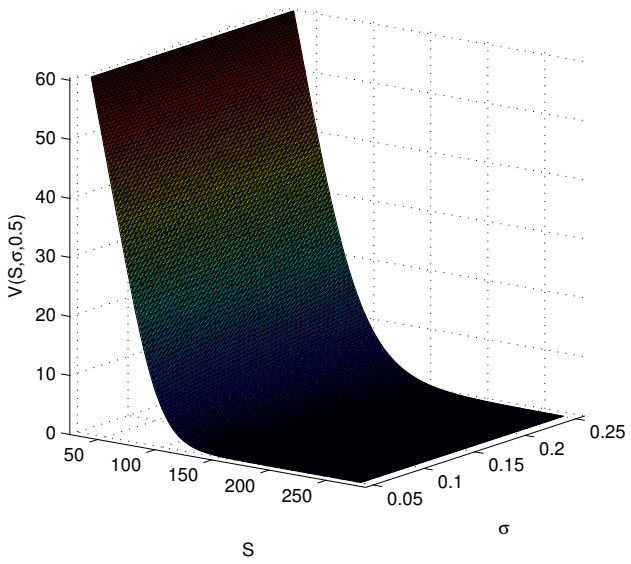

Figure 2.2: Numerical solution for price of European Put Option for default parameters.

in space for the accuracy of the scheme, we compute the l_2-norm and the l_∞ norm of the relative error for consecutive approximations for the option price at a given expiry over refined spatial meshes. To elaborate, we compute a reference solution, $C_0(x_{0_i}, y_{0_j})$, over a uniform 385×385 spatial mesh defined by:

$$x_{0_i} = -1 + \frac{i-1}{384}, \quad y_{0_j} = \frac{1}{2} + \frac{j-1}{384}, \quad i, j \in \{1, 2, \ldots, 384, 385\}. \tag{2.33}$$

The discretization in time is determined by the parabolic mesh ratio, and the final

Parameter	Value
Strike price	$E = 100$
Time to maturity	$T = 0.5$
Interest rate	$r = 0.05$
Volatility of volatility	$v = 0.1$
Mean reversion speed	$\kappa = 2$
Long run mean of volatility	$\theta = 0.1$
Correlation	$\rho = -0.5$
Parabolic mesh ratio	$\gamma = 0.5$
Stochastic volatility drift parameter	$\alpha = 0$
Stochastic volatility diffusion parameter	$\beta = 0.5$
x-range ($[L_1 - K_1]$)	$x \in [-1, 1]$
y-range ($[L_2 - K_2]$)	$y \in [0.5, 2.5]$

Table 2.1: Default input parameters for numerical experiments.

time step should be adapted such that we are able to evaluate the solution at the desired expiry ($T = 1$ in our case). Once this result is computed, we generate a refined spatial mesh for computing the next numerical approximation, $C_1(x_{1_i}, y_{1_j})$, by removing every other point from the spatial mesh of the previous approximation:

$$x_{1_i} = x_{0_{2i-1}}, \quad y_{1_j} = y_{0_{2j-1}} \quad i, j \in \{1, 2, \ldots, 195\}. \tag{2.35}$$

We compute the l_2 and l_∞ norms of the relative error for the two approximations as follows:

$$l_2(C_0, C_1) = h(\sum_{i=1}^{195}\sum_{j=1}^{195}(C_0(x_{0_{2i-1}}, y_{0_{2j-1}}) - C_1(x_{1_i}, y_{1_j}))^2)^{\frac{1}{2}}, \tag{2.36}$$

$$l_\infty(C_0, C_1) = \max_{i,j\in\{1,\ldots,195\}} |C_0(x_{0_{2i-1}}, y_{0_{2j-1}}) - C_1(x_{1_i}, y_{1_j})|, \tag{2.37}$$

where $h = \Delta_x = \Delta_y$ is the uniform mesh spacing for the coarser discretization in space. We utilize the same mesh refinement technique to approximate additional solutions over coarser spatial meshes, and compute the above norms for each consecutive pair of solutions. This process can be repeated up until it yields a 6×6 spatial discretization, producing a total of six values each for the error norms, which sufficiently allows us to approximate the order of the scheme in space.

Asymptotically, we expect these norms to converge as $\varepsilon = Ch^m$ for some m and C representing constants. This implies $\ln(\varepsilon) = \ln(C) + m\ln(h)$. Hence, the double-logarithmic plot ε against h should be asymptotic to a straight line with slope m. This gives a method for experimentally determining the order of the scheme. The expected numerical convergence rate is approximately order $\mathcal{O}(h^4)$ in space. For comparison, additional experiments

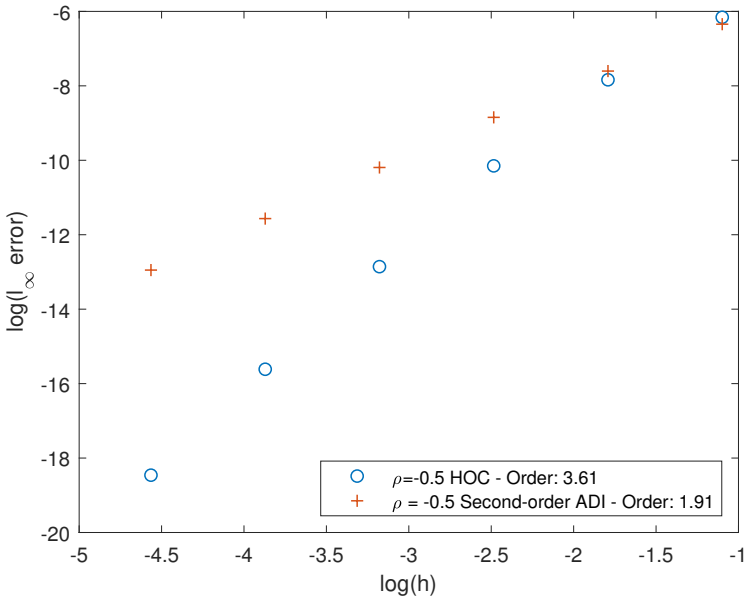

Figure 2.3: l_∞-error comparison of the high-order ADI scheme with standard second-order in space ADI scheme for $\rho = -0.5$. We observed similar results for $\rho = -0.2$ and $\rho = 0.1$.

have been conducted using a standard, second-order ADI scheme based on (2.8) combined with a second-order central difference discretisation in space. Figure 2.3 shows the double logarithmic plot of l_∞-error versus space step h. The observed numerical convergence order here is sufficiently close to the theoretical fourth-order accuracy of the scheme in space. The high-order ADI scheme significantly outperforms the standard second-order ADI scheme in almost all cases for a given mesh width h. In other words, to realise a chosen level of accuracy one could use a coarser grid for the high-order ADI scheme than the standard second-order scheme which translates into solving smaller linear systems, which is more computationally efficient.

Numerical stability analysis

In this section we investigate whether there are any stability restrictions on the choice of the time-step Δ_t for the high-order ADI scheme. Unlike for standard second-order schemes, the algebraic complexity of the numerical stability analysis of high-order compact schemes is very high since the established stability notions imply formidable algebraic problems for high-order compact schemes. As a result, there are only few stability results for high-order compact schemes in the literature [26, 23, 27]. This is even more pronounced in higher spatial dimensions, as most of the existing studies with analytical stability results for high-order compact schemes are limited to a one-dimensional setting.

For diffusion equations (without convection) with mixed derivative terms and constant coefficients, a stability analysis of the ADI method (2.8) with standard second-order discretisation in space [35] revealed it to be unconditionally stable. The analysis in [35] is based on studying the stability for a simplified, linear test equation which implies the

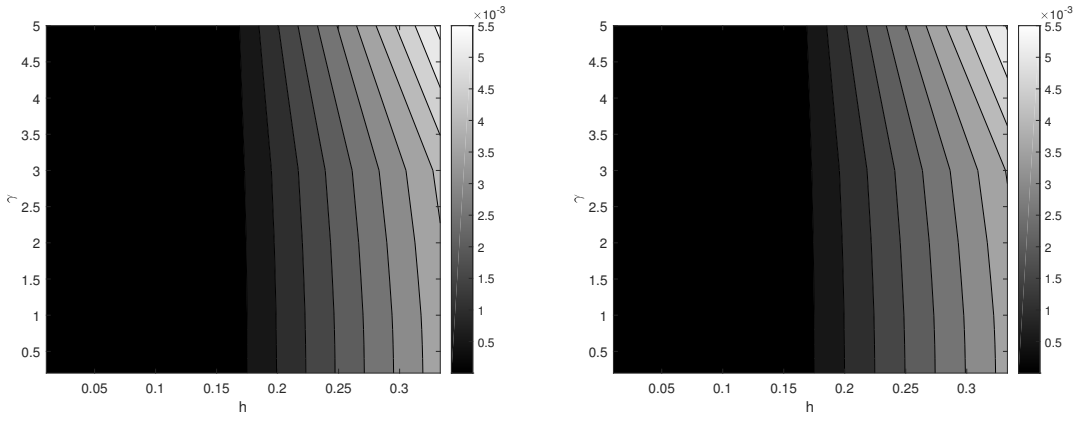

Figure 2.4: Contour plots of the l_2-error for $\rho = 0$ (left) and $\rho = -0.5$ (right) against parabolic mesh ratio $\gamma = \Delta_\tau / h^2$ and mesh width h

assumption that all involved discretisation matrices are normal and commuting. The discretisation matrices of high-order compact schemes generally do not fulfil these assumptions and, hence, in the present case with non-constant coefficients, the situation is much more involved. A thorough stability analysis is therefore beyond the scope of the present paper. Instead, to validate the stability of the scheme, we perform additional numerical stability tests. Stable behaviour has been consistently observed throughout numerical experiments. I have computed numerical solutions for varying values of the parabolic mesh ratio $\gamma = \Delta_t / h^2$ and the mesh width h. Plotting the associated l_2-norm errors in the plane should allow us to detect stability restrictions, this approach for a numerical stability study was also used in [23, 25].

Results for the European Put option using the parameters from Table 2.1 are provided here. For the stability plots I have used $\gamma = k/10$ with $k = 2, \ldots, 10$, as well as $\gamma = 1.5, 2.0, 3.0, 5.0$ and a descending sequence of spatial grid points, where $h = \frac{1}{192}, \frac{1}{96}, \ldots, \frac{1}{3}$. Figure 2.4 shows the stability plots for the correlation parameter $\rho = 0$ and $\rho = -0.5$. The influence of the parabolic mesh ratio γ on the l_2-error is only marginal and the relative error does not exceed 5.5×10^{-3} as a value for both stability plots. For increasing values of h the error grows gradually, and no oscillations in the numerical solutions occur.

These observations are confirmed by additional numerical convergence tests for varying parabolic mesh ratio γ. The other parameters are set to the default values given in Table 2.2. The numerical convergence orders reported in Table 2.2 show that the numerical convergence order for the high-order scheme, measured both in the l_2-norm and l_∞-norm is sufficiently close to four, and does not depend on the parabolic mesh ratio γ.

Choice of the parameter ϕ

In this section additional numerical experiments, conducted to investigate the influence of the parameter ϕ on our high-order ADI scheme in a convection dominated regime, are discussed (although we note that for practical parameter ranges the problem (2.5) is usually quite diffusive).

$\gamma = \Delta_t/h^2$	0.2	0.4	0.6	0.8	1.0	1.5	3.0	5.0
HO-ADI l_2-error	3.94	3.94	3.94	3.94	3.94	3.94	3.93	3.95
Standard ADI l_2-error	2.21	2.22	2.22	2.22	2.23	2.24	2.24	2.24
HO-ADI l_∞-error	3.61	3.61	3.61	3.61	3.61	3.61	3.62	3.63
Standard ADI l_∞-error	1.91	1.91	1.91	1.91	1.91	1.91	1.91	1.91

Table 2.2: Numerical convergence order in space for varying parabolic mesh ratio $\gamma = \Delta_t/h^2$

The typical choice for this parameter is $\phi = 0.5$ in the ADI splitting (2.8). In the literature larger values are sometimes found to be necessary to ensure stability for convection dominated problems, e.g. in [44] the condition $\phi \geq 1/2 + \sqrt{3}/6$ for a second-order discretisation for two-dimensional convection diffusion equations without mixed derivative is given. While a full stability analysis, which takes into account the interplay of ADI splitting and the high-order spatial discretisation would be beyond the scope of this work, some numerical investigation of the role of the parameter ϕ for this particular method seems in order. In order to facilitate a convection dominated problem (for pratical parameter ranges the problem is usually quite diffusive), we take the volatility of volatility parameter v to be ever closer to zero. We compare numerical convergence rates for the scheme using l_2- and l_∞-errors for the scheme for values $\phi = 1/2$ and $\phi > 1/2$. The results are report in Tables 2.3 and 2.4. In general, for increasingly convection dominated problems, there

$\gamma = \Delta_t/h^2$	0.2	0.4	0.6	0.8	1.0	1.5	3.0	5.0
$\phi = \frac{1}{2}$, l_2-error	3.87	3.91	3.90	3.90	3.91	3.90	3.90	3.88
$\phi = \frac{1}{2}$, l_∞-error	3.47	3.64	3.64	3.64	3.64	3.64	3.64	3.69
$\phi = \frac{1}{2} + \frac{\sqrt{3}}{6}$, l_2-error	3.88	3.91	3.91	3.89	3.90	3.90	3.90	3.90
$\phi = \frac{1}{2} + \frac{\sqrt{3}}{6}$, l_∞-error	3.64	3.64	3.64	3.54	3.64	3.64	3.64	3.66

Table 2.3: Numerical convergence order in space for the high-order ADI scheme for different parameters ϕ and $v = 0.04$ for increasing parabolic mesh ratios $\gamma = \Delta_t/h^2$; other parameters are set to the default values shown in Table 2.1.

$\gamma = \Delta_t/h^2$	0.2	0.4	0.6	0.8	1.0	1.5	3.0	5.0
$\phi = \frac{1}{2}$, l_2-error	1.34	1.74	2.02	2.79	1.93	2.24	3.00	3.10
$\phi = \frac{1}{2}$, l_∞-error	3.91	1.22	1.50	3.70	3.81	1.78	3.83	4.15
$\phi = \frac{1}{2} + \frac{\sqrt{3}}{6}$, l_2-error	1.59	2.48	1.84	2.62	2.11	2.27	2.83	3.29
$\phi = \frac{1}{2} + \frac{\sqrt{3}}{6}$, l_∞-error	3.85	1.97	1.32	2.12	3.86	1.70	2.29	4.14

Table 2.4: Numerical convergence order in space for the high-order ADI scheme for different parameters ϕ and $v = 0.025$ for increasing parabolic mesh ratios $\gamma = \Delta_t/h^2$; other parameters are set to the default values shown in Table 2.1.

is no observed advantage for the alternative choice of the parameter ϕ. For $v = 0.04$ the method is stable for both choices of ϕ over a wide range of parabolic mesh ratios, while there is negligible difference in the numerical convergence rates for both values of ϕ. For $v = 0.025$ the problem looses more and more its parabolic character and a loss of stability is reflected by the sporadic convergence order in space for various parabolic mesh ratios, with no apparent advantage for the alternative choice of ϕ.

Chapter 3

Calibrating forward jump diffusion option pricing model parameters to market prices

We propose to solve a parameter calibration problem for the forward jump-diffusion option pricing model. A distributed optimal control approach is employed, with a partial-integro differential equation as our state equation. With an emphasis on the analysis of the optimal control approach, we investigate the necessary regularity conditions for our parameters of interest. Following this, the existence of optimal solutions is proven under certain analytical conditions. Furthermore, the first-order necessary conditions for optimality are also established. In section 3.1, we introduce the jump diffusion option pricing model and the corresponding forward jump-diffusion model, which is the state equation for our optimal control approach. We formally specify the optimal control problem in 3.2 and construct a Lagrangian. In 3.3, we consider the existence of a solution to the state equation. In 3.4 we prove the existence of optimal solutions under particular regularity constraints, then in 3.5 we derive first-order necessary conditions of optimality under the same constraints and derive the adjoint equation from the Lagrangian. Finally, in 3.6 we calculate the gradients for both controls, make practical considerations for the numerical implementation of our method and give results.

3.1 State equation

The jump-diffusion option pricing model, as proposed by [49], assumes that the returns of some arbitrary financial asset with value $S(t) > 0$, where $t > 0$, follows the stochastic behaviour as specified by (3.1).

$$\frac{dS(t)}{S(t-)} = (r(t) - q(t) - \lambda(t)m(t))dt + \sigma(t, S(t-))dW(t) + (J(t) - 1)d\pi(t). \tag{3.1}$$

The riskless interest rate and dividend yield are given by $r(t) \geq 0$ and $q(t) \geq 0$, respectively. Furthermore, the uncertainty of the returns is modelled by a diffusion process

and a Poisson process. The diffusion process is modelled as a Wiener process, given by $W(t)$, while the diffusion coefficient is modelled by a deterministic volatility surface $\sigma(S,t)$. The Poisson process is given by $\pi(t) \in \{0,1\}$ and is parameterized by some deterministic event rate $\lambda(t) > 0$. As part of the random event coefficient, $J(t)$ is randomly sampled from some stochastic process $\eta(t,.)$, where the distribution parameters of $\eta(t,.)$ are time dependent at most. The expectation of the magnitude of the jump is given by $m(T) = E[J(t)-1]$. Traditionally, $\eta(t,.)$ is a log-normal distribution, as proposed by [49], or a double exponential distribution, as proposed by [39].
The following backward partial integro-differential equation is derived from (3.1) using Ito's lemma and the equivalent for the Poisson term, as both detailed in [42]:

$$C_t + (r(t) - q(t) - \lambda(t)m(t))SC_S + \frac{1}{2}\sigma^2(S,t)S^2C_{SS} + \lambda(t)\int_0^\infty C(Sz,t)\eta(z,t)dz = (r(t)+\lambda(t))C. \quad (3.2)$$

If, for example, (3.2) is used to model a European call option with exercise price $E > 0$ and expiry $T > 0$, then the boundary and initial conditions are given as follows:

$$C(0,t) = 0, \quad (3.3)$$

$$\lim_{S\to\infty} C(S,t) = S, \quad (3.4)$$

$$C(S,T) = \max(S-E,0). \quad (3.5)$$

The forward jump-diffusion option pricing model proposed by [1] is solved by the option price $\widetilde{C}(x,T)$, where $x = \ln(K)$ is the natural log of the strike price for $K > 0$ and $T > 0$ is the expiry. The value of the underlying asset, $S > 0$, is fixed at whatever the current asset value is.

$$\widetilde{C}_T + (\lambda(T)(1+m(T)) + q(T))\widetilde{C} + (r(T) - q(T) - \lambda(T)m(T) + \frac{1}{2}\widetilde{\sigma^2}(x,T))\widetilde{C}_x - \frac{1}{2}\widetilde{\sigma^2}(x,T)\widetilde{C}_{xx} - \lambda(T)\int_{-\infty}^\infty \widetilde{C}(x-y,T)e^{2y}\eta(e^y,T)dy = 0. \quad (3.6)$$

This option pricing model is particularly well-suited for parameter calibration. Market price data is available over a range of expiries and strikes, therefore we may calibrate model parameters to the forward jump-diffusion option pricing model by solving a single coupled system over such a domain for fixed $S > 0$ and $t = 0$. Otherwise, calibrating model parameters to non-forward option pricing models, such as (3.2), over a range of expiries and strikes requires us to solve a coupled system for every point (K,T) at which market price data is available. If we model European call options for a particular underlying asset, for example, using this forward PIDE, then the boundary and initial conditions are given as follows:

$$\lim_{x\to-\infty}\widetilde{C}(T,x)=Se^{-rT}, \tag{3.7}$$

$$\lim_{x\to\infty}\widetilde{C}(T,x_{max})=0, \tag{3.8}$$

$$\widetilde{C}(0,x)=\widetilde{C}_0(x)=\max(S-e^x,0). \tag{3.9}$$

We choose to specify the model over some truncated domain: $Q=\Omega\times[0,T_{max}]$, where $\Omega=[x_{min},x_{max}]$. We also revert back to a simpler notation for ease of viewing, such that $\widetilde{C}=C$. For ease of future derivations we choose to parameterize the volatility of the underlying asset by setting $v(x,T)=\frac{\widetilde{\sigma}^2(x,T)}{2}$. Then our model becomes

$$C_T+(\lambda(T)(1+m(T))+q(T))C+(r(T)-q(T)-\lambda(T)m(T)+v(x,T))C_x-v(x,T)C_{xx}$$
$$-\lambda(T)\int_{x-x_{max}}^{x-x_{min}}C(x-y,T)e^{2y}\eta(e^y,T)dy=0, \tag{3.10}$$

while the initial and boundary conditions are given as follows:

$$C(T,x_{min})=C_D(T)=(S-e^{x_{min}})e^{-rT}, \tag{3.11}$$
$$C(0,x)=C_0(x)=\max(S-e^x,0),$$
$$C(T,x_{max})=0.$$

This system, as specified by (3.10) and (3.11), is known as the "state equation". It governs the relationship between the controls we wish to imply using our optimal control approach and the solution to the state equation over a domain Q. In this case the option price is the solution to the state equation, and is therefore referred to as the "state". The controls that we seek to optimize and calibrate to market prices, are the Poisson parameter $\lambda(T)$ and the parameter $v(x,T)$. The state equation must hold at all times of the optimization process. As part of our gradient descent method, we are required to solve this state equation for every time we update our control parameters. The well-posedness of the state equation is therefore of interest to us.

3.2 Constructing the optimal control problem

State constraints

In order to formulate the optimal control problem, we must specify our state equation as a mapping from the function spaces of our controls and state from which they originate, to the dual space of the Lagrangian multiplier functions that we shall employ later as part of our optimal control strategy. We must specify appropriate function spaces for our state, volatility surface and lambda curve that are as nonrestrictive as possible but still guarantee the existence of a unique optimal solution and satisfy first-order necessary conditions of optimality. These spaces, as we shall show later, are $W(0,T_{max})$, $H^{2,1}(Q)$ and $H^1(0,T_{max})$, respectively. In particular, we define:

$$W(0,T_{max}) = \{f(x,T) : f \in L^2(0,T_{max};H^1(\Omega)) : f_t \in L^2(0,T_{max};H^1(\Omega)')\}, \quad (3.12)$$

$$H^{2,1}(Q) = H^1(0,T_{max};L^2(\Omega)) \cup L^2(0,T_{max};H^2(\Omega)), \quad (3.13)$$

Our chosen function spaces for our Lagrangian multiplier functions are determined to be the least restrictive such that the first order conditions of optimality are met. These spaces, as shown later, are $L^2(0,T_{max};H_0^1(\Omega))$,$L^2(0,T_{max})$ and $L^2(\Omega)$. In particular, we define:

$$L^2(0,T_{max};H_0^1(\Omega)) = \{f(x,T) : f \in L^2(0,T_{max};H^1(\Omega)),\ where\, f(\partial\Omega,T) = 0\}. \quad (3.14)$$

We impose "box constraints" for our chosen controls. In other words, we impose some arbitrary upper and lower bounds on our controls in addition to the restrictions imposed by their respective function spaces. The bounds must satisfy $0 < v_a < v_b$ and $0 < \lambda_a < \lambda_b$. We refer to these spaces as being the "admissible sets" of our controls and appropriately label these sets accordingly:

$$v(x,T) \in V_{ad} = \{H^{2,1}(Q) : v_a < v(x,T) < v_b \qquad \forall (x,T) \in Q\}, \quad (3.15)$$

$$\lambda(T) \in \lambda_{ad} = \{H^1(0,T) : \lambda_a < \lambda(T) < \lambda_b \qquad \forall T \in [0,T_{max}]\}. \quad (3.16)$$

These spaces are chosen as the least restrictive such that we may guarantee the existence of a unique optimal solution. This unique optimal solution is simply the unique combination of state and controls which minimize our chosen cost functional.

We define a multilinear mapping from the composite function space defined by our state and controls' function spaces to the dual of the composite function space defined by our Lagrangian multipliers' function spaces. For brevity, we label our integral term as $F(C)$:

$$\begin{aligned} e(C,v,\lambda) &: W(0,T_{max}) \times V_{ad} \times \lambda_{ad} \to (L^2(0,T_{max};H_0^1(\Omega)))' \times L^2(\Omega) \times L^2(0,T_{max}), \\ e_1(\omega) &= C_T + (\lambda(1+m)+q)C + (r-q-\lambda m+v)C_x - vC_{xx} - \lambda F(C), \\ e_2(\omega) &= C(\cdot,x_{min}) - C_D(T), \\ e_3(\omega) &= C(0,x) - C_0(x). \end{aligned}$$

This representation of the state constraints is directly transferable to our Lagrangian.

Cost functional

The implementation of a Lagrangian for an optimal control problem must also include a cost functional that we are seeking to minimize. Our cost functional should serve to minimize the distance over some metric between the prices generated by our model and

empirical market prices. The inclusion of Tikhonov regularization terms, for v and λ, addresses the ill-posedness of the calibration of these parameters. The level of desired smoothness for these parameters is controlled via a pair of regularization coefficients. In order to guarantee the well-posedness of the optimal control problem and first-order necessary conditions while choosing the least restrictive function spaces, our regularized cost functional is constructed as follows:

$$J(C, v, \lambda) = \frac{1}{2}||C(T_{max}) - C_{obs}||^2_{L^2(\Omega)} + \frac{\alpha}{2}||v - v_{init}||^2_{H^{2,1}(Q)} + \frac{\beta}{2}||\lambda - \lambda_{init}||^2_{H^1(0,T_{max})}, \quad (3.17)$$

where C_{obs} is our empirical market data, $v_{init} \in V_{ad}$ and $\lambda_{init} \in \lambda_{ad}$ are our initial guesses for our controls and $\alpha > 0$, $\beta > 0$ are chosen as regularization parameters. The exclusion of prior estimates for our controls from the cost functional would result in the undesirable penalization of non-zero values for our calibrated controls, hence the inclusion of fixed v_{init} and λ_{init}. The regularization parameters may be tuned to the preference of the practitioner, larger values favoured by those who require smoother controls and smaller values for those who require a higher rate of residual convergence. In practice, we will have to compute this cost function using l_2 counterparts for these norms and central difference operators for the controls' partial derivatives. For the case of calibrating to market data for n additional expiries, $0 < T_i < T_{max}$ for $i = 1, \ldots, n$, our cost will become:

$$\begin{aligned} J(C, v, \lambda) = \frac{1}{2}||C(T_{max}) - C_{obs}(T_{max})||^2_{L^2(\Omega)} + \frac{1}{2}\sum_{i=1}^{n} ||C(T_i) - C_{obs}(T_i)|| \\ + \frac{\alpha}{2}||v - v_{init}||^2_{H^{2,1}(Q)} + \frac{\beta}{2}||\lambda - \lambda_{init}||^2_{H^1(0,T_{max})}. \quad (3.18) \end{aligned}$$

Without loss of generality, we limit the remaining analysis of the optimal control problem to the case of a single expiry.

Lagrangian

Together, the state constraints and the cost functional form the Lagrangian. A Lagrangian multiplier function is assigned to each component of the multilinear mapping $e(C, v, \lambda)$. These are associated with the PIDE constraint, the boundary condition and the initial condition, respectively, and are specified in (3.19) as $p \in L^2(0, T_{max}; H^1_0(\Omega))$, $\mu \in L^2(0, T_{max})$ and $\upsilon \in L^2(\Omega)$. We will later formulate an adjoint equation composed of these Lagrangian multiplier functions which can be solved to help us determine how we should update our controls most effectively for each iteration of our method:

$$\mathbb{L}(\omega,\chi)=\frac{1}{2}||C(T_{max})-C_{obs}||^2_{L^2(\Omega)}+\frac{\alpha}{2}||v-v_{init}||^2_{H^{2,1}(Q)}+\frac{\beta}{2}||\lambda-\lambda_{init}||^2_{H^1(0,T_{max})}$$
$$+\int_\Omega(C(x,0)-C_0)vdx+\int_0^{T_{max}}(C(T,x_{min})-C_D)\mu dT+\int_0^{T_{max}}\Big(<C_T,p>_{V*,V}$$
$$+(\lambda(T)(1+m(T))+q(T))\int_\Omega Cpdx+\int_\Omega(r(T)-q(T)-\lambda(T)m(T)$$
$$+v(x,T))C_xpdx-\int_\Omega v(x,T)pC_{xx}dx-\lambda(T)\int_\Omega\int_{x-x_{max}}^{x-x_{min}}C(T,x-y)e^{2y}\eta(e^y,T)dypdx\Big)dT. \tag{3.19}$$

3.3 Solution to the state equation

A priori estimate

The following a priori estimate for the uniform boundedness of the solution is a standard result, the proof follows a similar path to a related problem in [70] .

Theorem 1 *Let C be a generalized solution to our state PIDE. Then there exists a constant $B^*>0$ such that $C<B^*$, where $B^*=\max_{0<\tau\leq T_{max}}B_\tau$*

Proof 1 For any fixed $\tau\in(0,T_{max}]$, $\phi(C)=C(x,T)-B(\tau)$, over time-space domain defined by $Q_\tau=\Omega\times(0,\tau]$. Consider the test function $\phi(C)^+=max(0,\phi(C))\in L^2(0,\tau;H_0^1(\Omega))$. Taking a weak formulation of our PIDE with this test function yields:

$$\frac{1}{2}\int_\Omega(\phi(C(\tau))^{+2}-\phi(C(0))^{+2})dx+\int_{Q_\tau}(\lambda(1+m)+q)\phi(C)^{+2}dxdT$$
$$+\int_{Q_\tau}(r-q-\lambda m+v)\phi(C)^+_x\phi(C)^+dxdT-\int_{Q_\tau}v\phi(C)^+_{xx}\phi(C)^+dxdT$$
$$-\int_{Q_\tau}\lambda\phi(C)^+\int_{x-x_{max}}^{x-x_{min}}\phi(C(x-y))^+e^{2y}\eta(e^y,T)dydxdT=0.$$

The term $\int_{Q_\tau}(\lambda(1+m)+q)\phi(C)^{+2}dxdT\geq 0$. By dropping this term, we have the inequality:

$$\frac{1}{2}\int_\Omega(\phi(C(\tau))^{+2}-\phi(C(0))^{+2})dx+\int_{Q_\tau}(r-q-\lambda m+v)\phi(C)^+_x\phi(C)^+dxdT$$
$$-\int_{Q_\tau}v\phi(C)^+_{xx}\phi(C)^+dxdT-\int_{Q_\tau}\lambda\phi(C)^+\int_{x-x_{max}}^{x-x_{min}}\phi(C(x-y))^+e^{2y}\eta(e^y,T)dydxdT\leq 0.$$

Given that r,q,λ,m,v are all bounded, we have for some $c\in\mathbb{R}$:

$$\int_{Q_\tau}(r-q-\lambda m+v)\phi(C)^+_x\phi(C)^+dxdT\geq c\int_{Q_\tau}\phi(C)^+_x\phi(C)^+dxdT. \tag{3.20}$$

Due to the compact support of $\phi(C)^+$ at the spatial boundaries, integrating by parts yields the following:

$$\int_{Q_\tau} \phi(C)_x^+ \phi(C)^+ dxdT = -\int_{Q_\tau} \phi(C)^+ \phi(C)_x^+ dxdT \to \int_{Q_\tau} \phi(C)_x^+ \phi(C)^+ dxdT = 0. \quad (3.21)$$

Dropping this term from our inequality yields:

$$\frac{1}{2}\int_\Omega (\phi(C(\tau))^{+2} - \phi(C(0))^{+2})dx - \int_{Q_\tau} v\phi(C)_{xx}^+ \phi(C)^+ dxdT$$
$$- \int_{Q_\tau} \lambda\phi(C)^+ \int_{x-x_{max}}^{x-x_{min}} \phi(C(x-y))^+ e^{2y}\eta(e^y, T)dydxdT \leq 0.$$

Seeing as v is bounded, there exists some $c > 0$ such that:

$$-\int_{Q_\tau} v\phi(C)_{xx}^+ \phi(C)^+ dxdT \geq -c\int_{Q_\tau} \phi(C)_{xx}^+ \phi(C)^+ dxdT. \quad (3.22)$$

Integrating by parts gives us:

$$-c\int_{Q_\tau} \phi(C)_{xx}^+ \phi(C)^+ dxdT = c\int_{Q_\tau} \phi(C)_x^{+2} dxdT. \quad (3.23)$$

This term is positive, therefore we drop it from our inequality. This yields:

$$\frac{1}{2}\int_\Omega (\phi(C(\tau))^{+2} - \phi(C(0))^{+2})dx$$
$$\leq \int_{Q_\tau} \lambda\phi(C)^+ \int_{x-x_{max}}^{x-x_{min}} \phi(C(x-y))^+ e^{2y}\eta(e^y, T)dydxdT.$$

Given that lambda is bounded, there exists some $c > 0$ such that:

$$\frac{1}{2}\int_\Omega (\phi(C(\tau))^{+2} - \phi(C(0))^{+2})dx$$
$$\leq c\int_{Q_\tau} \phi(C)^+ \int_{x-x_{max}}^{x-x_{min}} \phi(C(x-y))^+ e^{2y}\eta(e^y, T)dydxdT.$$

We know that $e^{2y}\eta(e^y)$ is bounded for $y \in [x_{min} - x_{max}, x_{max} - x_{min}]$. Therefore there exists some $c_\eta > 0$:

$$c\int_{Q_\tau} \phi(C)^+ \int_{x-x_{max}}^{x-x_{min}} \phi(C(x-y))^+ e^{2y}\eta(e^y, T)dydxdT$$
$$\leq c\int_{Q_\tau} \phi(C)^+ \int_{x-x_{max}}^{x-x_{min}} \phi(C(y))^+ c_\eta dydxdT. \quad (3.24)$$

We absorb c_η into the positive constant c. Simplifying this term:

$$c\int_{Q_\tau} \phi(C)^+ \int_{x-x_{max}}^{x-x_{min}} \phi(C(x-y))^+ dydxdT = c\int_{Q_\tau} \phi(C)^+ \int_{x_{min}}^{x_{max}} \phi(C(y))^+ dydxdT. \quad (3.25)$$

Applying Cauchy-Schwarz to the inner integral term:

$$c\int_{Q_\tau}\phi(C)^+\int_{x_{min}}^{x_{max}}\phi(C(y))^+dydxdT \leq c\int_{Q_\tau}\phi(C)^+||\phi(C)^+||_{L^2(\Omega)}||1||_{L^2(\Omega)}dxdT. \quad (3.26)$$

Absorbing $||1||_{L^2(\Omega)}$ into the positive constant c and then applying Cauchy-Schwarz to the remaining integral term:

$$c\int_0^\tau ||\phi(C)^+||_{L^2(\Omega)}\int_\Omega\phi(C)^+dxdT \leq c\int_0^\tau ||\phi(C)^+||^2_{L^2(\Omega)}||1||_{L^2(\Omega)}dT. \quad (3.27)$$

Absorbing $||1||_{L^2(\Omega)}$ into the postive constant c again yields:

$$c\int_0^\tau ||\phi(C)^+||^2_{L^2(\Omega)}dT = c\int_{Q_\tau}\phi(C)^{+2}dT. \quad (3.28)$$

Substituting this result into our inequality yields:

$$\frac{1}{2}\int_\Omega(\phi(C(\tau))^{+2}-\phi(C(0))^{+2})dx \leq c\int_{Q_\tau}\phi(C)^{+2}dT. \quad (3.29)$$

Choosing $B_\tau = max(C_0(x), C_D(T))$ for $x \in \Omega$ and $T \in (0,\tau]$, we have $\phi(C)^+(0) = 0$. Therefore:

$$\frac{1}{2}\int_\Omega\phi(C(\tau))^{+2}dx \leq c\int_{Q_\tau}\phi(C)^{+2}dT. \quad (3.30)$$

Applying Grönwall's Lemma to this inequality for fixed $\tau \in (0, T_{max}]$ gives us $C(\tau) < B_\tau$. Therefore, for all $T \in (0, T_{max}]$, we have $C < B^*$, where $B^* = \max_{0<\tau\leq T_{max}} B_\tau$.

Existence of weak solution

The existence of weak solutions to the state equation's type is classical, our proof follows a standard Galerkin approach as given in [61] for a class of similar parabolic PDEs. A definitive collection of such classical results for modern PDE theory is given in [71]. Alternatively, [72] outlines a generalized application of C^0 semigroup theory to obtain a weak existence result for a group of parabolic PDEs which are closely related to our state equation.

Theorem 2 *There exists a weak solution $C(x,T)$ to our state equation* (3.10) *with initial condition and boundary conditions* (3.11)*, such that $C \in W(0, T_{max})$.*

Assumptions: $\lambda, r, q, m \in L^\infty(0, T_{max}]$ and $v \in L^\infty(0, T_{max}; L^\infty(\Omega))$, initial condition $C_0(x) \in L^2(\Omega)$ and boundary condition $C_D(T) \in L^2(0, T_{max}]$.

Proof 2 The steps of the proof are to construct a sequence of Galerkin approximations for our solution, derive some estimates for these approximations, determine a weakly converging subsequence, prove the existence of this weak limit in a particular function space and finally extend this result to the function space $W(0, T_{max})$.

Part 1: Galerkin approximation

$H_0^1(\Omega)$ is a subspace of a separable Hilbert space, therefore it is also a separable Hilbert space and we may choose a countable dense set of linearly independent elements $\{z_i\}_{i=1}^{\infty} \in H_0^1(\Omega)$ as an orthonormal system in $H_0^1(\Omega)$ which is also complete in $H^1(\Omega)$. It is important to note that $H_0^1(\Omega)$ is, in general, dense in $H^1(\Omega)$, and is also continuously embedded in $H^1(\Omega)$. Therefore any results we obtain for a weak solution in $H_0^1(\Omega)$ shall hold for $H^1(\Omega)$ and our PIDE with non-zero boundary conditions. For some fixed $n \in \mathbb{N}$, we approximate solution $C_n = C_n(x,T)$ through the ansatz

$$C_n(x,T) = \sum_{i=1}^{n} u_i^n(T) z_i(x). \tag{3.31}$$

where $u_i^n : (0, T_{max}] \to \mathbb{R}$ for $i = 1, \ldots, n$ are unknown functions which are considered to be coefficient functions in the time domain. We define for any fixed $T \in (0, T_{max}]$ the following bilinear form, where, for brevity, we have taken $F(C) = \int_{x-x_{max}}^{x-x_{min}} C(x-y,T)e^{2y}\eta(e^y,T)dy$:

$$\begin{aligned} a[T;C,z] = \int_\Omega &(\lambda(T)(1+m(T)) + q(T))C(x,T)z(x) + (r(T) - q(T) - \lambda(T)m(T) \\ &+ v(x,T))C_x(x,T)z(x) - v(x,T)C_{xx}(x,T)z(x) - \lambda(T)F(C(x,T)))z_j dx. \end{aligned} \tag{3.32}$$

Taking the inner product of $z_j \in H_0^1(\Omega)$ with our the PIDE in $L^2(\Omega)$, we have the weak form of our PIDE:

$$(\frac{d}{dt}C_n(T), z_j) + a(T; C_n, z_j) = 0, \quad \forall z_j. \tag{3.33}$$

At the lower boundary of expiries $(C_n(.,0), z_j)_{L^2(\Omega)} = (C_0, z)_{L^2(\Omega)}$ for all $z_j \in H_0^1(\Omega)$ for any $j \in 1, \ldots, n$, for any $n \in \mathbb{N}$. If $C_n = \sum_{j=1}^n u_j^n z_j$, then due to the orthonormality of z_j we have:

$$\frac{d}{dt}u_j^n(T) + \sum_{i=1}^{n} u_i(T)a[t; z_i, z_j] = 0, \tag{3.34}$$

$$u_j^n(0) = (C_0, z_j). \tag{3.35}$$

for any $j = 1, \ldots, n$. According to Caratheodory's theorem, this initial value problem for n linear ordinary differential equations on $[0, T_{max}]$ for $u^n = (u_1^n, \ldots, u_n^n)$ has a unique absolutely continuous solution $u^n \in (H^1(0,T))^n$. If we multiply equation (3.33) by $u_j(T)$ and sum the equations from $j = 1$ to $j = n$, then for almost every $T \in (0, T_{max}]$:

$$(\frac{d}{dt}C_n(T), C_n(T)) + a[T, C_n(T), C_n(T)] = 0. \tag{3.36}$$

The function $C_n : [0, T_{max}] \to L^2(\Omega)$ is also almost everywhere differentiable with respect to $T \in (0, T_{max}]$.

Part 2: Estimates for C_n

Choosing any fixed $\tau \in (0, T_{max}]$, we have

$$\int_0^\tau (\frac{d}{dt}C_n(T), C_n(T))_{L^2(\Omega)}dT = \frac{1}{2}\int_0^\tau \frac{d}{dt}||C_n(T)||^2_{L^2(\Omega)}dT$$
$$= \frac{1}{2}||C_n(\tau)||^2_{L^2(\Omega)} - \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.37)$$

Integration of equation (3.36) over $[0, \tau]$ yields

$$\frac{1}{2}||C_n(\tau)||^2_{L^2(\Omega)} + \int_0^\tau a[T; C_n(T), C_n(T)]dT = \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.38)$$

If we expand the operator a, we have

$$\frac{1}{2}||C_n(\tau)||^2_{L^2(\Omega)} + \int_0^\tau ((1+m)\lambda + q)(C_n, C_n)_{L^2(\Omega)} + (r - q - \lambda m)((C_{nx}, C_n)_{L^2(\Omega)}$$
$$+ (vC_{nx}, C_n)_{L^2(\Omega)} - (vC_{nxx}, C_n)_{L^2(\Omega)} - \lambda(F(C_n), C_n)_{L^2(\Omega)}dT = \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.39)$$

The coefficient $((1+m)\lambda + q) > 0$, therefore we may take $c_1 = \inf((1+m)\lambda + q)$ such that:

$$((1+m)\lambda + q)(C_n, C_n)_{L^2(\Omega)} \geq c_1||C_n||^2_{L^2(\Omega)}. \quad (3.40)$$

Substituting this into our weak formulation:

$$\frac{1}{2}||C_n(\tau)||^2_{L^2(\Omega)} + \int_0^\tau c_1||C_n||^2_{L^2(\Omega)} + (r - q - \lambda m)(C_{nx}, C_n)_{L^2(\Omega)} + (vC_{nx}, C_n)_{L^2(\Omega)}$$
$$- (vC_{nxx}, C_n)_{L^2(\Omega)} - \lambda(F(C_n), C_n)_{L^2(\Omega)}dT \leq \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.41)$$

We integrate the next term $\int_0^\tau ((r - q - \lambda m)((C_{xn}, C_n)_{L^2(\Omega)}dT$ by parts as follows:

$$(r-q-\lambda m)(C_{nx}, C_n)_{L^2(\Omega)} = -(r-q-\lambda m)(C_n, C_{nx})_{L^2(\Omega)} + [(r-q-\lambda m)C_n^2]_{x_{min}}^{x_{max}}. \quad (3.42)$$

Given that $C_n = \sum_{i=1}^n u_j^n z_j^n$, and $z_j^n \in H_0^1(\Omega)$, we know that the value of C_n is equal to 0 at the boundaries x_{min} and x_{max}. Therefore the whole term is equivalent to 0:

$$(r - q - \lambda m)(C_{nx}, C_n)_{L^2(\Omega)} = 0. \quad (3.43)$$

Our weak formulation simplifies to the following:

$$\frac{1}{2}||C_n(\tau)||^2_{L^2(\Omega)} + \int_0^\tau c_1||C_n||^2_{L^2(\Omega)} + (vC_{nx}, C_n)_{L^2(\Omega)} - (vC_{nxx}, C_n)_{L^2(\Omega)}$$
$$- \lambda(F(C_n), C_n)_{L^2(\Omega)}dT \leq \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.44)$$

Given that our control parameters v and λ are bounded, we infer the existence of constants $c_2 > 0$ and $c_3 > 0$ such that:

$$\frac{1}{2}||C_n(\tau)||^2_{L^2(\Omega)} + \int_0^\tau c_1||C_n||^2_{L^2(\Omega)} + c_2(C_{nx}, C_n)_{L^2(\Omega)} - c_2(C_{nxx}, C_n)_{L^2(\Omega)} - c_3(F(C_n), C_n)_{L^2(\Omega)}dT \leq \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.45)$$

For the next term, $(C_{nx}, C_n)_{L^2(\Omega)}$, we can integrate by parts as follows:

$$\int_{x_{min}}^{x_{max}} C_n C_{nx} dx = -\int_{x_{min}}^{x_{max}} C_{nx} C_n dx + [C_n^2]_{x_{min}}^{x_{max}}. \quad (3.46)$$

Therefore this term is also equivalent to 0:

$$\int_{x_{min}}^{x_{max}} C_n C_{nx} dx = 0. \quad (3.47)$$

Our weak formulation then becomes:

$$\frac{1}{2}||C_n(\tau)||^2_{L^2(\Omega)} + \int_0^\tau c_1||C_n||^2_{L^2(\Omega)} - c_2((C_{nxx}, C_n)_{L^2(\Omega)} - c_3(F(C_n), C_n)_{L^2(\Omega)}dT \leq \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.48)$$

The next term $-(C_{nxx}, C_n)_{L^2(\Omega)}$ can be integrated by parts as follows:

$$-\int_{x_{min}}^{x_{max}} C_n C_{nxx} dx = \int_{x_{min}}^{x_{max}} C_{nx}^2 dx - [C_n C_{nx}]_{x_{min}}^{x_{max}} = ||C_{nx}||^2_{L^2(\Omega)}. \quad (3.49)$$

Then our weak formulation becomes:

$$\frac{1}{2}||C_n(\tau)||^2 + \int_0^\tau c_1||C_n||^2_{L^2(\Omega)} + c_2||C_{nx}||^2_{L^2(\Omega)} - c_3(F(C_n), C_n)_{L^2(\Omega)}dT \leq \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)}. \quad (3.50)$$

Now we consider the integral term $F(C_n)$. Referring to the a priori estimate, we have already shown that any C_n satisfying the state PIDE is bounded above by some $B^* > 0$. Therefore, for some $c_{\eta,\Omega,B^*} > 0$:

$$(F(C_n), C_n)_{L^2(\Omega)} = \int_{x_{min}}^{x_{max}} C_n(x,T) \int_{x-x_{max}}^{x-x_{min}} C_n(x-y,T) e^{2^y} \eta(e^y, T) dy dx < c_{\eta,\Omega,B^*}. \quad (3.51)$$

By taking the integral term over to the right hand side and using this estimate, we have:

$$\frac{1}{2}||C_n(\tau)||^2 + \int_0^\tau c_1||C_n||^2_{L^2(\Omega)} + c_2||C_{nx}||^2_{L^2(\Omega)}dT \leq \frac{1}{2}||C_n(0)||^2_{L^2(\Omega)} + c_{\eta,\Omega,B^*}. \quad (3.52)$$

Choosing $\tau = T_{max}$, for some $K^* > 0$, we have:

$$||C_n(T)||^2_{C(0,T;L^2(\Omega))} + ||C_n||^2_{W_2^{1,0}(Q)} \leq K^*. \quad (3.53)$$

Therefore, there exists a constant $K > 0$ such that:

$$||C_n(T)||_{C(0,T;L^2(\Omega))} + ||C_n||_{W_2^{1,0}(Q)} \leq K \quad \forall n \in \mathbb{N}. \tag{3.54}$$

By virtue of the continuity of C_n with respect to the domain of expiries,

$$\sum_{i=1}^{n} |u_i^n(T)|^2 \leq K^2 \quad \forall\, T \in (0, T_{max}],\, \forall n \in \mathbb{N}.$$

Part 3: Weak convergence of C_n

Due to the uniform boundedness of $u_j^n(T)$ for all t, j and n, the sequence of functions for any fixed $j \in \mathbb{N}$ forms an equicontinuous set in $C[0, T_{max}]$ and the Arzela-Ascoli theorem can be applied to any such sequence. Through a diagonal selection procedure of subsequences, we may construct a subsequence of indices $\{n_k\}_{k=1}^{\infty}$ through which the corresponding sequence of functions converge strongly for each $j \in \mathbb{N}$:

$$\lim_{k\to\infty} u_j^{n_k} = u_j \quad \text{strongly in } C[0, T_{max}] \quad \forall\ j \in \mathbb{N}. \tag{3.56}$$

With the limit functions u_j constructed for each $j \in \mathbb{N}$, we define the function:

$$C(x, T) = \sum_{i=1}^{\infty} u_i(T) v_i(x), \quad (x, T) \in Q. \tag{3.57}$$

If the sequence is $||C_n(T)||_{C(0,T;L^2(\Omega))}$ bounded for all $n \in \mathbb{N}$ then there must exist some sub-sequence $\{C_{n_k}(., T)\}$ that converges weakly in $L^2(\Omega)$ to $C(., T)$, uniformly with respect to $T \in [0, T_{max}]$. The norm for $C(0, T_{max}; L^2(\Omega))$ is weakly lower sequential semicontinuous and we can conclude that $||C(T)||_{L^2(\Omega)} \leq K$ for almost all T. Therefore $C \in L^{\infty}(0, T_{max}; L^2(\Omega))$. Additionally $C_{n_k}(0)$ converges strongly in $L^2(\Omega)$ to C_0, as $k \to \infty$

Part 4: C is a weak solution in $W_2^{1,0}(Q)$

Due to the result of (3.54), we infer that $\{C_{n_k}\}_{k=1}^{\infty}$ converges weakly in $W_2^{1,0}(Q)$ to C. We seek to construct a weak formulation for the PIDE in the weakly convergent subsequence of C_{n_k} with a test function $w(x, T)$ given by

$$w(x, T) = \sum_{j=1}^{\infty} \alpha_j(T) z_j(x), \tag{3.58}$$

where $\alpha_j \in C^1[0, T_{max}]$ for $j \in \mathbb{N}$. It then follows that a weak formulation of our PIDE in C_{n_k} with w is given by:

$$\frac{d}{dt}(C_{n_k}(T), w(T))_{L^2(\Omega)} + a[T; C_{n_k}(T), w(T)] = 0. \tag{3.59}$$

If we integrate this weak formulation over our domain of expiries:

$$-\int_0^{T_{max}}(C_{n_k}(T),\frac{d}{dt}w(T))dT+\int_0^{T_{max}}a[T;C_{n_k}(T),w(T)]dT=0. \tag{3.60}$$

If $C_{n_k}\to C$ weakly in $W_2^{1,0}(Q)$ and $C_{n_k}(0)\to C_0$ strongly in $L^2(\Omega)$, then as $k\to\infty$:

$$-\int_0^{T_{max}}(C(T),\frac{d}{dt}w(T))dT+\int_0^{T_{max}}a[T;C(T),w(T)]dT=0. \tag{3.61}$$

Therefore, we have that $C\in W_2^{1,0}(Q)\cap L^\infty(0,T_{max};L^2(\Omega))$ satisfies the weak formulation and that it is a weak solution.

Part 5: C is a weak solution in $W(0,T_{max})$

Consider solution $C\in W_2^{1,0}(Q)$. We may take a weak formulation with $w(x,T)=\phi(T)z(x)$, where $\phi\in C_0^\infty(0,T_{max})$ such that $\phi(0)=\phi(T_{max})=0$ and $z\in H_0^1(\Omega)$. The weak formulation of the PIDE solved by $C(x,T)$, with test function w, is given by:

$$\begin{aligned}\int_0^{T_{max}}(C_T(T)\phi(T),z)_{L^2(\Omega)}dT&=\int_0^{T_{max}}(C_{xx}(T),vz)_{L^2(\Omega)}\phi(T)dT\\&+\int_0^{T_{max}}\lambda(T)\phi(T)(F(C),z)_{L^2(\Omega)}dT-\int_0^{T_{max}}(r-q-\lambda m)(C_x,z)_{L^2(\Omega)}\phi(T)dT\\&-\int_0^{T_{max}}(vC_x,z)_{L^2(\Omega)}\phi(T)dT-\int_0^{T_{max}}((1+m)\lambda+q)(C,z)_{L^2(\Omega)}\phi(T)dT.\end{aligned} \tag{3.62}$$

Integrating the term $(C_{xx}(T),vz)_{L^2(\Omega)}$ by parts gives us:

$$(C_{xx}(T),vz)_{L^2(\Omega)}=-(C_x(T),v_xz)_{L^2(\Omega)}-(C_x(T),vz_x)_{L^2(\Omega)}. \tag{3.63}$$

Substituting this into our weak formulation, we get:

$$\begin{aligned}\int_0^{T_{max}}(C_T(T)\phi(T),z)_{L^2(\Omega)}dT&=-\int_0^{T_{max}}(C_x(T),v_xz)_{L^2(\Omega)}dT\\&-\int_0^{T_{max}}(C_x(T),vz_x)_{L^2(\Omega)}dT+\int_0^{T_{max}}\lambda(T)\phi(T)(F(C),z)_{L^2(\Omega)}dT\\&-\int_0^{T_{max}}(r-q-\lambda m)(C_x,z)_{L^2(\Omega)}\phi(T)dT-\int_0^{T_{max}}(vC_x,z)_{L^2(\Omega)}\phi(T)dT\\&-\int_0^{T_{max}}((1+m)\lambda+q)(C,z)_{L^2(\Omega)}\phi(T)dT.\end{aligned} \tag{3.64}$$

If we integrate the term $\int_0^{T_{max}}(C_T(T)\phi(T),z)_{L^2(\Omega)}dT$ by parts with respect to T, the boundary terms dissapear due to the compact support of $\phi(T)$:

$$\int_0^{T_{max}}(C_T(T)\phi(T),z)_{L^2(\Omega)}dT=-\int_0^{T_{max}}(C(T)\phi'(T),z)_{L^2(\Omega)}dT. \tag{3.65}$$

Substituting this into our weak formulation:

$$-\int_0^{T_{max}}(C(T)\phi'(T),z)_{L^2(\Omega)}dT=-\int_0^{T_{max}}(C_x(T)v_x,z)_{L^2(\Omega)}\phi(T)dT$$
$$-\int_0^{T_{max}}(C_x(T)v,z_x)_{L^2(\Omega)}\phi(T)dT+\int_0^{T_{max}}\lambda(T)\phi(T)(F(C),z)_{L^2(\Omega)}dT$$
$$-\int_0^{T_{max}}(r-q-\lambda m)(C_x,z)_{L^2(\Omega)}\phi(T)dT-\int_0^{T_{max}}(vC_x,z)_{L^2(\Omega)}\phi(T)dT$$
$$-\int_0^{T_{max}}((1+m)\lambda+q)(C,z)_{L^2(\Omega)}\phi(T)dT. \quad (3.66)$$

If we fix $T\in(0,T_{max})$, then our solution must exist in $L^2(\Omega)$ if $C\in W_2^{1,0}(Q)$. Hence $C(.,T)$ exists in $L^2(\Omega)$ for every $T\in(0,T_{max})$. Moreover $C_x\in L^2(Q)$ and therefore $C_x(.,T)\in L^2(\Omega)$ for any $T\in(0,T_{max})$. Our boundary conditions also satisfy $C(x_{min},T)\in L^2(0,T_{max}]$ and $C(x_{max},T)\in L^2(0,T_{max}]$.

For any fixed T, the expressions on the right hand side of (3.66) are separately identified as linear functionals $F_i(T):H_0^1(\Omega)\to\mathbb{R}$:

$$\begin{aligned}
F_1(T):\ z &\to (v(T)C_x(T),z_x)_{L^2(\Omega)},\\
F_2(T):\ z &\to (v_x(T)C_x(T),z)_{L^2(\Omega)},\\
F_3(T):\ z &\to (\lambda(T)F(C(T)),z)_{L^2(\Omega)},\\
F_4(T):\ z &\to ((r(T)-q(T)-\lambda(T)m(T))C_x(T),z)_{L^2(\Omega)},\\
F_5(T):\ z &\to (v(T)C_x(T),z)_{L^2(\Omega)},\\
F_6(T):\ z &\to ((1+m(T))\lambda(T)+q(T))(C(T),z)_{L^2(\Omega)}.
\end{aligned}$$

If we can show that these linear functionals $F_i(T)$ for $1\le i\le 8$ are bounded, then they are continuous, linear functionals on $H_0^1(\Omega)$ for every T. We verify this feature for each functional in order:

$$|F_1(T)z|\le\int_{x_{min}}^{x_{max}}v(T)C_x(T)z_xdx\le||v||_{L^\infty(Q)}||C(T)||_{H^1(\Omega)}||z||_{H^1(\Omega)}\ \forall z\in H^1(\Omega). \quad (3.67)$$

The right hand side of the above inequality is finite for every T, therefore the linear functional must be continuous for every T. We follow the same approach for the remaining functionals.

$$|F_2(T)z|=\int_{x_{min}}^{x_{max}}v_x(T)C_x(T)zdx\le||v_x||_{L^\infty(Q)}||C(T)||_{H^1(\Omega)}||z||_{H^1(\Omega)}\ \forall z\in H^1(\Omega). \quad (3.68)$$

$$|F_3(T)z| = |\int_{x_{min}}^{x_{max}} (\lambda(T)F(C(T)), z)_{L^2(\Omega)} dx|$$
$$= |\lambda(T) \int_{x_{min}}^{x_{max}} z \int_{x-x_{max}}^{x-x_{min}} C(x-y,T)e^{2y}\eta(e^y,T)dydx|$$
$$\leq ||\lambda||_{L^\infty(0,T_{max})}| \int_{x_{min}}^{x_{max}} z \int_{x-x_{max}}^{x-x_{min}} C(x-y,T)e^{2y}\eta(e^y,T)dydx|. \quad (3.69)$$

The inner integral term itself is bounded for all $x \in (x_{min}, x_{max})$. This is due to the fact that $C(x,T)$ is at least an $L^2(\Omega)$ function. Therefore, $C(x-y,T)$ is an L^2 integrable function for $y \in (x-x_{min}, x-x_{max})$, where $x \in (x_{min}, x_{max})$. The fact that $e^{2y}\eta(e^y,T)$ is bounded for all $y \in (x_{min}-x_{max}, x_{max}-x_{min})$ and $T \in (0, T_{max})$, means that it is also L^2 integrable. Therefore, applying Holder's inequality to the inner integral term:

$$||\lambda||_{L^\infty(0,T_{max})}| \int_{x_{min}}^{x_{max}} z \int_{x-x_{max}}^{x-x_{min}} C(x-y,T)e^{2y}\eta(e^y,T)dydx|$$
$$\leq ||\lambda||_{L^\infty(0,T_{max})}| \int_{x_{min}}^{x_{max}} z||C(T)||_{L^2(\Omega)}||e^{2y}\eta(e^y,T)||_{L^2(\Omega_x)}dx|$$
$$= ||\lambda||_{L^\infty(0,T_{max})}(x_{max}-x_{min})||C(T)||_{L^2(\Omega)}| \int_{x_{min}}^{x_{max}} z||e^{2y}\eta(e^y,T)||_{L^2(\Omega_x)}dx|. \quad (3.70)$$

Here we have taken Ω_x to convey the domain $(x-x_{min}, x-x_{max})$ that is dependent on $x \in (x_{min}, x_{max})$. For any choice of $x \in (x_{min}, x_{max})$, we have that $||e^{2y}\eta(e^y,T)||_{L^2(\Omega_x)} \leq \infty$. Therefore $||e^{2y}\eta(e^y,T)||_{L^2(\Omega_x)}$ is an $L^\infty(\Omega)$ function. Therefore it must also be an $L^2(\Omega)$ function. We also have that $z \in H_0^1(\Omega)$ and is therefore also a member of $L^2(\Omega)$. Therefore we can apply Holder's inequality such that:

$$||\lambda||_{L^\infty(0,T_{max})}(x_{max}-x_{min})||C(T)||_{L^2(\Omega)}| \int_{x_{min}}^{x_{max}} z||e^{2y}\eta(e^y,T)||_{L^2(\Omega_x)}dx|$$
$$\leq ||\lambda||_{L^\infty(0,T_{max})}(x_{max}-x_{min})||C(T)||_{L^2(\Omega)}||z||_{L^2(\Omega)}||||e^{2y}\eta(e^y,T)||_{L^2(\Omega_x)}||_{L^2(\Omega)} < \infty$$
$$\leq ||\lambda||_{L^\infty(0,T_{max})}(x_{max}-x_{min})||C(T)||_{H^1(\Omega)}||z||_{H^1(\Omega)}||||e^{2y}\eta(e^y,T)||_{L^2(\Omega_x)}||_{L^2(\Omega)}. \quad (3.71)$$

To show the boundedness of the remaining functionals is much simpler:

$$|F_4(T)z| = |(r(T)-q(T)-\lambda(T)m(T))(C_x(T),z)_{L^2(\Omega)}| \leq c_{r,q,\lambda,m}||C(T)||_{H^1(\Omega)}||z||_{H^1(\Omega)}. \quad (3.72)$$

$$|F_5(T)| = |(v(T)C_x(T),z)_{L^2(\Omega)}| \leq ||v(T)||_{L^\infty(Q)}||C(T)||_{H^1(\Omega)}||z||_{H^1(\Omega)}. \quad (3.73)$$

$$|F_6(T)| = |((1+m(T))\lambda(T)+q(T))(C(T),z)_{L^2(\Omega)}| \leq c_{m,\lambda,q}||C(T)||_{H^1(\Omega)}||z||_{H^1(\Omega)}. \quad (3.74)$$

We find that $F_i(T) \in H_0^1(\Omega)^*$ for $i = 1, \ldots, 6$ and for every T. There is some constant $c > 0$ such that:

$$\sum_{i=1}^{6} ||F_i(T)||_{H^1(\Omega)^*} \leq c||C(T)||_{H^1(\Omega)}. \tag{3.75}$$

Since the sum of expressions on the right hand side of our PIDE are therefore in $L^2(0, T_{max})$, we may conclude that the term on the left hand side of our PIDE must also belong to this space. If $F_i \in L^2(0, T_{max}; H_0^1(\Omega)^*)$ for $1 \leq i \leq 6$, then we have that the functional $F = \sum_{i=1}^{6} F_i$ also belongs to $L^2(0, T_{max}; H_0^1(\Omega)^*$. The variational formulation of our PIDE for all $z \in H_0^1(\Omega)$ therefore satisfies:

$$-\int_0^{T_{max}} (C(T)\phi'(T), z)_{L^2(\Omega)} dT = (\int_0^{T_{max}} F(T)\phi(T)dT, z)_{H^1(\Omega)^*, H^1(\Omega)}. \tag{3.76}$$

We observe that the weak derivative of C with respect to T satisfies $C' = F$ in the sense of vector-valued distributions:

$$-\int_0^{T_{max}} C(T)\phi'(T)dT = \int_0^{T_{max}} F(T)\phi(T)dT \;\; \phi \in C_0^{\infty}(0, T_{max}). \tag{3.77}$$

hence $C' \in L^2(0, T_{max}; H_0^1(\Omega)^*)$. Therefore, we have $C' \in L^2(0, T_{max}; H^1(\Omega)^*)$ In conclusion, $C \in W_2^{1,0}(\Omega)$, such that $C_T \in L^2(0, T_{max}; H^1(\Omega)^*)$. We shall refer to this space as $W(0, T_{max})$.

3.4 Existence of optimal solutions

Our objective can be summed up simply as trying to minimize the regularized cost functional $J(C, v, \lambda)$ such that our multi-linear operator satisfies $e(C, v, \lambda) = 0$, where $v \in V_{ad}$ and $\lambda \in \lambda_{ad}$. Here, we consider the existence of such an optimal solution. Our approach follows [73], which serves as a reference text for the modern theory of regularized inverse problems. A similar application of this theory to the case of implying local volatility for Dupire's forward equation can be found in [64, 21]. The extension of this application to the case of Andersen and Andreasen's forward jump-diffusion equation involves proving that for our choice of admissible set of jump parameters, we are still able to pass to the limit in the weak formulation of our state equation. While this approach does not introduce any novel ideas, it has not yet been explicitly outlined for this particular inverse problem in the literature. It is of interest nonetheless, as the admissible sets of our controls determine the construction of the cost functional and thus, have a significant impact on our numerical results.

Theorem 3 *A local optimal solution,* $(C^*, v^*, \lambda^*) \in (W(0, T_{max}) \times V_{ad} \times \lambda_{ad}$ *exists such that our cost functional is minimized.*

Proof 3 As previously shown, there at least exists a non-empty set of triplets of state and controls which satsify our state constraints and control constraints:

$$A = \{(C, v, \lambda) \in (W(0, T_{max}) \times V_{ad} \times \lambda_{ad}) : e(C, v, \lambda) = 0\}. \tag{3.78}$$

We have $\iota = \inf\{J(C, v, \lambda) : (C, v, \lambda) \in A\} \geq 0$, due to the positivity of $J(C, v, \lambda)$. Assume that a minimizing sequence $(C_n, v_n, \lambda_n)_{n\in\mathbb{N}} \in A$ exists such that:

$$\lim_{n\to\infty} J(C_n, v_n, \lambda_n) = \iota. \tag{3.79}$$

Given that the sequence is bounded, in terms of the composite function space $(W(0, T_{max}) \times H^{2,1}(Q) \times H^1(0, T_{max}))$, then we infer the existence of a weakly converging subsequence $(C_{n_k}, v_{n_k}, \lambda_{n_k})$, where $k \in \mathbb{N}$. As $k \to \infty$, we have:

$$C_{n_k} \rightharpoonup C^* \quad \text{in } W(0, T), \tag{3.80}$$

$$v_{n_k} \rightharpoonup v^* \quad \text{in } H^{2,1}(Q), \tag{3.81}$$

$$\lambda_{n_k} \rightharpoonup \lambda^* \quad \text{in } H^1(0, T_{max}). \tag{3.82}$$

Now, if $v_{n_k} \in V_{ad}$ for all $k \in \mathbb{N}$, then we have:

$$v_{n_k} \in L^\infty(0, T_{max}; H^2(\Omega)) \cap H^1(0, T_{max}; L^2(\Omega)) \tag{3.83}$$

which is compactly embedded in $C(0, T_{max}; H^1(\Omega))$ according to Aubin's Lemma, following [56]. Thanks to the compact embedding, if this identity mapping is applied, then we can achieve strong convergence, such that for $k \to \infty$:

$$v_{n_k} \to v^* \quad \text{in } C(0, T_{max}; H^1(\Omega)). \tag{3.84}$$

Similarly for λ_{n_k}, we have that the space $H^1(0, T_{max})$ is compactly embedded in $L^2(0, T_{max})$ such that:

$$\lambda_{n_k} \to \lambda^* \quad \text{in } L^2(0, T_{max}). \tag{3.85}$$

By applying identity operators to the controls mapping from their originally chosen spaces to the spaces in which they can be compactly embedded, we are now able to pass to the limit in the weak formulation of the PIDE:

$$\int_0^{T_{max}} \int_\Omega (\lambda_{n_k}(1+m) + q)C_{n_k}\phi + (r - q - \lambda_{n_k}m + v_{n_k})C_{n_k x}\phi$$

$$+v_{n_k x}\phi C_{n_k x} + v_{n_k}\phi_x C_{n_k x} - \lambda_{n_k} \int_{x-x_{max}}^{x-x_{min}} C_{n_k}(x - y, T)e^{2y}\eta(e^y, T)dydxdT$$

$$\to \int_0^{T_{max}} \int_\Omega (\lambda^*(1+m) + q)C^*\phi + (r - q - \lambda^* m + v^*)C^*_x\phi$$

$$+v^*_x\phi C^*_x + v^*\phi_x C^*_x - \lambda^* \int_{x-x_{max}}^{x-x_{min}} C^*(x - y, T)e^{2y}\eta(e^y, T)dydxdT,$$

as $k \to \infty$ for every test function $\phi \in L^2(0, T_{max}; H_0^1(\Omega))$. Therefore, for this weakly convergent subsequence of controls and solution, our PIDE strongly converges as a member of the dual space of our test function. That is to say that $\lim_{k\to\infty} e_1(C_{n_k}, v_{n_k}, \lambda_{n_k}) = e_1(C^*, v^*, \lambda^*)$ in the space $(L^2(0, T_{max}; H_0^1(\Omega)))'$. Since our initial condition and boundary conditions are independent of our state and controls, we also have $e_2(C^*, v^*, \lambda^*) = 0$ and $e_3(C^*, v^*, \lambda^*) = 0$. As our cost functional is weakly lower semicontinuous, we have $J(C^*, v^*\lambda^*) \leq \lim_{k\to\infty} J(C_{n_k}, v_{n_k}, \lambda_{n_k})) = \iota$. As the bounded spaces for our controls V_{ad}, λ_{ad} are convex and closed in $H^{2,1}(Q)$ and $H^1(0, T_{max})$, respectively, and therefore weakly closed, then the weak limits v^*, λ^* exist in these spaces, respectively.

3.5 First-order necessary optimality conditions

First-order necesary conditions

The first order necessary conditions for an optimal solution (C^*, v^*, λ^*) along with associated Lagrangian multiplier functions (p^*, μ^*, v^*) can be summed up in terms of the Lagrangian:

$$\mathbb{L}_C(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta C = 0 \ \forall \delta C \in W(0, T), \tag{3.86}$$

$$\mathbb{L}_v(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta v \geq 0 \ \forall \delta v \in V_{ad}, \tag{3.87}$$

$$\mathbb{L}_\lambda(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta v \geq 0 \ \forall \delta \lambda \in \lambda_{ad}, \tag{3.88}$$

$$\mathbb{L}_p(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta p = 0 \ \forall \delta p \in L^2(0, T; L^2(\Omega)), \tag{3.89}$$

$$\mathbb{L}_v(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta v = 0 \ \forall \delta v \in L^2(\Omega), \tag{3.90}$$

$$\mathbb{L}_\mu(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta \mu = 0 \ \forall \delta \mu \in L^2(0, T). \tag{3.91}$$

Equations (3.86), (3.89), (3.90) and (3.91) simply restate the adherence to the state constraints. Equations (3.87) and (3.88) will be satisfied if the Lagrangian has reached a local minimum. More specifically:

$$\mathbb{L}_v(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta v > 0 \textbf{ if } v(x, T) = v_a \textbf{ or } v(x, T) = v_b, \tag{3.92}$$

$$\mathbb{L}_v(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta v = 0 \textbf{ if } v(x, T) \neq v_a \textbf{ and } v(x, T) \neq v_b, \tag{3.93}$$

$$\mathbb{L}_\lambda(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta \lambda > 0 \textbf{ if } \lambda(T) = \lambda_a \textbf{ or } \lambda(x, T) = \lambda_b, \tag{3.94}$$

$$\mathbb{L}_\lambda(C^*, v^*, \lambda^*, p^*, \mu^*, v^*)\delta \lambda = 0 \textbf{ if } \lambda(T) \neq \lambda_a \textbf{ or } \lambda(x, T) \neq \lambda_b. \tag{3.95}$$

As we have chosen to adopt a gradient descent method for our optimization algorithm, we must calculate the gradient of the Lagrangian with respect to each control in order to determine how to update our controls. This will require us to explicitly compute the Lagrangian multiplier functions at each iteration of our gradient descent numerical method. For brevity, we adopt the notation $\widetilde{v} = v - v_{init}$, where v_{init} is our prior estimate for v.

$$\mathbb{L}_v(C,v,\lambda,p,\mu,\upsilon)\delta v = \int_0^{T_{max}} \int_\Omega \alpha(\widetilde{v}_T\delta v_T + \widetilde{v}\delta v + \widetilde{v}_x\delta v_x + \widetilde{v}_{xx}\delta v_{xx}) + \delta v C_x p$$

$$+(\delta v p)_x C_x dx dT = \int_0^{T_{max}} \int_\Omega \alpha(\widetilde{v}_T\delta v_T + \widetilde{v}\delta v + \widetilde{v}_x\delta v_x + \widetilde{v}_{xx}\delta v_{xx})$$

$$+\delta v(C_x p - C_{xx}p)dx dT. \quad (3.96)$$

The function $(C_x p - C_{xx}p)$ can be identified with an element of the dual space $(H^{2,1}(Q))'$. We must map this function to a member of the space $H^{2,1}(Q)$ via the Riesz isomorphism defined by $R(F) : (H^{2,1}(Q))' \to (H^{2,1}(Q)))$, solving for u:

$$u_{TT} + F = u_{xx} - u_{xxxx} - u. \quad (3.97)$$

but practically speaking, this is a computationally expensive problem to solve on top of our coupled system. Therefore, in the numerical algorithm we simplify this by employing the Riesz operator for a weaker space $R(F) : (H^1(\Omega))' \to H^1(\Omega)$, solving for u:

$$-u_{xx} + u = F, \quad (3.98)$$

$$u_x(x_{min},.) = u_x(x_{max},.) = 0. \quad (3.99)$$

The solution to this stationary heat equation at every discrete point in time is taken as u_v and our directional derivative is taken to be:

$$\mathbb{L}_v(C,v,\lambda,p,\mu,\upsilon)\delta v = \langle \widetilde{v} + u_v, \delta v\rangle_{H^{2,1}(Q)}. \quad (3.100)$$

For the case of λ, we adopt the notation $\widetilde{\lambda} = \lambda - \lambda_{init}$, where λ_{init} is our prior estimate for λ:

$$\mathbb{L}_\lambda(C,\sigma,\lambda,p,\mu,\upsilon)\delta\lambda = \int_0^{T_{max}} \beta(\widetilde{\lambda}_T\delta\lambda_T + \widetilde{\lambda}\delta\lambda) + \delta\lambda(1+m)\int_\Omega C p dx - \delta\lambda m \int_\Omega C_x p dx$$

$$-\delta\lambda \int_\Omega \int_{x-x_{max}}^{x-x_{min}} C(T,x-y)e^{2y}\eta(e^y,T)dy p dx dT.$$

Similarly, we apply the Riesz isomorphism $R(F) : (H^1(0,T_{max}))' \to H^1(0,T_{max})$ to the term $F = (1+m)Cp - mC_x p - p\int_{x-x_{max}}^{x-x_{min}} C(T,x-y)e^{2y}\eta(e^y,T)dy$. The solution u_λ then forms part of the directional derivative, given by:

$$\mathbb{L}_\lambda(C,v,\lambda,p,\mu,\upsilon)\delta\lambda = \langle \widetilde{\lambda} + u_\lambda, \delta\lambda\rangle_{H^1(0,T_{max})}. \quad (3.101)$$

Beyond the regularizing effect of our cost functional, we observe that the gradients of our controls are highly dependent on the Lagrangian multiplier function p. It is possible to solve for this function at an intermediate stage between solving the state equation and updating our controls. The process for computing p is solving an adjoint PIDE, which forms a coupled system along with the state PIDE in this case. We derive this adjoint equation in the next subsection.

Adjoint equation

Now we consider how to calculate the Lagrangian multipliers associated with any given state and controls. We must solve an adjoint equation, which may be derived from our Lagrangian as detailed in this section. In practice we use these Lagrangian multipliers to determine the gradient vector used for updating our controls.

If we integrate $-\int_\Omega \frac{1}{2}v(T,x)pC_{xx}dx$ by parts, we obtain the following expression for the Lagrangian:

$$\begin{aligned}\mathbb{L}(\omega,\chi) = \frac{1}{2}||C(T_{max},.)-C_{obs}||^2_{L^2(\Omega)} + \frac{\alpha}{2}||v-v_{init}||^2_{H^{2,1}(Q)} + \frac{\beta}{2}||\lambda-\lambda_{init}||^2_{H^1(\Lambda)} \\ + \int_\Omega (C(0,.)-C_0)\upsilon dx + \int_0^{T_{max}} (C(.,x_{min})-C_D)\mu dT + \int_0^{T_{max}} \Big(\langle C_T, p\rangle_{V*,V} \\ +(\lambda(1+m)+q)\int_\Omega Cpdx + \int_\Omega (r-q-\lambda m+v)C_x p dx + (pv)_x C_x dx \\ -\lambda\int_\Omega \int_{x-x_{max}}^{x-x_{min}} C(x-y,.)e^{2y}\eta(e^y,T)dypdx\Big)dT.\end{aligned}$$

We have previously shown that the contents of the Lagrangian are continuously differentiable. Suppose that we have an optimal solution $\omega^* \in X$ which solves the parameter identification problem. Then it is necessary for a unique Lagrange multiplier $\chi^* = (p^*, \upsilon^*, \mu^*) \in Y$ to satisfy the adjoint equation:

$$\mathbb{L}_C(\omega^*,\chi^*)\delta C = 0 \ \forall \ \delta C \in W(0,T_{max}).$$

The above is equivalent to:

$$\begin{aligned}\int_\Omega (C^*(T)-C_{obs})\delta C(T)dx + \int_0^{T_{max}} \delta C(.,x_{min})\mu^* dT + \int_\Omega \delta C(0,.)\upsilon^* dx \\ + \int_0^{T_{max}} \langle \delta C_T, p^*\rangle_{V^*,V} dT + \int_0^{T_{max}}\int_\Omega ((1+m)\lambda^*+q)\delta C p^* dxdT \\ + \int_0^{T_{max}}\int_\Omega (r-q-\lambda^* m+\upsilon^*)\delta C_x p^* dxdT + \int_0^{T_{max}}\int_\Omega (\upsilon^* p^*)_x \delta C_x dxdT \\ - \int_0^{T_{max}} \lambda\int_\Omega\int_{x-x_{max}}^{x-x_{min}} \delta C(x-y,.)e^{2y}\eta(e^y,T)dyp^*dxdT = 0. \quad (3.102)\end{aligned}$$

This will also hold for all $\delta C(x,T) = \kappa(x)\theta(T)$ where $\kappa(x) \in H_0^1(\Omega)$ and $\theta(T) \in C_0^1(\Lambda)$. This reduces the above formulation to:

$$\begin{aligned}\int_0^{T_{max}}\int_\Omega \kappa\theta' p^* dxdT + \int_0^{T_{max}}\int_\Omega ((1+m)\lambda^*+q)\kappa\theta p^* dxdT \\ + \int_0^{T_{max}}\int_\Omega (r-q-\lambda^* m+\upsilon^*)\kappa'\theta p^* dxdT + \int_0^{T_{max}}\int_\Omega (\upsilon^* p^*)_x \kappa'\theta dxdT \\ - \int_0^{T_{max}} \lambda\int_\Omega\int_{x-x_{max}}^{x-x_{min}} \kappa(x-y)\theta e^{2y}\eta(e^y,T)dyp^*dxdT = 0. \quad (3.103)\end{aligned}$$

Addressing the components of the above formulation one by one:

$$\int_0^{T_{max}} \int_\Omega \kappa\theta' p^* dx dT = -\int_0^{T_{max}} \langle \kappa\theta, p_T^* \rangle_{H^{-1},H_0^1} dT = -\langle \int_0^{T_{max}} p_T^* \theta dT, \kappa \rangle_{H^{-1},H_0^1},$$

$$\int_0^{T_{max}} \int_\Omega ((1+m)\lambda^* + q)\kappa\theta p^* dx dT = \langle \int_0^{T_{max}} ((1+m)\lambda^* + q) p^* \theta dT, \kappa \rangle_{H^{-1},H_0^1},$$

$$\int_0^{T_{max}} \int_\Omega (r - q - \lambda^* m)\kappa'\theta p^* dx dT = -\langle \int_0^{T_{max}} (r - q - \lambda^* m) p_x^* \theta, \kappa \rangle_{H^{-1},H_0^1},$$

$$\int_0^{T_{max}} \int_\Omega v^* p^* \kappa'\theta dx dT = -\langle \int_0^{T_{max}} (v^* p^*)_x \theta, \kappa \rangle_{H^{-1},H_0^1},$$

$$\int_0^{T_{max}} \int_\Omega (v^* p^*)_x \kappa'\theta dx dT = -\langle \int_0^{T_{max}} (v^* p^*)_{xx} \theta dT, \kappa \rangle_{H^{-1},H_0^1},$$

$$\int_0^{T_{max}} \lambda^* \int_\Omega \int_{x-x_{max}}^{x-x_{min}} \kappa(x-y)\theta e^{2y} \eta(T, e^y) dy p^* dx dT$$
$$= \int_0^{T_{max}} \lambda^* \theta \int_\Omega \int_{x-x_{max}}^{x-x_{min}} \kappa(x-y) e^{2y} \eta(T, e^y) dy p^* dx dT.$$

We must use a change of variables $z = x - y$:

$$\int_0^{T_{max}} \lambda^* \theta(T) \int_\Omega \int_{x-x_{max}}^{x-x_{min}} \kappa(x-y) e^{2y} \eta(T, e^y) dy p^* dx dT$$
$$= \int_0^{T_{max}} \lambda^* \theta(T) \int_\Omega \int_{x_{min}}^{x_{max}} \kappa(z) e^{2x-2z} \eta(T, e^{x-z}) dz p^* dx dT.$$

We note that the end-points of integration of the inner-most integral are the boundary of Ω. Applying Fubini:

$$\int_0^{T_{max}} \lambda^* \theta \int_\Omega \int_\Omega \kappa(z) e^{2x-2z} \eta(T, e^{x-z}) dz p^* dx dT$$
$$= \int_0^{T_{max}} \lambda^* \theta(T) \int_\Omega \kappa(z) \int_\Omega p^* e^{2x-2z} \eta(T, e^{x-z}) dx dz dT$$
$$= \langle \int_0^{T_{max}} \lambda^* \theta \int_\Omega e^{2z-2x} \eta(T, e^{z-x}) p^*(z) dz dT, \kappa(x) \rangle_{H^{-1},H_0^1}. \quad (3.104)$$

Collecting these terms and substituting them into (3.103):

$$\langle \int_0^{T_{max}} \Big(- p_T^* - (v^* p^*)_{xx} + ((1+m)\lambda^* + q) p^* - (r - q - \lambda^* m) p_x^* - (v^* p^*)_x$$
$$- \lambda^* \int_\Omega e^{2z-2x} \eta(T, e^{z-x}) p^*(z) dz \Big) \theta(T) dT, \kappa(x) \rangle_{H^{-1},H_0^1} = 0. \quad (3.105)$$

Expanding the term $(v^* p^*)_{xx}$ yields:

$$- p_T^* - (v_{xx}^* p^* + 2 v_x^* p_x^* + v^* p_{xx}^*) + ((1+m)\lambda^* + q) p^* - (r - q - \lambda^* m) p_x^*$$
$$- (v^* p^*)_x - \lambda^* \int_\Omega e^{2z-2x} \eta(T, e^{z-x}) p^*(z) dz = 0. \quad (3.106)$$

Collecting terms by the partial derivatives of p^* simplifies the previous equation:

$$-p_T^* - v^* p_{xx}^* + ((1+m)\lambda^* + q - v_{xx}^* - v_x^*)p^* - (r - q - \lambda^* m + 2v_x^* + v^*)p_x^* \\ - \lambda^* \int_\Omega e^{2z-2x}\eta(T, e^{z-x})p^*(z)dz = 0. \quad (3.107)$$

If we substitute this relationship into (3.102) where $\delta C \in W(0,T)$ is arbitrarily chosen, then we must introduce the boundary terms from the integration by parts:

$$\langle (C^*(T_{max}) - C_{obs} + p^*(T_{max})), \delta C(T_{max})\rangle_{L^2(\Omega)} + \langle \mu^* - (v^* p^*)_x(., x_{min}), \delta C(., x_{min})\rangle_{L^2(\Lambda)} \\ + \langle (\upsilon^* - p^*(0)), \delta C(0, .)\rangle_{L^2(\Omega)} = 0. \quad (3.108)$$

Therefore we determine the initial and boundary conditions along with identities:

$$p^*(T_{max}) = -(C^*(T_{max}) - C_{obs}), \quad (3.109)$$

$$p^*(., x_{min}) = p^*(., x_{max}) = 0, \quad (3.110)$$

$$\mu^* = \frac{1}{2}(\sigma^2 p^*)_x(., x_{min}), \quad (3.111)$$

$$\upsilon^* = p^*(0). \quad (3.112)$$

For the case of calibrating to market data for n additional expiries, $0 < T_i < T_{max}$ for $i = 1, \ldots, n$, the right hand side of (3.107) at $T = \{T_1, \ldots, T_n\}$ is non-zero and is set to $-(C^*(T_i) - C_{obs}(T_i))$.

3.6 Optimization method, numerical experiments, results and extension of scheme

Gradient descent method

Here, we outline the optimization methodology. Clear definitions have been provided for our state equation and adjoint equation. Now we must outline the optimal control methodology for updating and calibrating our controls to the given option price data. The process of updating and calibrating our controls is performed by a gradient descent method. This requires us to calculate the gradient of our Lagrangian with respect to a given control. This gradient vector describes the direction of steepest descent for the cost functional which we are attempting to minimize. Therefore, the direction in which we update our control is fully determined by this gradient vector. It is then necessary to update the control by some given magnitude in this direction. This magnitude is described as the step-size of the gradient descent. The step-sizes for each control are given by $\Delta_v > 0$ and $\Delta_\lambda > 0$, respectively such that:

$$v_{n+1} = v_n - \Delta_v \nabla_v \mathbb{L}, \quad (3.113)$$

$$\lambda_{n+1} = \lambda_n - \Delta_\lambda \nabla_\lambda \mathbb{L},$$

where $\nabla_v \mathbb{L}$ and $\nabla_\lambda \mathbb{L}$ specify the vectors of directional derivatives of the Lagrangian with respect to our controls, respectively. The equations for updating either control are given in (3.113). We have taken v_{n+1} and λ_{n+1} to indicate the updated controls, while the previous estimate for the controls are given by v_n and λ_n, respectively.

Two choices that must be decided upon for this chosen methodology is the order in which we update these controls and the choice for the step-sizes. The choice of step-size at any given stage of the optimal control problem is quite non-trivial. If it is too small then we may not converge to the local minimum in sufficient time. If it is too large then we risk overstepping the local minimum. We have therefore chosen to treat this parameter dynamically for both controls, prescribing a heuristic step-size optimization method based on maximizing the reduction of the cost functional. For the order of control updates, we opt to update the controls in an alternating pattern.

Numerical implementation

Before discussing numerical experiments and results, a complete overview of the discretized optimal control methodology is necessary. We refer back to many equations presented earlier throughout this chapter here, although in this overview we are considering the discrete counterpart of many of these equations. We propose the following algorithm for a fixed pair of regularization parameters $\alpha > 0, \beta > 0$:

1. Assign parameters, initialize matrices and vectors, and collect observed data C_{obs}.
2. Set a priori guess for controls, $v = v_0$ and $\lambda = \lambda_0$. Set $n = 0$ and number of iterations to n_{max}
3. Solve state PIDE (3.6) for option price $C_n(x, T)$ using current control estimates v_n, λ_n using Euler's implicit finite difference method, estimating integral term using a trapezium method explicitly at previous time-step for simplicity.
4. Calculate cost functional $J(C_n, v_n, \lambda_n)$ from (3.18) and residual
$$\sum_{i=1}^{m} ||C(T_i)_n - C_{obs}(T_i)||_{l_2}.$$
   (a) If $n = n_{max}$ STOP.
5. Solve adjoint PIDE (3.107) for $p_n(x, T)$ with boundary conditions and final conditions specified in (3.110) and (3.109), respectively, using Euler's implicit finite difference method, estimating integral term using a trapezium method explicitly at previous time-step for simplicity.
6. Update control. Choose control according to an alternating pattern:
   (a) Approximate the gradient of the Lagrangian with respect to the chosen control.

(b) Solve the state PIDE (3.6) using Euler's implicit finite difference method using updated controls for three different step sizes and compute cost functional for each set of solution and controls: $J_{n,1}$,$J_{n,2}$, $J_{n,3}$

(c) Update the control according to whichever step-size has yielded the greatest reduction in the cost functional $\Delta J_n = |J_{n,i} - J_n|$. Increment n.

7. Go to step 4 (a)

The treatment of integral terms in both the state and the adjoint equations cases as non-homogeneous right hand side terms, explicitly calculated at the previous time step, makes the inversion of the sparse left hand side coefficient matrix far less computationally expensive. This approach also makes the numerical implementation of such a scheme more straightforward for the practitioner.

Numerical experiments

In this section we supply results from separate numerical experiments where we have calibrated to model generated price data and calibrated to empirical market data. The model generated price data is produced by simply prescribing plausible values for the parameters v_{pre} and λ_{pre}, and generating option price data at every point over the time-space cylinder of our discretized domain by taking the solution of our state PIDE. We initialize the parameters to $v_0 \neq v_{pre}$ and $\lambda_0 \neq \lambda_{pre}$, and attempt to calibrate these parameters to their respectively prescribed values via the proposed projected gradient method. This method of calibrating the controls to pre-determined known values allows us to test the quality of our methodology, thus removing any obfuscation caused by "noisey" market data. Once we are satisfied with the performance of our methodology for the artificial price data, we may then consider empirical market data and gauge the practical performance of our approach independently. In both cases we have tested a range of values for the regularization parameters to determine their effect on the calibration. The results in this section show that our method produces satisfactory reductions in the residual term while exhibiting stable behaviour for a wide range of regularization parameters in both sets of experiments.

Artficial data set

We set the current value of the underlying asset to $S = 300$ and uniformly discretize the domain of transformed strikes and expiries; $x_i \in [log(10), log(500)]$ and $T_j \in [0, 1]$, where $i = 1, \ldots, 50$ and $j = 1, \ldots, 25$. Computing numerical approximations for the solutions to the state equation, adjoint equation and the Riesz isomorphisms, as well as the implementation of a step-size decision making process involves solving many PDEs/PIDEs per iteration. This is a computationally expensive task for non-specific hardware. Therefore, for practical purposes, we consider using a relatively sparse mesh of uniformly discretized points for our numerical experiments. Here we have chosen a 50×25 grid in x and T, respectively.

The distribution η which describes the size of the random jump scaled as a proportion of the underlying asset value is taken to be log-normally distributed with mean $\mu = -0.85$ and standard deviation $\sigma = 0.45$. This corresponds to a high likelihood of a significant loss in value for the underlying asset in the occurrence of the Poisson event. The range of values for step-sizes Δ_v and Δ_λ have been experimentally determined to ensure stable behaviour while guaranteeing a significant reduction in the cost functional. Numerical experiments have revealed that excessively large step-size values result in unstable behaviour, while excessively small step-size values result in plateauing reductions in the cost functional at a premature stage of the method.

For this numerical experiment, we generate observed data by approximating numerical solutions to the state PIDE for a prescribed pair of parameters. The prescribed values for surface $v(x, T)$ and the curve $\lambda(T)$ are given in figure 3.1. The function chosen for $v(x, T)$

Parameter	Value
Asset value	$S = 300$
Interest rate	$r = 0.03$
Dividend rate	$q = 0.01$
Lognormal mean	$\mu = -0.85$
Lognormal standard deviation	$\sigma = 0.45$
v update step-size	$\Delta_v \in \{1.0e^{-3}, 5e^{-4}, 2.5e^{-4}\}$
λ update step-size	$\Delta_\lambda \in \{2e^{-4}, 1e^{-4}, 5e^{-3}\}$
x-range ($[x_{min} - x_{max}]$)	$x \in [log(10), log(500)]$
Final expiry	$T_{max} = 1$

Table 3.1: Input parameters for artificial data.

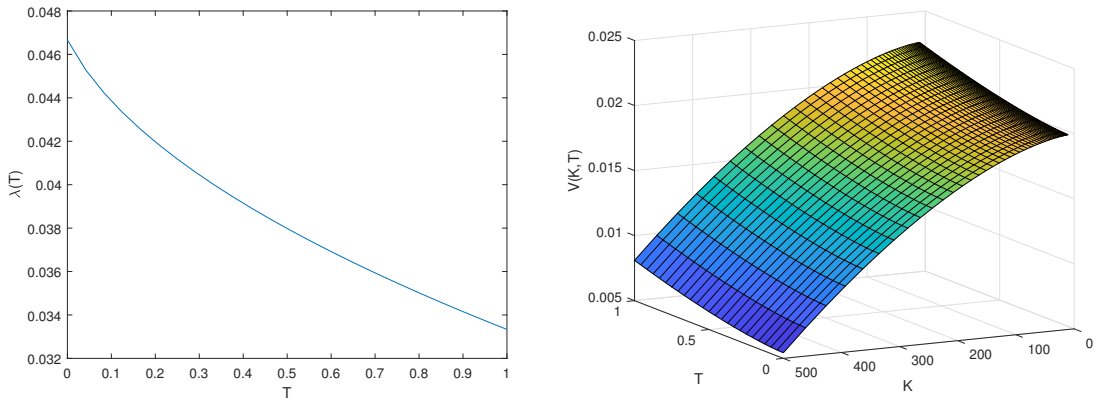

Figure 3.1: Our prescribed lambda surface on the left is generated by $\lambda(T) = 0.05 - \frac{(24T+1)^{0.5}}{300}$, while the prescribed value for v is generated by $v(x,T) = 0.02 - \frac{0.7e^{1.6x}}{10^6} + \frac{t^{1.5}}{375}$. Here we plot transformed $V(exp(x),T) = v(x,T)$ plotted against $K = exp(x)$ and T on the right.

reflects a typical volatility skew towards in the money strikes. In the case of a European Call, this corresponds to a skew towards lower strike values, as is commonly observed in markets. The curve for $\lambda(T)$ is skewed towards shorter expiries in this case, which yields higher prices for shorter term options. We include results for this typical case, although additional numerical experiments where $\lambda(T)$ has been skewed towards longer terms have yielded equally satisfactory fitting in both parameters. The initial guesses for our controls, v_{init} and λ_{init}, are given in figure 3.2. We have ensured that these values are sufficiently dissimilar from those used to generate the observed price data.

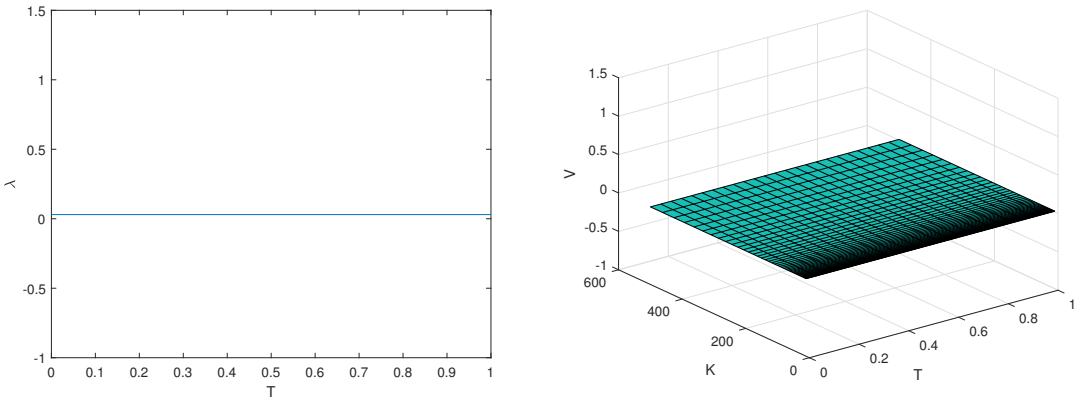

Figure 3.2: Initial guesses for our controls are $\lambda(T) = 0.02$, shown on the left, and $v(x,T) = 0.01$. Again, we plot transformed $V(exp(x),T) = v(x,T)$ plotted against $K = exp(x)$ and T on the right.

We generate price data by numerically solving the state PIDE over the entire domain of transformed strikes x_i, where $i = 1,\ldots,50$ for expiries $T_{13} = 0.5$ and $T_{25} = 1$. The contrast between the relative densities of the price data with respect to expiries and strikes is typical of market data, where prices are available for only a few expiries but for many strikes. While the calibration is driven by the minimization of our regularized cost functional, we assess the performance of our method by observing the reduction of the following residual:

$$\Delta_C = ||C(x_i,0.5) - C_{obs}(x_i,0.5)||_{l^2(\Omega)} + ||C(x_i,1) - C_{obs}(x_i,1)||_{l^2(\Omega)} \tag{3.114}$$

For our experiments we fix the regularization parameters $\alpha,\beta > 0$ and run the gradient descent numerical method for a fixed number of iterations. We have chosen a fixed number of iterations for our method in these experiments so that we may effectively measure the effect of our choices of regularization parameters on the associated calibrated parameters. In practice, one would likely enforce a stopping criteria based on the norms of successive differences in the parameters, the ratio of consecutive reductions of the cost functional or residual, or some absolute threshold of cost functional or residual.

In tables 3.2 and 3.3, we have recorded the evolution of the residual, Δ_C, and the cost functional, $J(C,v,\lambda)$, respectively, after successive iterations for particular combinations of α and β. We see that the rate of reduction of both of these metrics is highest at the start

of the scheme and plateaus as the scheme continues. In every case we observe that the sequences of residuals and cost functional evaluations are both monotonically decreasing as the scheme iterates. The slopes of reductions, in both cases, are approximately replicated across the range of choices for α and β. The results for the residual and cost functional after 1000 iterations are given in figure 3.3 across the whole domain of regularization parameters $(\alpha, \beta) \in \{10, 1, 0.1, 0.01, 0.001\} \times \{10, 1, 0.1, 0.01, 0.001\}$. The calibration was stable for every combination of α and β and we reduced the residual to less than 4% by the end of each run.

Iterations	0	25	50	100	200	300	500	1000
$\alpha = 10, \beta = 10$	5.96	3.40	2.40	1.26	0.46	0.31	0.26	0.20
$\alpha = 0.1, \beta = 0.1$	5.96	3.42	2.40	1.22	0.38	0.22	0.17	0.13
$\alpha = 0.001, \beta = 0.001$	5.96	3.42	2.40	1.22	0.40	0.24	0.19	0.14
$\alpha = 10, \beta = 0.001$	5.96	3.40	2.40	1.26	0.45	0.30	0.26	0.20
$\alpha = 0.001, \beta = 10$	5.96	3.42	2.41	1.23	0.40	0.25	0.19	0.14

Table 3.2: This table tracks the evolution of the residual, Δ_C, for successive iterations of our methodology for various combinations of regularization parameters, α and β.

Iterations	0	25	50	100	200	300	500	1000
$\alpha = 10, \beta = 10$	9.558	3.15	1.577	0.463	0.086	0.0507	0.039	0.029
$\alpha = 0.1, \beta = 0.1$	9.558	3.423	2.402	1.223	0.385	0.221	0.169	0.135
$\alpha = 0.001, \beta = 0.001$	9.558	3.424	2.404	1.224	0.397	0.243	0.185	0.138
$\alpha = 10, \beta = 0.001$	9.558	3.403	2.396	1.258	0.451	0.305	0.256	0.199
$\alpha = 0.001, \beta = 10$	9.558	3.424	2.405	1.228	0.402	0.246	0.186	0.138

Table 3.3: This table tracks the evolution of the cost functional, $J(C, v, \lambda)$, for successive iterations of our methodology for various combinations of regularization parameters, α and β. The starting value is the same due to the fact that we have set $v(x, T)$ and λ to constant values at the start of the scheme.

Predictably, the cost functional directly depends on α and β, as indicated in figure 3.3 on the right, while on the left the reduction in the residual depends greatly on our choice of α, while it is far less dependent on our choice of β. This should be attributed to higher price sensitivity with respect to small changes in $v(x, T) = \frac{1}{2}\sigma^2(x, T)$ than the jump frequency $\lambda(T)$. We observed that the residual grows for $\alpha > 1$ and that variation in the residual for $\alpha \leq 1$ was negligible.

Extending the complete domain of strikes past the region of interest to include a "buffer zone" reduces the effect of the calibration inertia at the boundaries on the calibration closer to the money at $S = 300$. On the other hand, the increase in the residual for large α can be explained by the downward movement of v for lower strikes sufficiently far from the lower boundary, where there is a higher density of points in our transformed domain. This

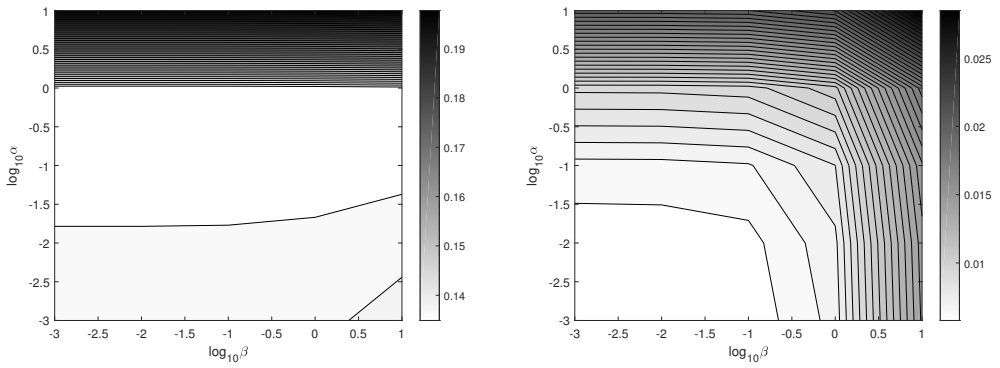

Figure 3.3: The figure on the left plots the contours of the residual against $\log_{10}\alpha$ and $\log_{10}\beta$, while the figure on the right plots the contours of the cost functional against $\log_{10}\alpha$ and $\log_{10}\beta$. The plots account for 25 results in both cases where $\alpha \in \{10, 1, 0.1, 0.01, 0.001\}$ and $\beta \in \{10, 1, 0.1, 0.01, 0.001\}$.

is shown in the top two graphs in figure 3.4. The dip forms part of an oscillating shape which is a manifestation of a high regularization penalty. The results for smaller values for α do not exhibit this behaviour, although they fail to capture the smoothness of the prescribed parameter at the money. This can also be said for the case of $\lambda(T)$, shown in figure 3.5. For calibrating to market prices, the parameters used in practice are unlikely to resemble the smoothness of our idealized prescribed parameters here, thus we extend our results to the case of empirical prices in the next set of results.

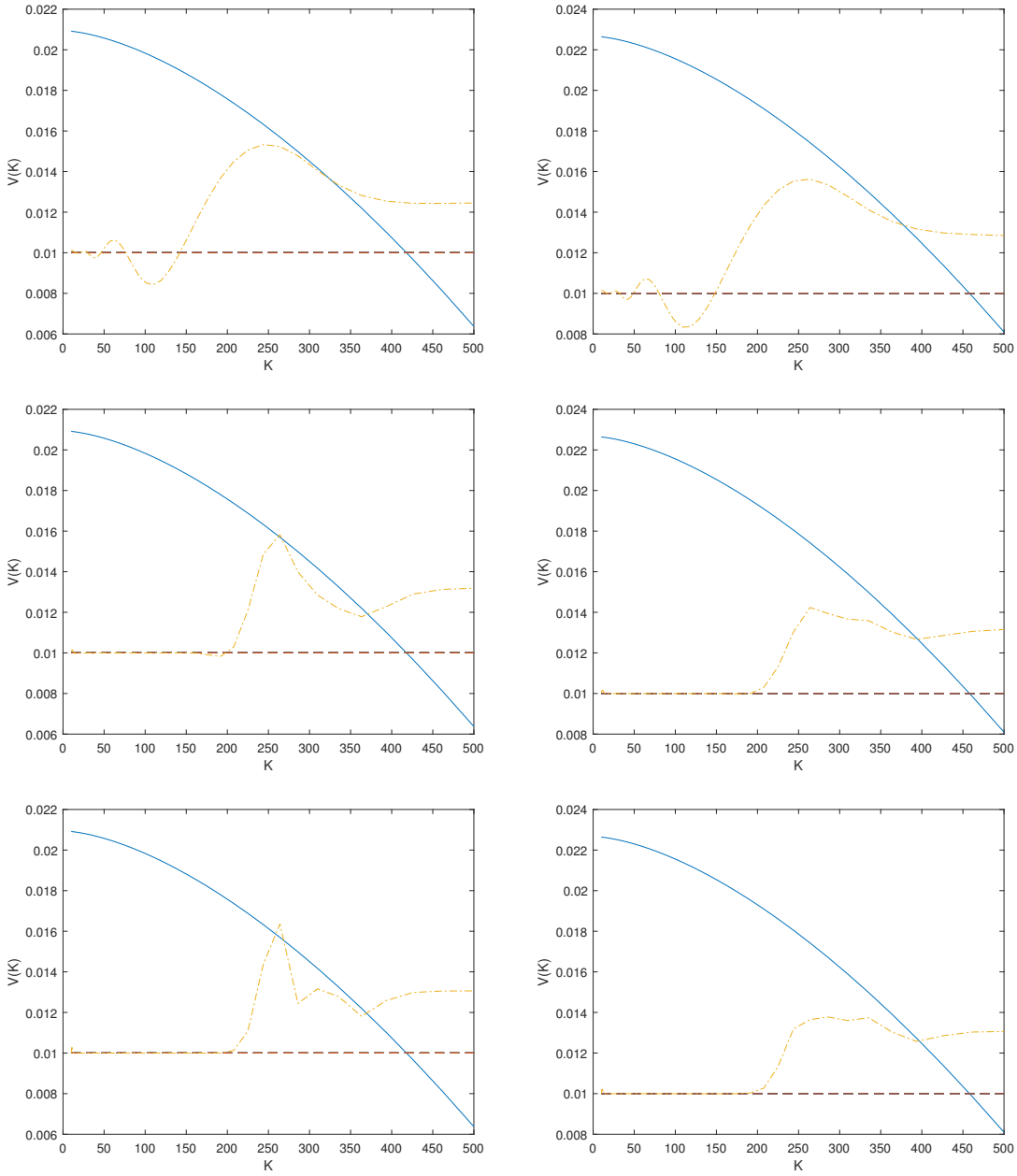

Figure 3.4: Here, we have plotted results for $V(\exp(x), T) = v(x, T)$ against $K = \exp(x)$, for $T = 0.5$ on the left and $T = 1$ on the right and for $\alpha = 10$, $\beta = 10$ on top, $\alpha = 0.1$, $\beta = 0.1$ in the middle and $\alpha = 0.001$, $\beta = 0.001$ below. The graphs plot the values used to generate the prices as the solid curve, our initial guess as the flat dashed line and the resulting calibrated curve after 1000 iterations as the dotted and dashed curve.

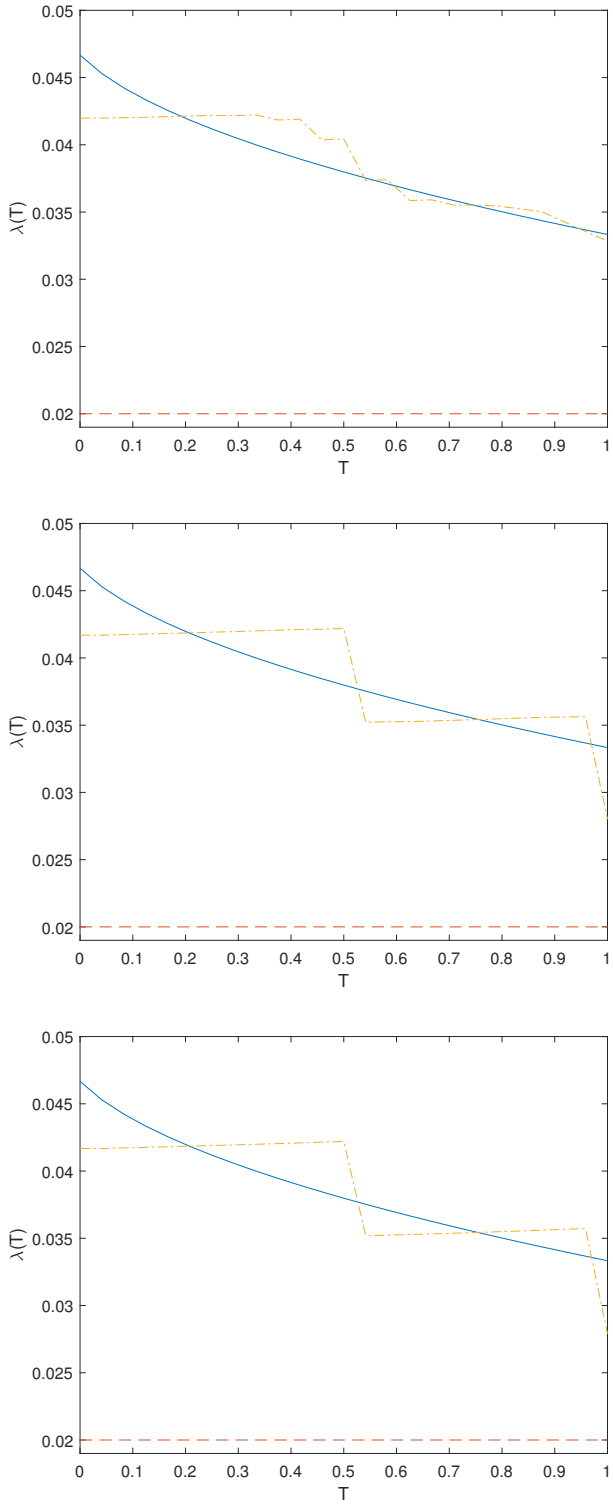

Figure 3.5: Here, we have plotted results for $\lambda(T)$ where $\alpha = 10$, $\beta = 10$ in the top graph, $\alpha = 0.1$, $\beta = 0.1$ in the middle graph and $\alpha = 0.001$, $\beta = 0.001$ in the graph at the bottom. The graphs plot the values used to generate the prices as the solid curve, our initial guess as the flat dashed line and the resulting calibrated curve after 1000 iterations as the dotted and dashed curve.

Historical market data

The market data is taken from FTSE index call options on 11/02/2000 from [14]. The prices are shown in table 3.5. The publicly known parameter values on this date for this FTSE index value are the underlying asset value, $S = 6219$, the riskless interest rate, $r = 0.061451$ and the dividend rate, $q = 0$. By calibrating exclusively to prices from a single day over the available range of strikes and expiries, the results of the calibration capture the dependence of the parameters on the strike price and the expiry according to the markets at that given time. The obvious disadvantage of using prices from a single day is that the market for option prices is incomplete over the domain of strikes and expiries which we wish to calibrate our parameters to. Typically, prices are particularly sparse in the domain of expiries. We could interpolate the price data to produce a more complete set of prices to calibrate to but, in general, this does not produce stable solutions.

One might also consider calibrating to today's prices for a single expiry over a range of strikes, as well as past prices for this fixed point in the future in order to expand the relative range of expiries for which prices are available. While this inclusion of past prices gives us a more complete domain of price data, the dynamics of an equity index such as the FTSE 100 appear to vary significantly during boom periods, compared to periods of recession or prolonged periods of uncertainty. Traders are likely to re-evaluate volatility and jump-likelihood on a regular basis. The regularity of this refactoring is arbitrary. Calibrating parameters to a mixture of past and present prices is therefore likely to produce erroneous results, with respect to the current market. For regular calibration, the adoption of previously calibrated parameters as prior guesses at the beginning of the next calibration is not necessarily an advantage, for the same reason.

Despite the sparseness of prices, we can enforce an arbitrary degree of regularity on our controls through our choices for regularization parameters, $\alpha > 0$ and $\beta > 0$, for the local volatility and jump-likelihood parameters, respectively.

K	$C_{obs}(T = 0.09589)$	K	$C_{obs}(T = 0.19178)$
5825	469.5	5725	631
6175	223.5	6175	314.5
6225	195.5	6225	284.5
6275	169	6275	256.5
6325	144.5	6325	229.5
6575	56.5	6725	72.5
7225	0.5	7025	24

Table 3.4: The set of observed data over various strikes at two expiries

Another caveat of our parameter calibration is updating the parameter $v(x, T)$ near the boundaries of Ω, where the boundary conditions of the underlying state equation are independent of $v(x, T)$. For accurate calibration of v to the empirical market data, we must consider choosing Ω such that it constructs a buffer zone around the strikes for which

Parameter	Value
Asset value	$S = 6219$
Interest rate	$r = 0.061451$
Dividend rate	$q = 0$
Lognormal mean	$\mu = -0.85$
Lognormal standard deviation	$\sigma = 0.45$
v update step-size	$\Delta_v \in \{1e^{-5}, 5e^{-6}, 2.5e^{-6}\}$
λ update step-size	$\Delta_\lambda \in \{2e^{-6}, 1e^{-6}, 5e^{-7}\}$
x-range ($[x_{min} - x_{max}]$)	$x \in [log(4050), log(8000)]$
Final expiry	$T_{max} = 0.19178$

Table 3.5: Input parameters for empirical market data.

we have empirical market data. Here we choose a domain for Ω that extends beyond the bounds of our available market data; $x \in \ln(4050), \ln(8000)$. To interpolate the available market data over our chosen domain of strikes, our method is as follows:

1. Use a cubic spline method to interpolate the observed price data over the spatial domain of discrete points that are bounded by the available data;
   $x \in [\ln(5725), \ln(7225)]$ for $T = 0.191781$ and $T = 0.0959$.

2. Use a Newton-Raphson method for finding the implied Black-Scholes volatility, $\sigma_{BS}(x, T)$ at every interpolated price over the domain of $x \in [\ln(5725), \ln(7225)]$ and $T \in \{0.191781, 0.0959\}$.

3. Set $\sigma_{BS}(x, T) = \sigma_{BS}(\ln(5725), T)$ for $x \in [\ln(4050), \ln(5725))$ and $\sigma_{BS}(x, T) = \sigma_{BS}(\ln(7225), T)$ for $x \in (\ln(7225), \ln(8000)]$ at $T = 0.191781$ and $T = 0.0959$.

4. Solve Black-Scholes for a European Call with the same parameters as our price data at every discrete point over $x \in [\ln(4050), \ln(5725)) \cup (\ln(7225), \ln(8000)]$ for $T = 0.191781$ and $T = 0.0959$. These prices along with the interpolated price data form a complete set of observed data, C_{obs}, for the domain of discrete points over $x \in [\ln(4050), \ln(8000)]$ and $T = 0.191781$ and $T = 0.0959$.

Our extension of the domain of strikes to include these buffer regions serve to minimize the truncation of our boundary conditions for the state PIDE but this requires us to address their effect on the calibration of our parameters. To resolve this, we simply set the values of parameter $v(x, T)$ at each iteration to the newly calibrated values of $v(ln(5725), T)$ and $v(ln(7225), T)$ for $x \in (x_{min}, ln(5725))$ and $x \in (\ln(7225), x_{max})\}$, respectively, thus imposing $\frac{\partial v}{\partial x} = 0$ over the buffer regions. We measure the cost functional and residual Δ_C only over the domain of interest where empirical price data is available. The initial guess, λ_{init}, for the lambda curve is set to $\lambda(T) = 0.02$. Along with the choices of $\mu = -0.85$ and $\sigma = 0.45$, this corresponds to a market crash every 50 years, which is plausible for

an initial guess. The range of step-sizes has been experimentally determined to balance stable behaviour with significant reductions in the residual and cost functional.

In tables 3.6 and 3.7, we have recorded the evolution of the residual, Δ_C, and the cost functional, $J(C, v, \lambda)$ after successive iterations for particular combinations of α and β. The results for the residual and cost functional after 1000 iterations are given in figure 3.6 for every combination of α and β where $(\alpha, \beta) \in \{10, 1, 0.1, 0.01, 0.001\} \times \{10, 1, 0.1, 0.01, 0.001\}$.

Iterations	0	25	50	100	200	300	500	1000
$\alpha = 10, \beta = 10$	6.91	4.48	3.20	1.86	1.32	1.50	1.57	1.78
$\alpha = 0.1, \beta = 0.1$	6.91	3.81	2.41	1.22	1.05	0.98	0.74	0.68
$\alpha = 0.001, \beta = 0.001$	6.91	3.81	2.42	1.23	1.04	0.97	0.69	0.52
$\alpha = 10, \beta = 0.001$	6.91	4.48	3.20	1.86	1.32	1.50	1.57	1.79
$\alpha = 0.001, \beta = 10$	6.91	3.81	2.42	1.23	1.04	0.97	0.69	0.51

Table 3.6: This table tracks the evolution of the residual, Δ_C, for successive iterations of our methodology for various combinations of regularization parameters, α and β.

Iterations	0	25	50	100	200	300	500	1000
$\alpha = 10, \beta = 10$	15.42	6.58	3.74	1.94	1.48	1.37	1.29	1.54
$\alpha = 0.1, \beta = 0.1$	14.17	4.23	1.62	0.43	0.32	0.27	0.17	0.17
$\alpha = 0.001, \beta = 0.001$	14.16	4.21	1.60	0.40	0.30	0.24	0.12	0.07
$\alpha = 10, \beta = 0.001$	15.42	6.58	3.74	1.94	1.46	1.34	1.25	1.48
$\alpha = 0.001, \beta = 10$	14.16	4.21	1.60	0.41	0.31	0.26	0.16	0.12

Table 3.7: This table tracks the evolution of the cost functional, $J(C, v, \lambda)$, for successive iterations of our methodology for various combinations of regularization parameters, α and β.

The calibration yielded stable parameters for every combination of α and β, although, as table 3.6 reveals, the residual reaches a lower bound between 100 and 300 iterations for the cases where $\alpha = 10$. For later iterations, further reductions in the cost functional are achieved through trading off reductions in the residual for regularity of the calibrated volatility parameter. This can only be traded off due to the high contribution to the cost functional made by the penalization of non-smooth $v(x, T)$ by a high value for α. As table 3.7 indicates, some time after 500 iterations, the scheme hits a local minimum for the cost functional and we observe a significant increase in this value by the end of 1000 iterations. In practice, a stopping criteria should be applied to prevent this increase, although for the sake of comparison we have chosen to run our scheme for a fixed number of iterations. For all other cases where $\alpha \leq 1$, we observe that the sequences of residuals and cost functional evaluations are both monotonically decreasing as the iterations of the scheme continue. Again, we observe a far greater dependence of the reduction of the residual on the value of

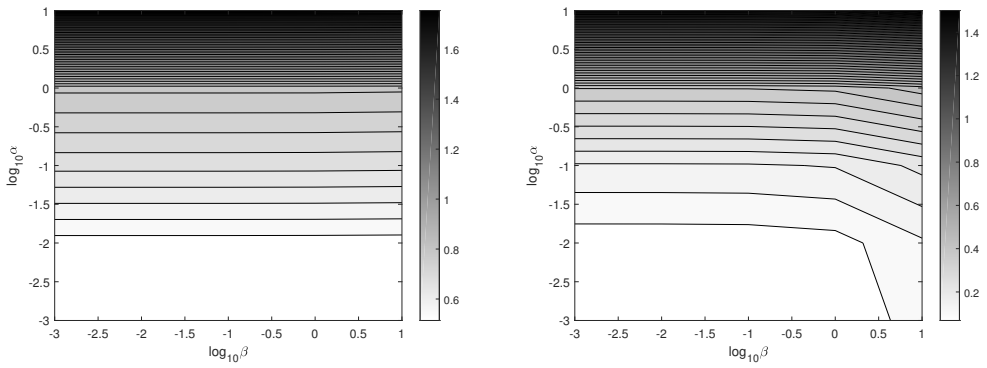

Figure 3.6: The figure on the left plots the contours of the residual against $\log_{10}\alpha$ and $\log_{10}\beta$, while the figure on the right plots the contours of the cost functional against $\log_{10}\alpha$ and $\log_{10}\beta$. The plots account for 25 results in both cases where $\alpha \in \{10, 1, 0.1, 0.01, 0.001\}$ and $\beta \in \{10, 1, 0.1, 0.01, 0.001\}$.

α than on β. The difference in the total reduction of the residual with respect to changes in α was far greater than for the idealized model-generated prices.

Again, the cost functional directly depends on α and β, as indicated in figure 3.6 on the right. The range of the final residual after 1000 iterations, as shown in the colour bar on the left of figure 3.6, goes from approximately 1.8 at worst to almost 0.5 at best, as α decreases. The vastly improved results for smaller α suggest that the market view of volatility might be irregular and that over-penalizing non-smoothness with high values for α may produce inaccurate results. One should be careful to balance this with the awareness of noise in the market data. If one knowingly calibrates to price data with high levels of noise, then high regularization values should be adopted to prevent over-fitting.

The effect of α on $v(x, T)$ is visualized clearly in the graphs on the right hand side of figure 3.8. For increasing α, this has the desired effect of increasing the smoothness of the calibrated surface. As mentioned previously, there is a trade-off between the regularity of the controls and how well we can fit them to the market data in this case. Conversely, the left hand side of figure 3.8 indicates that the calibration of β is almost independent of the value of β. This is likely explained by the sparsity of price data in the expiry domain. We calibrate by computing the gradient via the adjoint equation, which results in a relatively smooth $\lambda(T)$ in cases where price data is so sparse in the expiry domain, independently of our choice of β. For calibrating to price data for a larger number of expiries, we would expect to be able to more effectively control the regularity of the resulting jump frequency through our choice of β.

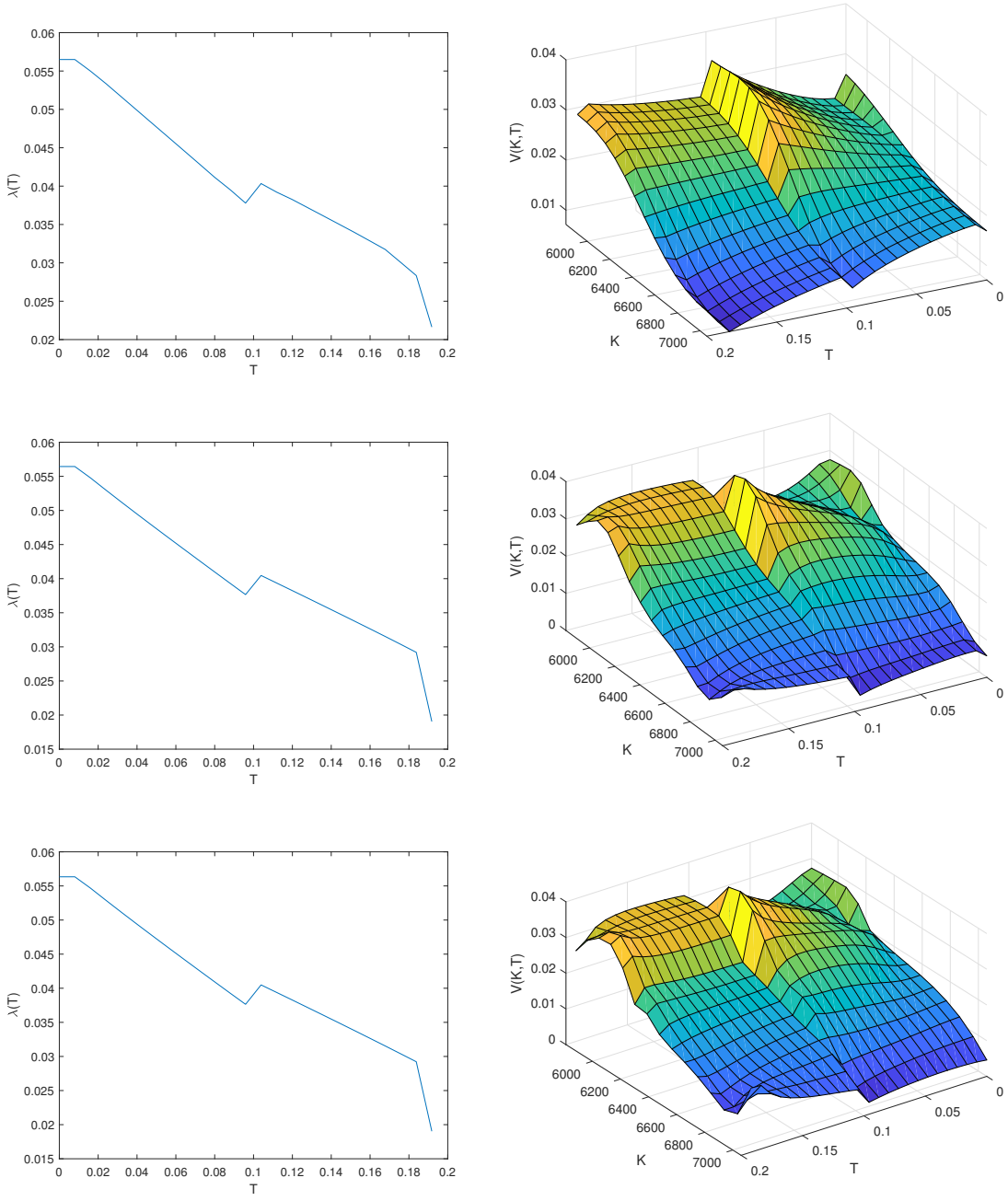

Figure 3.7: These are the resulting calibrated controls after 1000 iterations of our methodology for $\alpha = 10$, $\beta = 10$ on top, $\alpha = 0.1$, $\beta = 0.1$ in the middle and $\alpha = 0.001$, $\beta = 0.001$ below. The graphs on the left plot the calibrated value of $\lambda(T)$ against T, while the graphs on the right plot the calibrated $V(exp(x), T) = v(x, T)$ against $K = exp(x)$ and T.

Extension to additional class of options

Given that our model of the underlying asset follows a jump-diffusion path, we now consider the influence of the type of option which we take our price data from, on our results. We should be concerned that the European Call prices do not necessarily factor a jump-term into their chosen models. The sensitivity of various exotic option types with respect

to the balance of volatility and jump frequency, will differ from European Calls significantly. An improved scheme could leverage more than one option type in our calibration to produce a more complete evaluation of the jump frequency of the underlying. By choosing an additional class of option with a higher sensitivity to jumps in the underlying, this should theoretically improve the credibility of the calibrated $\lambda(T)$.

Barrier options, for example, use barrier values to activate or terminate the existence of the option depending on whether the value of the underlying falls above, below, inside or outside of deterministic barrier values over the lifetime of the option. An introductory exposure to barrier options is given in [74]. Including this class of options in the calibration would provide the scheme with additional information about the jump term due to the high sensitivity of the option value to jumps in the underlying.

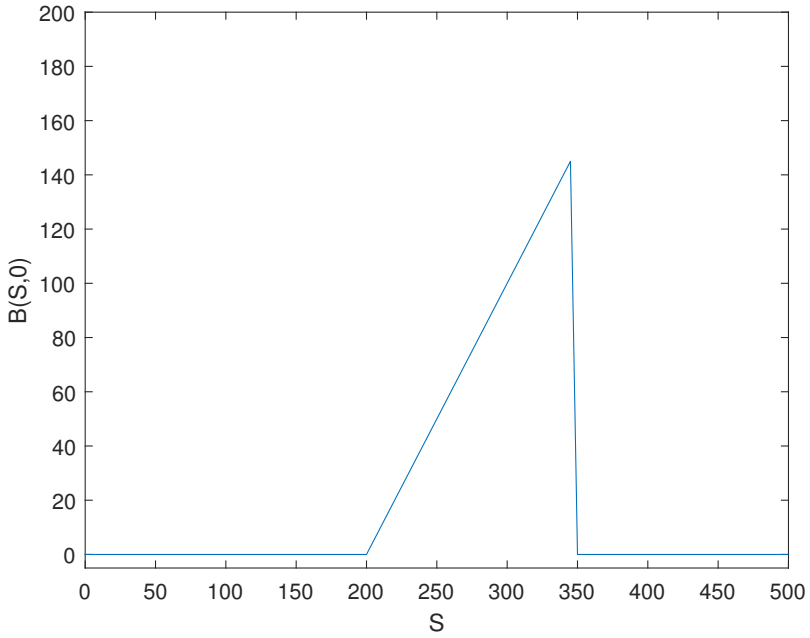

Figure 3.8: This is the payoff for a knock-out barrier call option with a single "up-and-out" time-dependent barrier $B_H(t)$, where $B_H(0) = 350$ and strike $K = 200$. The term "knock-out" is used to indicate that if $S(t) > B_H(t)$ for any $t \in (0, T]$ then the value of the option goes to 0 and the contract expires instantly. The value of such an option is much more sensitive to frequent jumps in the value of the underlying asset than its European counterpart, making it an ideal additional class of option to calibrate $\lambda(T)$ to.

In order to extend our scheme, we require an additional forward equation to model the barrier option price as a function of strike and expiry. A forward equation for knock-out barrier options is proposed in [75], for example. If we want to calibrate to knock-out call prices as well as european call prices on the same underlying, the cost functional should be expanded to include the norm of the differences between the barrier call price data $B_{obs}(T_j)$ for $j \in \{1, \ldots, m\}$, where data is available over $m > 0$ expiries, and the solution to the corresponding forward equation, $B(K, T_j)$, over the domain of expiries Ω:

$$J(C,B,v,\lambda) = \sum_{i=1}^{n} \frac{1}{2}||C(T_i) - C_{obs}(T_i)||^2_{L^2(\Omega)} + \sum_{j=1}^{m} \frac{1}{2}||B(T_j) - B_{obs}(T_j)||^2_{L^2(\Omega)} + \frac{\alpha}{2}||v - v_{init}||^2_{H^{2,1}(Q)} + \frac{\beta}{2}||\lambda - \lambda_{init}||^2_{H^1(0,T_{max})}.$$

The forward equation for $B(K,T)$, along with the lower boundary condition $B_D(T)$ and an initial condition $B_0(K)$, are treated as extra equality constraints in the optimization problem. We expand our Lagrangian to include the weak formulations of these additional constraints. $\widetilde{\omega} = (B, v, \lambda)$ form the triplet of the state for the barrier call model and parameters we are calibrating. e_1, e_2 and e_3 are the operators for each weak formulation of the additional equality constraints, with their associated Largrangian multiplier functions, $\{\chi_i\}_{i=1}^3$:

$$\mathbb{L}(\omega,\chi) = \sum_{i=1}^{n} \frac{1}{2}||C(T_i) - C_{obs}(T_i)||^2_{L^2(\Omega)} + \sum_{j=1}^{m} \frac{1}{2}||B(T_j) - B_{obs}(T_j)||^2_{L^2(\Omega)} + \frac{\beta}{2}||\lambda - \lambda_{init}||^2_{H^1(0,T_{max})} + \int_\Omega (C(x,0) - C_0)\upsilon dx + \int_0^{T_{max}} (C(T,x_{min}) - C_D)\mu dT + \int_0^{T_{max}} \Big(< C_T, p >_{V*,V} + (\lambda(T)(1+m(T)) + q(T)) \int_\Omega C p dx + \int_\Omega (r(T) - q(T) - \lambda(T)m(T) + v(x,T))C_x p dx - \int_\Omega v(x,T) p C_{xx} dx - \lambda(T) \int_\Omega \int_{x-x_{max}}^{x-x_{min}} C(T, x-y) e^{2y} \eta(e^y, T) dy p dx \Big) dT + e_1(\widetilde{\omega}, \widetilde{\chi_1}) + e_2(\widetilde{\omega}, \widetilde{\chi_2}) + e_3(\widetilde{\omega}, \widetilde{\chi_3}), \quad (3.115)$$

where $\chi = (p, \upsilon, \mu, \chi_1, \chi_2, \chi_3)$ and $\omega = (C, B, v, \lambda)$. Naturally this would require the derivation of an additional adjoint equation which must be solved separately at each iteration in order to compute the direction of steepest descent for updating our parameters. The derivation of this adjoint should follow the same approach as section 3.5. The numerical implementation of this coupled system of state PIDEs and adjoint equations poses a significant technical challenge, however theoretically speaking it is a simple extension of our existing scheme. As such, we leave such an extension for possible future work.

Chapter 4

Summary and Conclusion

By combining fourth-order (compact and non-compact) finite difference schemes in space with Hundsdorfer and Verwer's second-order ADI time-stepping scheme, we have constructed a new numerical method for solving option pricing problems for stochastic volatility models. Numerical experiments for approximating the price of a European Put option using Heston's stochastic volatility model with generic parameters confirm the numerical convergence of the scheme in space and time while the results for a wide range of parabolic mesh ratios suggest good stability properties. A further consideration for the work completed in this sphere could be to extend this scheme to the pricing of American options, a class of option pricing models which allows for an early exercise.

We have provided a result for the existence of a solution to the forward jump-diffusion equation. We have outlined an optimal control methodology for simultaneously calibrating a transformed local volatility surface and a Poisson jump parameter to the forward jump-diffusion option pricing model, with respect to observed option price data. We have provided a Tikhonov regularized cost functional and proven the existence of locally optimal parameters for minimizing this cost functional. We have established first-order necessary conditions of optimality and derived a corresponding adjoint equation with a strong emphasis on functional analysis. We have suggested and applied a gradient descent method to solve the optimal control problem, where results from numerical experiments for artificial data and historical market data support the method for practical use. A deeper exploration of the optimal control approach to calibrating model parameters for the forward jump-diffusion equation could investigate second order sufficient conditions for locally optimal solutions and implement an efficient numerical method which leverages these conditions. We also have discussed the possibility of extending the scheme to calibrate to an additional class of option which is more sensitive to jumps in the value of the underlying.

Chapter 5

Bibliography

[1] L. Andersen and L. Andreasen. Jump-Diffusion Processes: Volatility Smile Fitting and Numerical Methods for Option Pricing. *Review of Derivatives Research*, **4**(3), 231–262, 2000.

[2] M. Avellaneda, C. Friedman, R. Holmes and D. Samperi. Calibrating volatility surfaces via relative entropy minimization. *Appl. Math. Finance*, **4**(1), 37-–64, 1997.

[3] R.M. Beam and R.F. Warming. Alternating direction implicit methods for parabolic equations with a mixed derivative. *Siam J. Sci. Stat. Comput.*, **1**(1), 131–159, 1980.

[4] E. Benhamou, E. Gobet, and M. Miri. Time dependent Heston model. *SIAM J. Finan. Math.*, **1**, 289–325, 2010.

[5] F. Black and M. Scholes. The pricing of options and corporate liabilities. *J. Polit. Econ.*, **81**, 637–659, 1973.

[6] I. Bouchouev and V. Isakov. Uniqueness, stability and numerical methods for the inverse problem that arises in financial markets. *Inverse Problems* **15**(3), 95—116, 1999.

[7] R.P. Brent. An algorithm with guaranteed convergence for finding a zero of a function. *Computer Journal*, **14**(4), 422–425, 1971.

[8] P. Carr and L. Cousot. A PDE approach to jump-diffusions. *J. Quant. Fin.*, **11**(1), 33–52, 2011.

[9] P. Christoffersen, K. Jacobs and K. Mimouni. Volatility dynamics for the S&P500: evidence from realized volatility, daily returns, and option prices. *Review of Financial Studies*, **23**(8), 3141–3189, 2010.

[10] N. Clarke and K. Parrott. Multigrid for American option pricing with stochastic volatility. *Appl. Math. Finance* **6**(3), 177–195, 1999.

[11] R. Cont, P. Tankov and E. Voltchkova. Option Pricing Models with Jumps: Integro-Differential Equations and Inverse Problems. *European Congress on Computational Methods in Applied Sciences and Engineering*, 24–28, 2004.

[12] R. Cont and P. Tankov. Retrieving Lévy Processes from Option Prices: Regularization of an Ill-Posed Inverse Problem. *SIAM J. Cont. and Opt.*, **45**(1), 1-–25, 2006.

[13] S. Crépey. Calibration of the Local Volatility in a Generalized Black-Scholes Model Using Tikhonov Regularization. *SIAM J. Math. Anal.*, **34**(5), 1183–1206, 2003.

[14] S. Crépey. Calibration of the local volatility in a trinomial tree using Tikhonov regularization. *Inv. Problems*, **19**(1), 91–127, 2003.

[15] J. Douglas. Alternating direction methods for three space variables. *Numer. Math.*, **4**, 41–63, 1962.

[16] J. Douglas and J. E. Gunn. A general formulation of alternating direction methods. *Numer. Math.*, **6**(1), 428–453, 1964.

[17] J. Duan. The GARCH option pricing model. *Math.Finance*, **5**(1), 13–32, 1995.

[18] B. Dupire. Pricing with a smile. *Risk* **7**, 18–20, 1994.

[19] B. Düring, M. Fournié and A. Jüngel. High-order compact finite difference schemes for a nonlinear Black-Scholes equation. *Intern. J. Theor. Appl. Finance*, **6**(7), 767–789, 2003.

[20] B. Düring, M. Fournié and A. Jüngel. Convergence of a high-order compact finite difference scheme for a nonlinear Black-Scholes equation. *Math. Mod. Num. Anal.*, **38**(2), 359–369, 2004.

[21] B. Düring, A. Jüngel and S. Volkwein. Sequential Quadratic Programming Method for Volatility Estimation in Option Pricing. *J. Optim. Theory & Appl.* , **139**(3), 515-–540 2008.

[22] B. Düring. Asset pricing under information with stochastic volatility. *Rev. Deriv. Res.* **12**(2), 141–167, 2009.

[23] B. Düring and M. Fournié. High-order compact finite difference scheme for option pricing in stochastic volatility models. *J. Comput. Appl. Math.*, **236**(17), 4462–4473, 2012.

[24] B. Düring, M. Fournié, and A. Rigal. High-order ADI schemes for convection-diffusion equations with mixed derivative terms. *Spectral and High Order Methods for Partial Differential Equations - ICOSAHOM'12*, M. Azaïez et al. (eds.), 217–226, Lecture Notes in Computational Science and Engineering 95, Springer, Berlin, Heidelberg, 2013.

[25] B. Düring, M. Fournié and C. Heuer. High-order compact finite difference schemes for option pricing in stochastic volatility models on non-uniform grids. *J. Comput. Appl. Math.*, **271**(18), 247–266, 2014.

[26] B. Düring and C. Heuer. High-order compact schemes for parabolic problems with mixed derivatives in multiple space dimensions. *SIAM J. Numer. Anal.*, **53**(5), 2113–2134, 2015.

[27] M. Fournié and A. Rigal. High order compact schemes in projection methods for incompressible viscous flows. *Commun. Comput. Phys.*, **9**(4), 994–1019, 2011.

[28] M.M. Gupta, R.P. Manohar and J.W. Stephenson. A single cell high-order scheme for the convection-diffusion equation with variable coefficients. *Int. J. Numer. Methods Fluids*, **4**, 641–651, 1984.

[29] B. Gustafsson. The convergence rate for difference approximation to general mixed initial-boundary value problems. *SIAM J. Numer. Anal.* **18**(2), 179–190, 1981.

[30] C. He, J.S. Kennedy, T.F. Coleman, P.A. Forsyth, Y. Li and K.R. Vetzal. Calibration and hedging under jump diffusion. *Review of Derivatives Research*, **9**(1), 1—35. 2006.

[31] C. Hendricks, M. Ehrhardt and M. Günther. High-order-compact ADI schemes for diffusion equations with mixed derivatives in the combination technique. *J. Appl. Num. Math.*, **101**(3), 36–52, 2016.

[32] S.L. Heston. A closed-form solution for options with stochastic volatility with applications to bond and currency options. *Review of Financial Studies* **6**(2), 327–343, 1993.

[33] N. Hilber, A. Matache and C. Schwab. Sparse wavelet methods for option pricing under stochastic volatility. *J. Comput. Financ.*, **8**(4), 1–42, 2005.

[34] K.J. in't Hout and S. Foulon. ADI finite difference schemes for option pricing in the Heston model with correlation. *Int. J. Numer. Anal. Mod.* **7**(2), 303–320, 2010.

[35] K.J. in't Hout and B.D. Welfert. Stability of ADI schemes applied to convection-diffusion equations with mixed derivative terms. *Appl. Num. Math.* **57**(1), 19–35, 2007.

[36] W. Hundsdorfer. Accuracy and stability of splitting with stabilizing corrections. *Appl. Num. Math.*, **42**, 213–233, 2002.

[37] W. Hundsdorfer and J.G Verwer. Numerical solution of time-dependent advection-diffusion-reaction equations. Springer Series in Computational Mathematics, **33**, Springer-Verlag, Berlin, 2003.

[38] S. Ikonen and J. Toivanen. Efficient numerical methods for pricing American options under stochastic volatility. *Numer. Methods Partial Differential Equations* **24**(1), 104–126, 2008.

[39] S.G. Kou. A Jump-Diffusion Model for Option Pricing. *Management Science*, **48**(8), 1086–1101, 2002.

[40] H.O. Kreiss, V. Thomée and O. Widlund. Smoothing of initial data and rates of convergence for parabolic difference equations. *Comm. Pure Appl. Math.*, **23**, 241–259, 1970.

[41] D.M. Pooley, K.R. Vetzal and P.A. Forsyth. Convergence remedies for non-smooth payoffs in option pricing. *J. Comp. Fin.* , **6**(4), 25–40, 2003.

[42] H. Kushner. Stochastic stability & control. *Academic Press*, **33**(1), 12–22, 1967.

[43] R. Lagnado and S. Osher. A technique for calibrating derivative security pricing models: numerical solution of the inverse problem. *J. Comput. Finance*, **1**(1), 13-–25, 1997.

[44] D. Lanser, J, Blom and J. Verwer. Time integration of the shallow water equations in spherical geometry. *J. Comp. Phys.* **171**, 373–393, 2001.

[45] H.A. Latane and R.J. Rendleman, Jr. Standard deviations of stock price ratios implied in option prices. *J. Finance* **31**(2), 369–381, 1976.

[46] A.L. Lewis. *Option valuation under stochastic volatility.* Finance Press, Newport Beach, CA, 2000.

[47] W. Liao and A.Q.M. Khaliq. High-order compact scheme for solving nonlinear Black-Scholes equation with transaction cost. *Int. J. Comput. Math.*, **86**(6), 1009–1023, 2009.

[48] S. Manaster and G. Koehler. The calculation of implied variances from the black-scholes model: a note. *J. Finance*, **37**(1), 227–230, 1982.

[49] R. Merton. Option Pricing when Underlying Stock Returns are Discontinuous. *J. Fin. Econ.* **3**(1-2), 125—144, 1976.

[50] A. R. Mitchell and G. Fairweather. Improved forms of the alternating direction methods of Douglas, Peaceman, and Rachford for solving parabolic and elliptic equations. *Numer. Math.* **6**(1), 285–292, 1964.

[51] D.W. Peaceman and H.H. Rachford Jr. The numerical solution of parabolic and elliptic differential equations. *J. Soc. Ind. Appl. Math.* **3**(1), 28–41, 1959.

[52] A. Rigal. Schémas compacts d'ordre élevé: application aux problèmes bidimensionnels de diffusion-convection instationnaire I. *C.R. Acad. Sci. Paris. Sr. I Math.*, **328**(6), 535–538, 1999.

[53] M.J. Ruijter, C.W. Oosterlee. Two-dimensional Fourier cosine series expansion method for pricing financial options. *SIAM J. Sci. Comp.* **34**(5), 642–671, 2012.

[54] E. Sachs and M. Schu. Gradient Computation for Model Calibration with Pointwise Observations. Control and Optimization with PDE Constraints, Springer, 117–136, 2013.

[55] D. Shimko. Bounds of probability. *Risk*, **6**, 33–37, 1993.

[56] J. Simon. Compact sets in the space $L_p(O,T;B)$. *Annali di Matematica Pura ed Applicata*, **146**, 65-–96, 1986.

[57] W.F. Spotz and C.F. Carey. Extension of high-order compact schemes to time-dependent problems. *Numer. Methods Partial Differential Equations* **17**(6), 657–672, 2001.

[58] P. Tankov and E. Voltchkova. Jump-diffusion models: a practitioner's guide. *Banque et Marchés*, **99**, 2009.

[59] D. Tavella and C. Randall. *Pricing financial instruments: the finite difference method.* John Wiley & Sons, 2000.

[60] A. Tikhonov and V. Stepanov. Numerical Methods for the Solution of Ill-Posed Problems. Wiley, 1977.

[61] F. Tröltzsch. Optimal Control of Partial Differential Equations: Theory, Methods, and Applications. Springer, 2010.

[62] W. Zhu and D.A. Kopriva. A spectral element approximation to price European options with one asset and stochastic volatility. *J. Sci. Comput.* **42**(3), 426–446, 2010.

[63] R. Zvan, P.A. Forsyth and K.R. Vetzal. Penalty methods for American options with stochastic volatility. *J. Comp. Appl. Math.*, **91**(2), 199–218, 1998.

[64] H. Egger and H.W. Engl. Tikhonov regularization applied to the inverse problem of option pricing: convergence analysis and rates. *Inverse Problems*, **21**(3), 1027—1045, 2005.

[65] R. Cont and P. Tankov. Retrieving Lévy processes from option prices: regularization of an ill-posed inverse problem. *SIAM J. Cont and Opt.* , **45**(1), 1–25, 2006.

[66] Y. Achdou, G. Indragoby and O. Pironneau. Volatility calibration with American options. *Methods Appl. Anal.* , **11**(4), 533–556., 2004.

[67] Y. Achdou. An Inverse Problem for a Parabolic Variational Inequality Arising in Volatility Calibration with American Options. *SIAM Journal on Control and Optimization*, **43**(5), 1583–1615, 2005.

[68] N. Jackson, E. Süli and S.D. Howison. Computation of deterministic volatility surfaces. *J. Comp. Fin.* , **2**(2), 5–32, 1999.

[69] A.N. Tikhonov and V.Y. Arsenin. Solutions of ill-posed problems. W.H. Winston and Sons, 1977.

[70] B. Düring and A. Jüngel. Existence and uniqueness of solutions to a quasilinear parabolic equation with quadratic gradients in financial markets. *Nonlinear Analysis: Theory, Methods and Applications*, **62**(3), 519–544, 2005.

[71] L. C. Evans. Partial Differential Equations. Springer. 2010

[72] A.M. Matache, T. Von Petersdorff and C. Schwab. Fast Deterministic Pricing of Options on Levy Driven Asset. *ESIAM Math. Model. and Num. Anal*, **38**(1), 37–71, 2004.

[73] H. W. Engl, M. Hanke and A. Neubauer. Regularization of Inverse Problems. Springer, 2000.

[74] J. C. Hull. Options, Futures, and Other Derivatives Securities. Prentice Hall, 2003.

[75] P. P. Carr and A. Hirsa. Forward Evolution Equations for Knock-Out Options. Advances in Mathematical Finance, 195–217. Springer, 2007.