Implementation of Bivariate Unspanned Stochastic Volatility Models

Cian Cullinan

A dissertation submitted to the Faculty of Commerce, University of Cape Town, in partial fulfilment of the requirements for the degree of Master of Philosophy.

July 16, 2018

MPhil in Mathematical Finance,
University of Cape Town.

Declaration

I declare that this dissertation is my own, unaided work. It is being submitted for the Degree of Master of Philosophy to the University of Cape Town. It has not been submitted before for any degree or examination to any other university.

_____

July 16, 2018

Abstract

Unspanned stochastic volatility term structure models have gained popularity in the literature. This dissertation focuses on the challenges of implementing the simplest case – bivariate unspanned stochastic volatility models, where there is one state variable controlling the term structure, and one scaling the volatility. Specifically, we consider the Log-Affine Double Quadratic (1,1) model of Backwell (2017). In the class of affine term structure models, state variables are virtually always spanned and can therefore be inferred from bond yields. When fitting unspanned models, it is necessary to include option data, which adds further challenges. Because there are no analytical solutions in the LADQ (1,1) model, we show how options can be priced using an Alternating Direction Implicit finite difference scheme. We then implement an Unscented Kalman filter — a non-linear extension of the Kalman filter, which is a popular method for inferring state variable values — to recover the latent state variables from market observable data.

Acknowledgements

I would like to acknowledge and express my gratitude to my supervisor Alex Backwell for his advice and guidance throughout. I am also grateful to my friends and family for their support.

Contents

List of Figures

Chapter 1

Introduction

One typically expects term structure models to be complete, in the sense that contingent claims can be hedged solely with primary interest rate instruments such as zero-coupon bonds (ZCBs) or swaps (Backwell, 2015). However, papers by Collin-Dufresne and Goldstein (2002) and Heidari and Wu (2003) present evidence for *unspanned* factors — factors that are not related to the yield curve. This has seen the development of a number of models for the short rate that incorporate unspanned factors. Most commonly, these factors are introduced in the form of *unspanned stochastic volatility* (USV) state variables. The simplest class of such models are *bivariate USV models*. In keeping with the literature, we refer to these models as *(1,1) models*, since they contain one spanned factor (i.e. a factor that does affect the yield curve) and one USV factor. Specifically, this dissertation will focus on implementation of the *Log-Affine Double Quadratic* — or LADQ — $(1, 1)$ model of Backwell (2017).

Term structure modelling and their subsequent implementation is important in the field of risk management, among others. As such there is a vast literature on the topic. A large portion of this literature is dedicated to the class of *affine term structure models* (ATSMs), originally characterised by Duffie and Kan (1996). Under this specification, bond yields are affine functions of the state variables. A common technique for fitting ATSMs is the Kalman filter. In contrast to stock price modelling, the state variables are typically not directly observable in the market. We therefore refer to them as latent (or hidden) states. The Kalman filter allows for the recovery of the latent state variable process from panel data of market observables. It is characterised by two equations: the *measurement* equation, linking the latent states to the market observables, and the *state transition* equation describing how the states evolve through time. It is required that these functions be linear, which holds when considering ATSMs (at least for the measurement equation), as yields are affine functions of the short rate. Under the LADQ (1,1) model, the unspanned nature of the volatility makes it necessary to include options in the observation set

to recover the USV process. This introduces two complexities. Firstly, we must be able to price options which, under the LADQ (1,1) model (or any (1,1) model), have no analytical solutions. Secondly, the measurement equation of the filter is no longer linear with the addition of option prices. To implement the LADQ (1,1) model, we must be able to price options and extend the Kalman filter to incorporate non-linearity.

Fourier pricing methods are not possible as the characteristic function is not available in the LADQ (1,1) model. The state space is two-dimensional, and therefore a finite difference approach is possible. We implement an Alternating Direction Implicit finite difference scheme. This finite difference implementation requires the derivation of some non-trivial boundary conditions. To validate the accuracy of our finite difference scheme, we price a ZCB put option and make a comparison with the Monte Carlo price. We explain how finite difference schemes, despite being computationally expensive, provide two key efficiencies in the context of implementation.

To recover the state variable process from market observables, we implement the Unscented Kalman filter — a non-linear extension of the Kalman filter. We show how the unscented transform — which is the technique underlying the Unscented Kalman filter's incorporation of non-linearity — is applied when a finite difference scheme is used in lieu of an analytical measurement equation. Market observable option data is required to recover the USV state variable; however, we have the freedom to choose whether this data is taken in the form of option prices or implied volatilities. We show that it is necessary to use implied volatilities to recover the USV process. This is because option prices are disproportionally dependent on the short rate, while using implied volatilities removes most of this dependence. To demonstrate this, as well as the filter and the finite difference scheme, the filter is implemented to recover the latent state variable process, from simulated panel data. Finally, we show how a likelihood function can be derived from the Unscented Kalman filter.

The dissertation is structured as follows. In Chapter 2, we review traditional models for the short rate, before introducing the concept of unspanned stochastic volatility and the LADQ $(1,1)$ model. Chapter 3 describes how a finite difference scheme can be implemented to price fixed-income options — an essential element of the estimation process. Chapter 4 provides details on the Unscented Kalman filtering algorithm used to filter out latent state variables from observable market prices. Chapter 5 concludes the dissertation.

Chapter 2

The LADQ (1,1) Short Rate Model

In this section, we give a brief review of the literature and concepts building up to the bivariate USV model of Backwell (2017). Before we proceed, we shall first lay out some assumptions and define some key terms. Throughout, we will work with a standard probability space $(\Omega, \mathcal{F}, \mathbb{P}, \{\mathcal{F}_t\}_{t\in[0,\infty)})$ where $\mathbb{P}$ is the real-world measure and $\{\mathcal{F}_t\}$ is generated by an appropriately multidimensional Brownian motion $\{W_t^{\mathbb{P}}\}$. We assume the absence of arbitrage and therefore the existence of a risk-neutral measure $\mathbb{Q}$. We denote the price of a ZCB at time t with maturity T as

$$P_{tT} = P(t, T, X_t),$$

where the price is assumed to be given by a function of the current and maturity times as well as some *state process* $\{X_t\}$, which can be multidimensional. It is often the case that the state process is the short rate process — $\{r_t\}$. To write a model for the ZCB price directly is generally difficult due to the dependence between variables t and T (Backwell, 2015). It is therefore a common practice to specify a model for the short rate r_t instead.

2.1 Short Rate Models

We will define models for the short rate r_t by their stochastic dynamics, either under the real-world measure ($\mathbb{P}$) or risk-neutral measure ($\mathbb{Q}$). Short rate modelling was first introduced by Vasicek (1977). Vasicek defined the dynamics of the short rate under $\mathbb{P}$ which, via a Girsanov change in measure, gives rise to the following $\mathbb{Q}$ dynamics:

$$dr_t = \kappa(\theta - r_t)dt + \sigma dW_t^{\mathbb{Q}}, \tag{2.1}$$

with initial condition $r_0 \in \mathbb{R}$, constant parameters κ, θ and σ, and where $W_t^{\mathbb{Q}}$ is a one-dimensional Brownian motion under $\mathbb{Q}$. These dynamics provide us with some tractable properties, such as an explicit solution for the short rate, as well as Gaussian distribution (Brigo and Mercurio, 2007). The penalty associated with

this tractability, however, is that rates can become negative. Also, the initial yield curve is endogenous to the model, and cannot be externally specified, which can be considered a disadvantage. The bond price P_{tT} can be expressed as a function

$$P_{tT} = P(T-t, \kappa, \theta, \sigma, r_t),$$

and therefore the initial yield curve is known, given the parameters κ, θ, σ and initial short rate r_0. The reason that the bond price turns out to depend on time to maturity $T-t$ is that neither the drift nor the diffusion coefficients are functions of time. This can be generalised by introducing a time-dependent drift or diffusion, such as in the Hull and White (1990) model.

The Vasicek model is an example of a one-factor model (a model with only one source of randomness). Multi-factor models are often preferred for risk management as they provide more realistic scenarios of yield curve movements (Brigo and Mercurio, 2007). One such model is the two-factor model of Fong and Vasicek (1991) which adds a stochastic volatility factor. The short rate satisfies

$$\begin{aligned} dr_t &= \kappa_r(\theta_r - r_t)dt + \sqrt{v_t}dW_t^{\mathbb{Q},1}, \text{and} \\ dv_t &= \kappa_v(\theta_v - v_t)dt + \sigma_v\sqrt{v_t}dW_t^{\mathbb{Q},2}, \end{aligned}$$

with constant parameters κ_r, θ_r , κ_v and θ_v, and where $\{W_t^{\mathbb{Q},1}\}$ and $\{W_t^{\mathbb{Q},2}\}$ are Brownian motions under $\mathbb{Q}$. Under this specification, ZCB prices are given by

$$P_{tT} = P(T-t, r_t, v_t). \tag{2.2}$$

Importantly, the bond pricing function in Equation (2.2) is a function of both the state variables. This implies that both the state variables are *spanned*. The sensitivities to both state variables can be hedged using ZCBs, since $\frac{\partial P(T-t,r,v)}{\partial r} \neq 0$ and $\frac{\partial P(T-t,r,v)}{\partial v} \neq 0$. Clearly the Fong and Vasicek (1991) model has stochastic volatility; however, this stochastic volatility is spanned. We therefore refer to this model as a *spanned stochastic volatility* model.

2.2 Bivariate Unspanned Stochastic Volatility Models

A USV model includes at least one stochastic volatility state variable which introduces risk which cannot be hedged with primary yield curve instruments (Backwell, 2017). Put simply, in the bivariate case, if the two state variables are $\{r_t\}$ and a USV state variable $\{u_t\}$, then ZCB prices can be given by

$$P_{tT} = P(T-t, r_t).$$

Sensitivities to the stochastic volatility state variable can no longer be hedged using ZCBs as $\frac{\partial P(T-t,r)}{\partial u} = 0$. The stochastic volatility is therefore referred to as *unspanned*.

We focus on these bivariate USV models, which due to the one spanned and one unspanned state variable can also be referred to as $(1,1)$ models. We have discussed the implications of USV but we have yet to specify what type of models will display this phenomenon. Backwell (2017) provides the following theorem which characterises (1,1) models:

Theorem 2.1. *For a bivariate short rate model, the following are equivalent:*

$$\begin{aligned}
&(A) \quad P_{tT} = P(t,T,r_t) \quad \textit{(i.e., the model is a (1,1) model),}\\
&(B) \quad P_{tT} = g(t,T) - f(t,T)r_t,\\
&(C) \quad \mu_t^{\mathbb{Q},r} = r_t^2 + r_t\alpha_1(t) + \alpha_2(t).
\end{aligned}$$

A proof of this theorem is given in Backwell (2017).

It follows from Theorem 2.1 that stochastic volatility short rate models where the short rate has a quadratic drift coefficient under $\mathbb{Q}$ will display USV. In a USV $(1,1)$ model, bond prices are affine in the short rate implying that bond yields are log-affine. This is an interesting result as it differentiates these models from the large class of ATSMs, where bond prices are exponential-affine.

2.3 The LADQ (1,1) Model

While Theorem 2.1 characterises all (1,1) models, we now introduce a specific bivariate USV model. Backwell (2017) specifies a (1,1) model termed the *Log-Affine Double Quadratic* — or LADQ — (1,1) model. Assuming no arbitrage, and therefore the existence of a risk-neutral measure, we specify the following $\mathbb{Q}$ dynamics for the short rate:

$$dr_t = (r_t - \lambda_1)(r_t - \lambda_2)dt + r_t(\bar{r} - r_t)u_t dW^{\mathbb{Q},1}, \text{ and} \tag{2.3}$$

$$du_t = \kappa(\theta - u_t)dt + \sigma\sqrt{u_t}(\rho dW_t^{\mathbb{Q},1} + \sqrt{1-\rho^2}dW_t^{\mathbb{Q},2}), \tag{2.4}$$

where λ_1, λ_2, $\bar{r}$, κ, θ, σ and ρ are constant parameters. Here $\{W^{\mathbb{Q},1}\}$ and $\{W^{\mathbb{Q},2}\}$ are independent $\mathbb{Q}$ Brownian motions. Expanding the risk-neutral drift term in Equation (2.3), which we denote $\mu_t^{\mathbb{Q},r}$, we get

$$\begin{aligned}
\mu_t^{\mathbb{Q},r} &= r_t^2 - r_t(\lambda_1 + \lambda_2) - \lambda_1\lambda_2,\\
&= r_t^2 + r_t\alpha_1 + \alpha_2,
\end{aligned}$$

where $\alpha_1 = \lambda_1 + \lambda_2$ and $\alpha_2 = \lambda_1\lambda_2$. This form for the drift conforms to condition (C) of Theorem 2.1. It follows that bond prices are affine and the model has a (1,1)

structure. The log-affine bond yields and the quadratic drift and volatility structure of the short rate give rise to the name *Log-Affine Double Quadratic* $(1,1)$.

The following conditions are imposed on the short rate:

$$0 < \lambda_1 < \bar{r} < \lambda_2, \tag{2.5}$$

$$0 < r_0 < \bar{r}. \tag{2.6}$$

Backwell (2017) shows that, in the short rate process, λ_1 is the level of mean reversion while λ_2 controls the rate of mean reversion.

The volatility of the short $\sigma^r(r,u) = r(\bar{r}-r)u$ bounds the short rate. If $r_t = \bar{r}$, then the local-volatility is zero (and therefore $\sigma^r(r,u) = 0$), while the drift term is $(\bar{r}-\lambda_1)(\bar{r}-\lambda_2)$, which is guaranteed to be negative by Equation (2.5). There is no volatility but negative drift which ensures that $\bar{r}$ is an upper bound for $\{r_t\}$. Similarly, when $r_t = 0$, the local-volatility is zero. The drift is now given by $\lambda_1\lambda_2$, which is positive, ensuring r_t is bounded from below by zero. These arguments rely on the continuity of $\{r_t\}$ (and the constraint in Equation (2.6)), so that $\{r_t\}$ cannot cross the boundary without first being equal to it.

The non-stochastic component of the diffusion coefficient, $r(\bar{r}-r)$, gives the short rate a *local-volatility* structure which is a deterministic function of the level of the short rate. The USV state variable scales the volatility. For the USV process $\{u_t\}$ the following constraints are imposed:

$$\min(\kappa,\theta,\sigma,u_0) > 0 \quad \text{and} \quad 2\kappa\theta > \sigma^2,$$

The second of these constraints is the Feller condition, which ensures the process $\{u_t\}$ is strictly positive (after imposing the first constraint). In Equation (2.4), the general level of the USV process under $\mathbb{Q}$ is given by θ. The parameters κ and σ dictate the influence of the unspanned parameter: the parameter σ dictates the size of shocks while κ dictates the rate of mean reversion and therefore the persistence of the shocks.

The LADQ (1,1) model falls under the scope of Theorem 2.1 and as such ZCB prices are affine functions:

$$P_{tT} = g(t,T) - f(t,T)r_t. \tag{2.7}$$

Backwell (2017) shows that, in the LADQ (1,1) model, the affine coefficients are given by

$$g(t,T) = \frac{1}{\lambda_2-\lambda_1}(e^{-\lambda_1(T-t)} - e^{-\lambda_2(T-t)}), \text{ and}$$

$$f(t,T) = \frac{1}{\lambda_2-\lambda_1}(\lambda_2 e^{-\lambda_1(T-t)} - \lambda_1 e^{-\lambda_2(T-t)}),$$

for all $0 \leq t \leq T$.

While the risk-neutral specification of the model is important for option pricing, if we want to apply the model longitudinally — to panel data — we need to consider its real-world specification. In order to change from the risk-neutral measure $\mathbb{Q}$ to the real-world measure $\mathbb{P}$, Backwell (2017) defines an adapted, two-dimensional Radon-Nikodym process $\{\zeta_t\}$ with

$$\zeta_t^1 = \frac{r_t(\lambda_1 + \lambda_2 - \delta_1 - \delta_2) - \lambda_1\lambda_2 + \delta_1\delta_2}{r_t(\bar{r} - r_t)u_t}, \text{ and}$$
$$\zeta_t^2 = \frac{\kappa^{\mathbb{P}}\theta^{\mathbb{P}} - \kappa\theta + u_t(\kappa - \kappa^{\mathbb{P}}) - \sigma\sqrt{u_t}\rho\zeta_t^1}{\sigma\sqrt{u_t}\sqrt{1-\rho^2}},$$

for all $t \geq 0$ and additional constant parameters $\delta_1, \delta_2, \kappa^{\mathbb{P}}$ and $\theta^{\mathbb{P}}$. This specification is chosen by Backwell (2017) in order to achieve the following real-world dynamics:

$$dr_t = (r_t - \delta_1)(r_t - \delta_2)dt + r_t(\bar{r} - r_t)u_t dW_t^{\mathbb{P},1}, \text{ and} \tag{2.8}$$
$$du_t = \kappa^{\mathbb{P}}(\theta^{\mathbb{P}} - u_t)dt + \sigma\sqrt{u_t}(\rho dW_t^{\mathbb{P},1} + \sqrt{1-\rho^2}dW_t^{\mathbb{P},2}), \tag{2.9}$$

where $\{W^{\mathbb{P},1}\}$ and $\{W^{\mathbb{P},2}\}$ are independent $\mathbb{P}$ Brownian motions. This specification allows for parametric control similar to that of the risk-neutral dynamics. As with the risk-neutral drift parameters, following constraint is imposed:

$$0 < \delta_1 < \bar{r} < \delta_2. \tag{2.10}$$

The general level of the short rate is given by δ_1 while the rate of reversion towards this level is given by δ_2.

It is useful now to view our implementation problem as a *state space system* (Piazzesi, 2010). This system consists of a state vector which is linked to observations via a *measurement* equation and a *state transition* equation which describes the dynamics of the states. This can be formalised mathematically. Given a vector $\boldsymbol{y}_k$, consisting of N observations at each discrete time-step $k = 1, \ldots, M$, we wish to estimate the bivariate time series of the latent state variable vector $\boldsymbol{x}_k$. The vectors,

$$\boldsymbol{y}_k = \begin{bmatrix} y_k^1 \\ \vdots \\ y_k^N \end{bmatrix} \text{ and } \boldsymbol{x}_k = \begin{bmatrix} r_k \\ u_k \end{bmatrix},$$

are linked via a set of measurement equations which relate the state variables to the observations. The observation vector $\boldsymbol{y}_k$ contains market observables, such as yields, prices or implied volatilities, so the measurement equations give the model implied observables from the state variables, for example the bond pricing function in Equation (2.2).

When fitting term structure models to market data, the problem of *stochastic singularity* must be addressed. In the context of ATSMs, stochastic singularity arises when we attempt to model N market yields with $M < N$ state variables. This is generally the case in ATSMs, where we attempt to fit many points along the yield curve using few state variables. If we attempt to fit N state variables to N yields, we can simply invert the yield curve to solve for the latent states. The addition of one extra yield, however, will create an inconsistency. While bivariate USV models do not fall under the class of ATSMs, the stochastic singularity problem is certainly present as we attempt to fit N market observables with the two-dimensional state variable vector $\boldsymbol{x}_k$.

The problem of stochastic singularity can be solved by discarding observations until $N \leq M$, but this approach is arbitrary. A popular alternative approach is Kalman filtering. Kalman filtering solves the problem of stochastic singularity (Piazzesi, 2010) by assuming each observation is made with some error, which seems a more sound approach than discarding observations. Intuitively, it is unrealistic for a model to be able to perfectly price every asset in the market. We seek a model which fits the markets as closely as possible but there will always be some residual error. The Kalman filter, based on the assumption of noisily measured observations $\{\boldsymbol{y}_k\}$, estimates the latent state process $\{\boldsymbol{x}_k\}$.

The Kalman filter dynamic system is characterised by two equations: the state transition equation relating $\boldsymbol{x}_{k-1}$ to $\boldsymbol{x}_k$ and the measurement equation relating the state $\boldsymbol{x}_k$ to the observation $\boldsymbol{y}_k$. The Kalman filter requires that both these equations are linear. This is not the case in the LADQ (1,1) model. The measurement equation is non-linear due to the inclusion of option prices in the observation set. The state transition equations, discretisations of the state variable stochastic differential equations (SDEs), are non-linear due to the quadratic terms in Equation (2.8) and square root term in Equation (2.9). There are a number of different filtering algorithms which can handle non-linear systems. Particle filters (or Sequential Monte Carlo methods), first introduced in Del Moral (1996), are an alternative filtering method to Kalman filtering. Particle filtering has been used for parameter estimation by, among others, Gellert and Schlögl (2018). For the purpose of this paper, however, particle filtering is likely to be too computationally expensive due to the complexity of the measurement equation. The Extended Kalman filter (EKF) and the Unscented Kalman filter (UKF) are both non-linear extensions of the Kalman filter. Wan and Van Der Merwe (2000) show that the UKF improves on the accuracy of the EKF without added computational complexity. Carr, Gabaix and Wu (2011) consider USV in the context of the *linearity generating* framework. They successfully implement the UKF for parameter estimation. In Chapter 4, we discuss

the derivation of the UKF and implement it to recover the latent state process $\{\boldsymbol{x}_k\}$ from simulated panel data.

Chapter 3

Option Pricing

The ability to price interest rate options is vital for state estimation due to the unspanned nature of the stochastic volatility. Since bond prices are independent of u_t, only option prices — or their implied volatilities — can provide information on this state variable. We do not have analytical solutions for option prices, nor can we use characteristic function pricing techniques, as the non-standard dynamics in Equation (2.3) preclude such methods (Backwell, 2017). We are left to consider two of the most common numerical pricing techniques: Monte Carlo and finite difference methods. Monte Carlo pricing is effective and simple to implement; however, in the iterative environment of estimation, it can be computationally taxing due to the large sample size such methods require. This dissertation will consider a finite difference approach due to two efficiencies it provides, which we highlight during the course of this chapter.

The use of finite difference schemes requires a partial differential equation (PDE), governing the dynamics of option values, equipped with initial and boundary conditions. We consider the problem of pricing a European option with terminal payoff based on ZCB prices. ZCB prices are functions of the short rate, and therefore the payoff at maturity is given by $\Phi(r)$. We denote the value of the option at time t by $F(t, r_t, u_t)$, a function of time and the state variables. A simple application of Itô's Lemma gives

$$\begin{aligned} dF(t,r_t,u_t) = \big(&F_t + (r_t-\lambda_1)(r_t-\lambda_2)F_r + \kappa(\theta-u_t)F_u \\ &+ \frac{1}{2}(r_t(\bar{r}-r_t)u_t)^2F_{rr} + \frac{1}{2}\sigma^2 u_t F_{uu} \\ &+ \rho(r_t(\bar{r}-r_t)u_t)(\sigma\sqrt{u_t})F_{ru}\big)dt \\ &+ r_t(\bar{r}-r_t)u_t d\hat{W}_t^{\mathbb{Q},1} + \sigma\sqrt{u_t}d\hat{W}_t^{\mathbb{Q},2}, \end{aligned} \tag{3.1}$$

where $\{\hat{W}_t^{\mathbb{Q},1}\}$ and $\{\hat{W}_t^{\mathbb{Q},2}\}$ are correlated $\mathbb{Q}$ Brownian motions with correlation ρ, and we use subscripts to denote the partial derivatives (for example, $F_r = \frac{\partial F}{\partial r}$).

From Equation (3.1), the risk-neutral drift of the option value is given by

$$F_t + \mu_t^{\mathbb{Q},r} F_r + \mu_t^{\mathbb{Q},u} F_u + \frac{1}{2}(\sigma_t^r)^2 F_{rr} + \frac{1}{2}(\sigma_t^u)^2 F_{uu} + \rho \sigma_t^r \sigma_t^u F_{ru}, \tag{3.2}$$

where

$$\mu_t^{\mathbb{Q},r} = (r_t - \lambda_1)(r_t - \lambda_2), \qquad \sigma_t^r = r_t(\bar{r} - r_t)u_t,$$
$$\mu_t^{\mathbb{Q},u} = \kappa(\theta - u_t), \qquad \sigma_t^u = \sigma\sqrt{u_t}.$$

Note that for the sake of presentation, the coefficients' dependencies on the state variables have been suppressed. The expected rate of return of the option, given by Equation (3.2), must be equivalent to the short rate under the risk-neutral measure. Equating the two, and setting $\tau = T - t$, reversing time and transforming the terminal condition into an initial condition, gives the following PDE:

$$F_\tau - \mu_t^{\mathbb{Q},r} F_r - \mu_t^{\mathbb{Q},u} F_u - \frac{1}{2}(\sigma_t^r)^2 F_{rr} - \frac{1}{2}(\sigma_t^u)^2 F_{uu} - \rho \sigma_t^r \sigma_t^u F_{ru} + rF = 0. \tag{3.3}$$

When considering initial boundary problems with two spatial domains, explicit methods generally have stability problems, while implicit methods are difficult to implement (Duffy, 2013). The two-step Alternating Direction Implicit (ADI) scheme provides a better alternative by introducing an intermediary half time-step to aid stability and simplicity. The ADI scheme was first outlined by Peaceman and Rachford (1955) and has since been reviewed by authors such as Duffy (2013) and Crépey (2013). This technique improves on the stability of the explicit scheme while remaining relatively simple to implement. In the first half step of the scheme, we solve the PDE implicitly in the first spatial direction and explicitly in the second spatial direction and vice versa for the next half step. A benefit of the ADI scheme, is that each step of the algorithm can be simplified to solving a tridiagonal matrix equation. This can be computed efficiently with methods such as the Thomas algorithm. Duffy (2013) shows that the ADI scheme, in two spatial dimensions, is unconditionally stable if the PDE does not include a mixed derivative. This condition is satisfied for Equation (3.3) if the two Brownian motions are uncorrelated ($\rho = 0$). When there is correlation, the coefficient of the mixed derivative is non-zero and the scheme is not unconditionally stable.

3.1 Implementing the ADI Scheme

Let us consider the problem of pricing a European put on a ZCB. This is a claim contingent on the price of the ZCB which, in the LADQ (1,1) model, is an affine function of the short rate. We consider ZCB puts, because they can be used as the

building blocks for pricing other interest rate options. For example, the payoff of caplet (and therefore a cap) can be represented as a combination of ZCB puts.

We equip the PDE in Equation (3.3) with an initial condition $\Phi(\cdot)$ equivalent to the payoff of the option at maturity. The initial condition is therefore a function of the short rate, given by

$$\begin{aligned}\Phi(r) &= (K - P(T,S))^+ \\ &= (K - g(T,S) + f(T,S)r)^+.\end{aligned}$$

Note that the PDE is time reversed and therefore the terminal payoff of the option becomes the initial condition for the PDE. To implement the finite difference scheme, it is necessary to truncate the domain of u_t with an upper bound of $\bar{u}$ and specify boundary conditions. This reduces the problem to a finite state space $[0, \bar{r}] \times [0, \bar{u}]$. Calculating the boundary conditions, which describe the behaviour along the edges of this domain, requires some careful reasoning.

In the r direction, we can use Neumann boundary conditions[1]. As $r \to 0$ and $r \to \infty$, the slope of the price surface tends to the slope of the terminal payoff surface (in the r direction). When $r = 0$, the option will be far out-of-the-money and $\frac{\partial \Phi(r_T)}{\partial r} = 0$. When $r = \bar{r}$, the option is deep in-the-money and we can effectively ignore the 'positive part' function. Expanding the payoff function and taking the partial derivative with respect to r we get:

$$\begin{aligned}\frac{\partial \Phi(r)}{\partial r} &= \frac{\partial}{\partial r}(K - g(T,S) + f(T,S)r), \\ &= f(T,S).\end{aligned}$$

It follows that Neumann boundary conditions can be given by

$$\frac{\partial}{\partial r}F(t,0,u) = 0, \text{ and} \qquad \frac{\partial}{\partial r}F(t,\bar{r},u) = f(T,S).$$

The u boundaries are slightly more complicated to deal with. Firstly, when $u = 0$ we notice that Equation (3.3) reduces to

$$F_\tau + (r_t - \lambda_1)(r_t - \lambda_2)F_r + \kappa\theta F_u - rF = 0.$$

This is implemented along the boundary by using a forward, rather than central, difference approximation for F_u. The value along the $u = \bar{u}$ boundary is numerically approximated via extrapolation. This is done by adding the difference between the two interior points to the point preceding the boundary to provide an

[1] Neumann boundary conditions specify a value for the normal derivative — the derivative with respect to the outward normal along the boundary — instead of specifying a value for the function itself (Duffy, 2013).

approximation of the value along the $\bar{u}$ boundary. The details of the scheme are given in full in Appendix A.

Let us consider a ZCB put option, with maturity of one year, written on the two-year bond. The payoff function is given by $\Phi(r) = (K - P(1,2))^+$. Figure 3.1 plots the price surface of an at-the-money put in the centre panel. The at-the-money strike is calculated assuming an initial interest rate of 3%. In the left panel, we fix $r_0 = 0.03$ and $u_0 = 1.5$ and price the put for a range of strikes. The ADI price is compared to the Monte Carlo price and 95% confidence bounds[2].

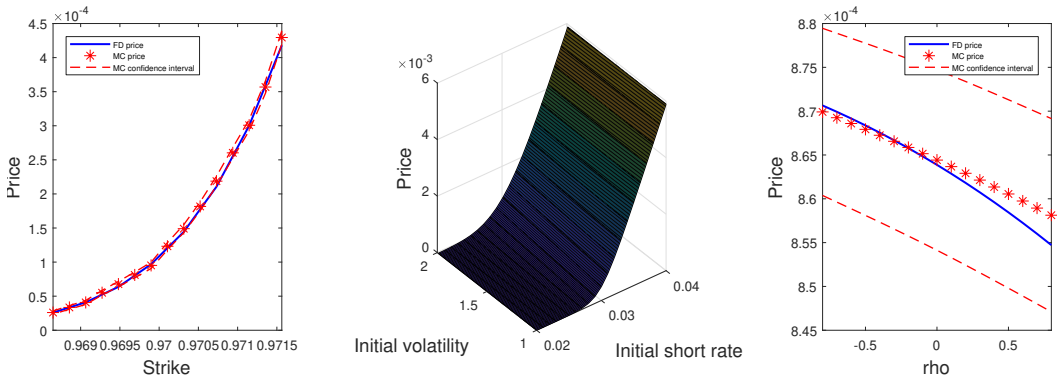

Fig. 3.1: The left and right panels plot finite difference (FD) prices for a range of strikes and correlations respectively against the corresponding Monte Carlo (MC) prices and 95% confidence bounds. The centre panel illustrates the FD pricing surface for an at-the-money put option written on the two-year ZCB with maturity of one year.

As discussed above, it is important to assess the stability of the scheme. We do this by stress testing the scheme with respect to the parameters affecting the mixed derivative (as we identified it as the possible source of instability). In the right panel of Figure 3.1, to assess the stability of the scheme with respect to ρ, we compare the ADI price to the Monte Carlo price with 95% confidence bounds[3]. We see that the ADI scheme remains stable for moderate correlation values in the range $-0.8 < \rho < 0.8$.

Using a finite difference scheme provides two computational efficiencies. Firstly, when pricing caps, pricing the longest dated caplet is sufficient as all shorter dated caplets will be recovered as intermediary solutions. Since bonds are a function of

[2] The Monte Carlo price was calculated using a sample size of 50 000 and a Euler-Maruyama discretisation of the state variable dynamics with 120 time steps.

[3] Again, we use a sample size of 50 000 and a Euler-Maruyama discretisation with 120 time steps; however, each Monte Carlo estimate is calculated using the same set of random numbers to isolate the effect of the changing parameter ρ.

$T - t$, all caplets have the same terminal condition. Therefore, the only variable when pricing caplets using finite difference methods (assuming they are all struck at the same rate) is the time to maturity of the option. Secondly, and most importantly, finite difference schemes provide a pricing surface for all values of the state variables while other methods like Monte Carlo only provide point-wise solutions. Using a pricing surface in the UKF is more efficient, as it only needs to be calculated once at the start of the filter rather than pricing at each step of the filter. This efficiency is discussed in detail in Section 4.4.

Chapter 4

The Unscented Kalman Filter

4.1 Background

The Kalman filter is an algorithm for extracting a time series of latent (or hidden) states driving a noisy observed process in a linear system. Observations $\{\boldsymbol{y}_k\}$ are driven by the latent state process $\{\boldsymbol{x}_k\}$, which is assumed to be Markov. This dependence structure is illustrated in Figure 4.1. While the Kalman filter was original developed for applications in engineering, it overcomes the problem of stochastic singularity in term structure state estimation, making it a popular tool in the financial literature. The Kalman filter has the same form as a discrete time Hidden Markov Model (HMM), discussed extensively by Zucchini and MacDonald (2009), except that it treats a continuous state space. As with the HMM, observations are conditionally independent. Formally

$$P(\boldsymbol{y}_k|\boldsymbol{y}_1,\ldots,\boldsymbol{y}_{k-1},\boldsymbol{x}_1,\ldots,\boldsymbol{x}_k) = P(\boldsymbol{y}_k|\boldsymbol{x}_k). \tag{4.1}$$

Equation (4.1) follows from the Markov nature of $\{\boldsymbol{x}_k\}$. This conditional independence allows for the filter dynamics to be broken up into a longitudinal and cross-sectional dynamics.

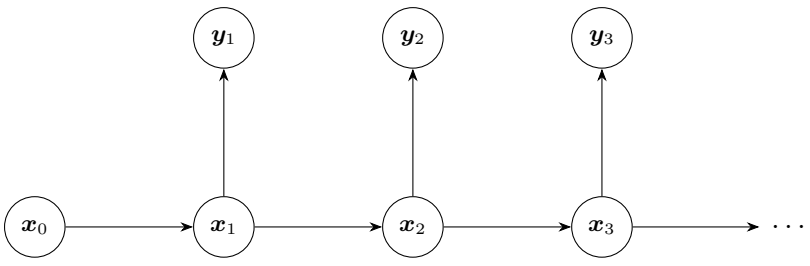

Fig. 4.1: An illustration of the dependence structure between the latent state process $\{\boldsymbol{x}_k\}$ and the observed process $\{\boldsymbol{y}_k\}$.

The system is characterised by the *state transition* and the *measurement* equations:

$$\begin{aligned} \boldsymbol{x}_k &= g(\boldsymbol{x}_{k-1}) + \epsilon_k && \text{(the state transition equation)} \\ \boldsymbol{y}_k &= h(\boldsymbol{x}_k) + \eta_k && \text{(the measurement equation)} \end{aligned}$$

The innovations, ϵ_k and η_k, in each equation are assumed to be independent and normally distributed with $\epsilon_k \sim N(0, Q_k)$ and $\eta_k \sim N(0, R_k)$. The Kalman filter requires that g and f are linear functions. This is to ensure Gaussian random variables propagated through the state transition and measurement equations remain Gaussian. In many systems, the linearity constraint on f and g is untenable. In the next section, we introduce the unscented transform which can be incorporated to extend the Kalman filter to non-linear systems.

4.2 The Unscented Transform

The unscented transform (UT) is a method for approximating the distribution of a multi-dimensional random variable $\boldsymbol{y} = g(\boldsymbol{x})$: a non-linear transform of a normally distributed random variable $\boldsymbol{x}$. This is accomplished through the formation of $2n+1$ sigma points, where n is the dimension of $\boldsymbol{x}$. Assume $\boldsymbol{x} \sim N(\boldsymbol{\mu}, P)$ and define $\boldsymbol{y} = g(\boldsymbol{x}) + \epsilon$ where $\epsilon \sim N(\boldsymbol{0}, Q)$ and $g(\cdot)$ is some non-linear function. Define the scaling parameter $\lambda = \alpha^2(n + \kappa) - n$. The tuning parameters α and κ are used to determine the spread of the sigma points. We form the matrix $\mathcal{X}$ of sigma points as follows:

$$\begin{aligned} \mathcal{X}_0 &= \boldsymbol{\mu}, \\ \mathcal{X}_i &= \boldsymbol{\mu} + \sqrt{n+\lambda}(\sqrt{P})_i \quad i = 1, \ldots, n, \\ \mathcal{X}_i &= \boldsymbol{\mu} - \sqrt{n+\lambda}(\sqrt{P})_{i-n} \quad i = n+1, \ldots, 2n, \end{aligned}$$

where $(\sqrt{P})_i$ is the i^{th} column of the matrix square root of P. The matrix square root is calculated using the Cholesky decomposition $P = LL^T$ such that $\sqrt{P} = L$. The corresponding weights vectors $\boldsymbol{w}^{(m)}$ and $\boldsymbol{w}^{(c)}$ are given by:

$$\begin{aligned} \boldsymbol{w}_0^{(m)} &= \lambda/(n+\lambda), \\ \boldsymbol{w}_0^{(c)} &= \lambda/(n+\lambda) + (1 - \alpha^2 + \beta), \\ \boldsymbol{w}_i^{(m)} &= W_i^{(c)} = 1/[2(n+\lambda)] \quad i = 1, \ldots, 2n, \end{aligned}$$

where β is used to incorporate prior knowledge of the distribution of $\boldsymbol{x}$. The parameter α is normally set to a small positive number, κ is normally set to 0 and for Gaussian prior distributions $\beta = 2$ is optimal (Wan and Van Der Merwe, 2000). The

posterior matrix of sigma points $\mathcal{Y}$ is formed by applying the non-linear function $g(\cdot)$ to the sigma points:

$$\mathcal{Y}_i = g(\mathcal{X}_i) \quad \text{for } i = 0, \ldots, 2n.$$

The mean and covariance of $\boldsymbol{y}$ can then be approximated using:

$$\boldsymbol{\mu_y} \approx \sum_{i=0}^{2n} \boldsymbol{w}_i^{(m)} \mathcal{Y}_i, \text{ and}$$
$$P_{\boldsymbol{y}} \approx \sum_{i=0}^{2n} \boldsymbol{w}_i^{(c)} (\mathcal{Y}_i - \boldsymbol{\mu_y})(\mathcal{Y}_i - \boldsymbol{\mu_y})^\top.$$

such that $\boldsymbol{y} \dot{\sim} N(\boldsymbol{\mu_y}, P_{\boldsymbol{y}})$. Here the symbol $\dot{\sim}$ denotes approximately distributed.

This method allows us to approximate the posterior distribution — after propagation through a non-linear function of a Gaussian random variable — as Gaussian using a small number of sigma points. Linearity is required in the Kalman filter in order to calculate the mean and covariance after propagation through the measurement and state transition equations. The UT allows us to calculate these statistics after a non-linear transform. This is done in a computationally efficient manner, as only $2n + 1$ sigma points need to be computed.

4.3 Filter Details

The Unscented Kalman filter algorithm consists of two steps: the *prediction* and the *update* step corresponding to the transition and measurement equations respectively. In the linear filter, at each step a mean vector and covariance matrix are computed. The UKF applies the unscented transform to approximate these statistics.

We assume that the conditional distribution of the previous latent state is Gaussian:

$$\boldsymbol{x}_{k-1} | \boldsymbol{y}_{1:k-1} \sim N(\boldsymbol{m}_{k-1}, P_{k-1}).$$

We refer to this as the *filtering distribution* since it is the distribution of the latent or filtered states conditional on all previous observations.

The joint distribution of $\boldsymbol{x}_{k-1}$ and $\boldsymbol{x}_k = g(\boldsymbol{x}_{k-1}) + \boldsymbol{\epsilon}_{k-1}$, the propagation of $\boldsymbol{x}_{k-1}$ through the non-linear dynamic model, can be approximated as Gaussian with the UT. The matrix of sigma points $\mathcal{X}$ of $\boldsymbol{x}_{k-1} \sim N(\boldsymbol{m}_{k-1}, P_{k-1})$ is formed and propagated through the state transition model to give the matrix of transformed sigma vectors $\hat{\mathcal{X}}$. The joint distribution is then given by

$$\begin{bmatrix} \boldsymbol{x}_{k-1} \\ \boldsymbol{x}_k \end{bmatrix} \dot{\sim} N\left(\begin{bmatrix} \boldsymbol{m}_{k-1} \\ \hat{\boldsymbol{m}}_k \end{bmatrix}, \begin{bmatrix} P_{k-1} & P_{12} \\ (P_{12})^\top & \hat{P}_k \end{bmatrix} \right),$$

where

$$\hat{\boldsymbol{m}}_k = \sum_i \boldsymbol{w}_{i-1}^{(m)} \hat{\mathcal{X}}_i,$$
$$P_{12} = \sum_i \boldsymbol{w}_{i-1}^{(c)} (\mathcal{X}_i - \boldsymbol{m}_{k-1})(\hat{\mathcal{X}}_i - \hat{\boldsymbol{m}}_k),$$
$$\hat{P}_k = \sum_i \boldsymbol{w}_{i-1}^{(c)} (\hat{\mathcal{X}}_i - \hat{\boldsymbol{m}}_k)(\hat{\mathcal{X}}_i - \hat{\boldsymbol{m}}_k) + Q_{k-1}.$$

Therefore, at the prediction step $\boldsymbol{x}_k \sim N(\hat{\boldsymbol{m}}_k, \hat{P}_k)$.

Next, we update these predictions using the observation at time k. We consider the joint distribution of $\boldsymbol{x}_k$ and $\boldsymbol{y}_k = h(\boldsymbol{x}_k) + \boldsymbol{\eta}_k$. Again, using the UT, the joint distribution can be approximated as Gaussian. The matrix of sigma points $\mathcal{X}$ of $\boldsymbol{x}_k \sim N(\hat{\boldsymbol{m}}_k, \hat{P}_k)$ is formed and propagated through the measurement model to give the matrix of transformed sigma points $\hat{\mathcal{Y}}$. The joint distribution is given by

$$\begin{bmatrix} \boldsymbol{x}_k \\ \boldsymbol{y}_k \end{bmatrix} \dot{\sim} N\left(\begin{bmatrix} \hat{\boldsymbol{m}}_k \\ \tilde{\boldsymbol{y}}_k \end{bmatrix}, \begin{bmatrix} \hat{P}_k & C_k \\ (C_k)^\top & S_k \end{bmatrix} \right), \tag{4.2}$$

where

$$\tilde{\boldsymbol{y}}_k = \sum_i \boldsymbol{w}_{i-1}^{(m)} \mathcal{Y}_i, \tag{4.3}$$
$$C_k = \sum_i \boldsymbol{w}_{i-1}^{(c)} (\mathcal{X}_i - \hat{\boldsymbol{m}}_k)(\mathcal{Y}_i - \tilde{\boldsymbol{y}}_k), \tag{4.4}$$
$$S_k = \sum_i \boldsymbol{w}_{i-1}^{(c)} (\hat{\mathcal{Y}}_i - \tilde{\boldsymbol{y}}_k)(\hat{\mathcal{Y}}_i - \tilde{\boldsymbol{y}}_k) + R_k. \tag{4.5}$$

To compute the distribution of $\boldsymbol{x}_k$ conditional on the observation $\boldsymbol{y}_k$, we need the following well-known lemma.

Lemma 4.1. *If*

$$\begin{bmatrix} x \\ y \end{bmatrix} \sim N\left(\begin{bmatrix} a \\ b \end{bmatrix}, \begin{bmatrix} A & C \\ C^\top & B \end{bmatrix} \right),$$

then

$$x|y \sim N(a + CB^{-1}(y - b), A - CB^{-1}C^\top).$$

A proof is given by Anderson (1958).

Applying Lemma 4.1 to Equation (4.2), the conditional mean and covariance of $\boldsymbol{x}_k$ are given by:

$$\boldsymbol{m}_k = \hat{\boldsymbol{m}}_k + K_k(\boldsymbol{y}_k - \tilde{\boldsymbol{y}}_k),$$
$$P_k = \hat{P}_k - K_k S_k^{-1} K_k^\top,$$

where $K_k = C_k S_k^{-1}$ is termed the Kalman gain. Intuitively, $(\boldsymbol{y}_k - \tilde{\boldsymbol{y}}_k)$ is the error between the realised observation and the model's prediction. The Kalman gain term scales the residual to update the predicted mean of the state variable vector.

4.4 Application to the LADQ (1,1)

The UKF can be applied longitudinally to data simulated from the LADQ (1,1) model, assuming known model parameters, to recover the latent state variables. The state variables under the LADQ (1,1) model specification are Markov (importantly under $\mathbb{P}$), so the Markov assumption of the UKF is fulfilled. The state transition equation is derived using a Euler-Maruyama discretisation of the SDEs governing the real-world dynamics of the state variables. The drift terms are incorporated in the transition function $g(\cdot)$, while the diffusion terms are included in the covariance of the innovations Q_k.

The measurement equations are given by the appropriate asset pricing functions, noting that assets are priced under the risk-neutral measure. ZCBs are priced using the closed form bond price given in Equation (2.7). Caps are priced using the ADI finite difference scheme discussed in Chapter 3. This is where we see the benefit of using a finite difference scheme over Monte Carlo methods to price options. Suppose we have an observation vector of size n, containing $n_{\text{opt}} < n$ options, sampled at M time points. At each iteration of the filter, we need to price the n_{opt} options at the $2n + 1$ sigma points. Over the M iterations of the filter, the Monte Carlo simulation would have to be implemented $M(2n + 1)n_{\text{opt}}$ times. The finite difference scheme is more efficient since it calculates a surface of prices for all r and u. Therefore, it need only be implemented once at the start of the filter. The measurement equation simply references points on the surface (interpolating when necessary) rather than re-pricing at each iteration.

To demonstrate the UKF, we consider a simulated data set consisting of a one-year ZCB and a two-year cap written on the three-month rate. We observe these prices daily over a 301-day window. An Euler-Maruyama discretisation of the LADQ (1,1) model dynamics is used to simulate the state variables and the corresponding market observables — which are augmented with additional Gaussian noise to induce a measurement error[1]. Due to the simulated nature of the data, the true latent states and all model parameters are known. The initial values for the state variable mean $\boldsymbol{m}_0$ and covariance P_0, as well as the covariance of the measurement residuals R must by specified *a priori*. We fix R as a constant and set it to the covariance matrix of the additive Gaussian noise used during the simulation, leaving only the initial mean $\boldsymbol{m}_0$, covariance matrix P and the filter tuning parameters as discretionary inputs.

[1] We note that using an Euler-Maruyama discretisation to derive the state transition equation and simulate the latent states removes discretisation error that would arise when considering real-world data. Therefore the results on simulated data likely overstate the power of the filter.

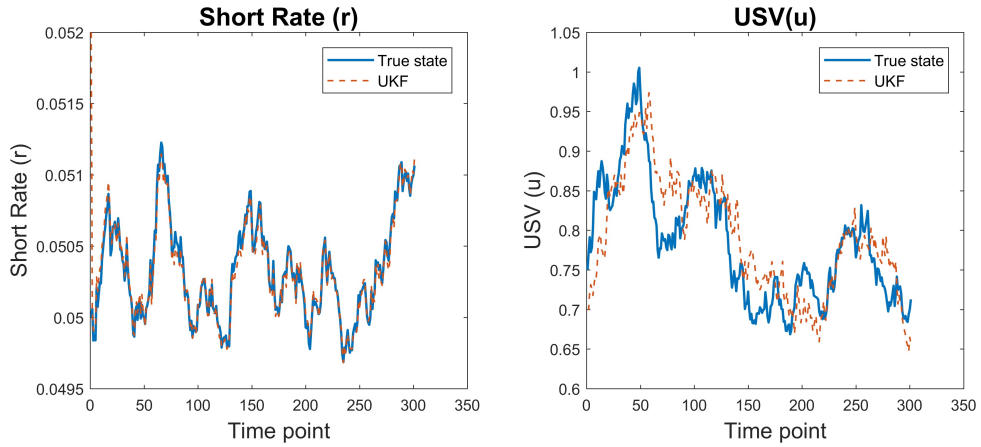

Fig. 4.2: Latent state variable recovery from simulated ZCB and cap prices with additive Gaussian noise.

Figure 4.2 illustrates the recovery of the latent state variables by the UKF. In the left panel, the short rate is initialised at a level of 5.2%. After a short *burn-in* period, the UKF tracks the true short rate series almost perfectly. In the right panel, u is initialised at 0.7. The UKF is able to partially track the true USV process but it is far from the accuracy of the left panel. This is due to the disproportionate effect the state variables have on the price of the cap. The cap price is sensitive to the volatility, but is also highly sensitive to the short rate. This prevents the filter from accurately estimating the USV state process.

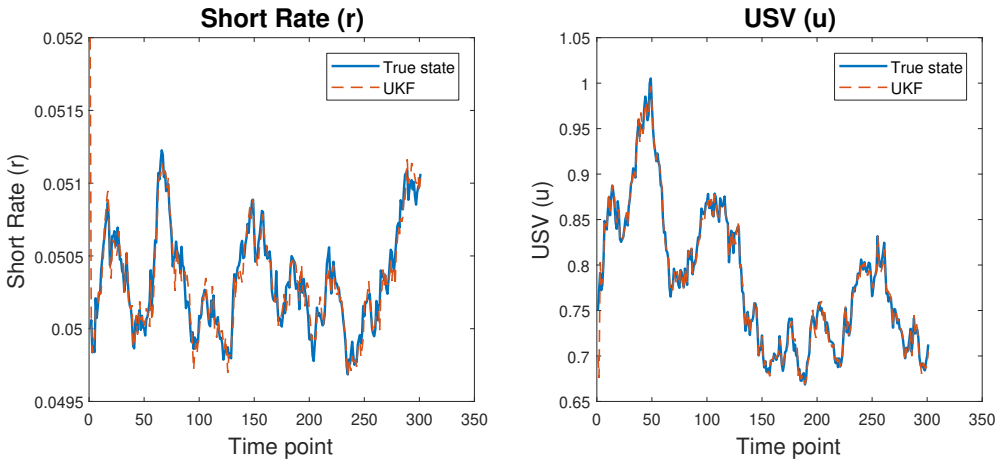

Fig. 4.3: Latent state variable recovery from simulated ZCB and cap implied volatilities with additive Gaussian noise.

To improve the estimation of the USV state variable, we consider cap implied volatilities rather than prices. While cap price sensitivity is disproportionately

skewed towards the short rate, their implied volatilities are more sensitive to changes in the USV state variable. This is because taking implied volatilities removes the moneyness effect. In the Black-Scholes world, this is done perfectly. Under other models, which do not assume constant volatility, we expect to see a volatility skew or smile. The implied volatility will still have some sensitivity to the short rate; however, it is far less sensitive than price. The implied volatilities allow the UKF to get a better read on the USV process. We consider the same simulated data set as before, except with cap prices converted to implied volatilities. Figure 4.3 illustrates the tracking results. In the right panel, we see a marked improvement in the state recovery from Figure 4.2. The UKF gives an almost perfect estimate of the true latent state process.

The $\boldsymbol{m}_0$ used for Figures 4.2 and 4.3 was selected for illustrative purposes but the filter still converges when the initial guess is far from the true value (although the burn-in time is slightly longer). The initial state covariance P_0 is parametrised as a diagonal matrix. There is no significant sensitivity of the UKF results to P_0, as long as extremely small numbers are avoided.

One of the major issues encountered during implementation was performing the Cholesky decomposition for the unscented transform. A Cholesky decomposition must be performed on the covariance P_k and predicted covariance $\hat{P}_k$ matrices in order to generate the sigma points. Setting the tuning parameter $\kappa \geq 0$ guarantees a semi positive-definite covariance matrix; however, the Cholesky decomposition requires the input matrix to be positive-definite (Kandepu, Foss and Imsland, 2008). We also found that while theoretically the covariance matrix should be semi positive-definite, in practice this was often not the case due to numerical error. The entries in the state covariance matrices, at later stages of the filter, become very small allowing for computer floating point error.

The issue was addressed in two ways. Firstly, to prevent non positive-definite covariance matrices wherever possible, we used the tuning parameter α to adjust the spread of the sigma points. Interestingly, we found that larger α values produced better results. Adjusting α reduced the number of non positive-definite matrix occurrences but failed to eliminate them entirely. For the remaining cases, we implemented an altered version of the *nearestSPD* algorithm of D'Errico (2013). The algorithm finds the nearest symmetric positive-definite matrix to the input. In this setting, the nearest matrix is defined as the matrix which minimises the Frobenius norm of the two. This numerical 'work-around' should have an insignificant impact on the estimation results as it only affects the covariance of the state variables.

In this chapter, we have assumed known parameters; however, in practice this is unlikely to be the case. For example, if we would like to conduct an empirical study

of the model using market panel data, the parameters would be unknown and would need to be estimated. This can be done via maximum likelihood estimation.

4.5 Maximum Likelihood Estimation

In practice, the model parameters are unknown. They can be estimated from market panel data using maximum likelihood estimation. To perform maximum likelihood estimation, we must first derive a likelihood function. The likelihood function is a measure of the plausibility of the parameter set given the observed data. The likelihood function is given by

$$L(\boldsymbol{y}|\boldsymbol{\pi}) = \prod_{k=1}^{M} f(\boldsymbol{y}_k|\boldsymbol{y}_1, \ldots, \boldsymbol{y}_{k-1}),$$

where f is the conditional probability density function and $\boldsymbol{\pi}$ denotes the set of model parameters. The UKF can also be expressed as a generative model which generates observations given the latent state estimate. The observations in the filter are conditionally independent, as shown in Equation (4.1), hence all information from previous observations are contained in the current state estimate. The distribution of interest for calculation of the likelihood function is the marginal distribution of $\boldsymbol{y}_k$. The joint distribution of the current state estimate and the current observation is given in Equation (4.2). From this, the marginal distribution of $\boldsymbol{y}_k$ can be shown to be

$$\boldsymbol{y}_k \dot{\sim} N(\tilde{\boldsymbol{y}}_k, S_k),$$

where $\tilde{\boldsymbol{y}}_k$ and S_k are calculated in Equations (4.3) and (4.5) respectively. We marginalise out $\boldsymbol{x}_k$ so that the likelihood function is a direct measure of the parameter likelihood given $\{\boldsymbol{y}_k\}$. Since the observations are Gaussian, the log-likelihood can be written directly as

$$\ell(\boldsymbol{y}|\boldsymbol{\pi}) = -\frac{1}{2}(MN\log(2\pi) + \sum_{k=1}^{M}\log|S_k| + \sum_{k=1}^{M}(\boldsymbol{y}_k - \tilde{\boldsymbol{y}}_k)^T S_k^{-1}(\boldsymbol{y}_k - \tilde{\boldsymbol{y}}_k)), \tag{4.6}$$

where N is the dimension of the observation vector $\boldsymbol{y}_k$ and $|\cdot|$ is the matrix determinant function. Given an initial guess for the parameter set $\boldsymbol{\pi}$, a numerical optimiser can be used to solve for the $\boldsymbol{\pi}$ which maximises Equation (4.6).

Here we can see the benefits of using the UKF. Firstly, we have a simple closed-form solution for the likelihood. Secondly, it is a computational efficient method of non-linear filtering. This is important when maximising the likelihood, as the filter will have to be run multiple times at each iteration of the numerical optimiser.

Finally, with regards to parameter estimation, careful attention must be paid to the data included in the observation set $\{\boldsymbol{y}_k\}$. We have already shown that it is preferable to use bond prices and option implied volatilities. For state estimation, one bond price and one cap implied volatility are sufficient; however, for parameter estimation, it will be necessary to include more assets in the observation set. For example, the price of a ZCB given, the parameters λ_1 and λ_2, is an affine function of the short rate only. The short rate can therefore be inferred from one bond price. If the parameters are unknown, bond prices are now a function of three variables — λ_1, λ_2 and the short rate. Therefore, we must include at least three bonds to estimate the parameters. Similarly, it is also necessary to include more implied volatilities to recover the parameters associated with the volatility. Increasing the size of the observation improves the cross-sectional fit of the model; however, it does not improve the longitudinal fit. As a result, the real-world parameters remain difficult to estimate accurately, as illustrated by Duffee and Stanton (2012), regardless of the number of measurements.

Chapter 5

Conclusion

In spanned models, ZCB data (yields or prices) are sufficient to fit the model and filter out the latent state variables. In ATSMs, bond yields are affine functions of the short rate, as a result the linear Kalman filter is a popular technique for recovering the state variables. Implementing term structure models with unspanned factors poses some additional challenges.

For bivariate USV models — such as the LADQ (1,1) model — it is necessary to include option data to filter out the USV state variable. The inclusion of option data and the non-standard dynamics of the LADQ (1,1) model introduces non-linearity into the dynamic system. This precludes the use of the standard linear Kalman filter; however, the Unscented Kalman filter is an efficient and effective extension that handles non-linearity.

The USV state variable makes it necessary during filtering to price options. There are no closed-form solutions for option prices in the LADQ (1,1) model. A finite difference approach, while not trivial to implement, provides two important numerical efficiencies during state estimation. Specifically, we implement an Alternating Direction Implicit finite difference scheme.

When using option data in the UKF, we have the choice between prices or implied volatilities. Option prices are extremely sensitive to the short rate relative to the USV, while implied volatilities are more sensitive to changes in USV. It is therefore recommended that option implied volatilities are used to accurately recover the USV process.

A further area of study, would be to perform an empirical investigation using market panel data. The model parameters can be estimated by maximising the log-likelihood function given in Section 4.5. A specific area of interest would be to investigate the split in the volatility of the short rate between the USV and the local-volatility.

Bibliography

Anderson, T. W. (1958). *An Introduction to Multivariate Statistical Analysis*, 3 edn, John Wiley & Sons.

Backwell, A. (2015). Hedging evidence for unspanned stochastic volatility. Available at SSRN: https://ssrn.com/abstract=2701911orhttp://dx.doi.org/10.2139/ssrn.2701911.

Backwell, A. (2017). Bivariate unspanned stochastic volatility models. Available at SSRN: https://ssrn.com/abstract=2818372orhttp://dx.doi.org/10.2139/ssrn.2818372.

Brigo, D. and Mercurio, F. (2007). *Interest rate models-theory and practice: with smile, inflation and credit*, Springer Science & Business Media.

Carr, P., Gabaix, X. and Wu, L. (2011). Linearity-generating processes, unspanned stochastic volatility, and interest-rate option pricing. Available at SSRN: https://ssrn.com/abstract=1789763.

Collin-Dufresne, P. and Goldstein, R. S. (2002). Do bonds span the fixed income markets? theory and evidence for unspanned stochastic volatility, *The Journal of Finance* **57**(4): 1685–1730.

Crépey, S. (2013). *Financial Modeling: A Backward Stochastic Differential Equations Perspective*, Springer Science & Business Media.

Del Moral, P. (1996). Non-linear filtering: Interacting particle resolution, *Markov Processes and Related Fields* **2**(4): 555–581.

D'Errico, J. (2013). Finding the nearest positive definite matrix, https://www.mathworks.com/matlabcentral/fileexchange/42885-nearestspd. MATLAB Central File Exchange.

Duffee, G. R. and Stanton, R. H. (2012). Estimation of dynamic term structure models, *The Quarterly Journal of Finance* **2**(02).

Duffie, D. and Kan, R. (1996). A yield-factor model of interest rates, *Mathematical Finance* **6**(4): 379–406.

Duffy, D. J. (2013). *Finite Difference Methods in Financial Engineering: A Partial Differential Equation Approach*, John Wiley & Sons.

Fong, H. G. and Vasicek, O. A. (1991). Fixed–income volatility management, *The Journal of Portfolio Management* **17**(4): 41–46.

Gellert, K. and Schlögl, E. (2018). Parameter learning and change detection using a particle filter with accelerated adaptation.

Heidari, M. and Wu, L. (2003). Are interest rate derivatives spanned by the term structure of interest rates?, *The Journal of Fixed Income* **13**(1): 75–86.

Hull, J. and White, A. (1990). Pricing interest-rate-derivative securities, *The Review of Financial Studies* **3**(4): 573–592.

Kandepu, R., Foss, B. and Imsland, L. (2008). Applying the unscented kalman filter for nonlinear state estimation, *Journal of Process Control* **18**(7-8): 753–768.

Peaceman, D. W. and Rachford, Jr, H. H. (1955). The numerical solution of parabolic and elliptic differential equations, *Journal of the Society for Industrial and Applied Mathematics* **3**(1): 28–41.

Piazzesi, M. (2010). Affine term structure models, *Handbook of Financial Econometrics* **1**: 691–766.

Vasicek, O. (1977). An equilibrium characterization of the term structure, *Journal of Financial Economics* **5**(2): 177–188.

Wan, E. A. and Van Der Merwe, R. (2000). The unscented kalman filter for nonlinear estimation, *In proceedings of IEEE Symposium 2000 (AS-SPCC)*.

Zucchini, W. and MacDonald, I. L. (2009). *Hidden Markov Models for Time Series: An Introduction using R*, CRC press.

Appendix A

ADI Scheme Details

We calculate our finite difference surface on a $N_r \times N_u$ grid. The discretised option value is given by $U_m^{i,j}$, where i indexes the short rate, j indexes the USV and m indexes time. We discretise the PDE using finite difference approximations for each partial differential. For the partial derivative with respect to time, we use a forward difference, while for the spatial derivatives we use central difference approximations. The spatial approximations can either be computed implicitly or explicitly in time. In an explicit scheme, if we know the values of U at time step m, we can explicitly compute the value at time step $m+1$ since the value for U_{m+1} only depends on values of U at time m. In an implicit scheme, the system of equations must be solved in an implicit manner as the value U_{m+1} depends on other values at the same time step. The ADI scheme uses a combination of the two methods. For the first half step, between time m and $m + \frac{1}{2}$, we solve implicitly in the r direction and explicitly in the u direction for both the first and second order derivatives. The finite difference derivative approximates are as follows:

$$
\begin{aligned}
\frac{\partial F}{\partial \tau} &= \frac{U_{m+\frac{1}{2}}^{i,j} - U_m^{i,j}}{\frac{\delta_\tau}{2}}, \\
\frac{\partial F}{\partial r} &= \frac{U_{m+\frac{1}{2}}^{i+1,j} - U_{m+\frac{1}{2}}^{i-1,j}}{2\delta_r}, \\
\frac{\partial F}{\partial u} &= \frac{U_m^{i,j+1} - U_m^{i,j-1}}{2\delta_u}, \\
\frac{\partial^2 F}{\partial r^2} &= \frac{U_{m+\frac{1}{2}}^{i+1,j} - 2U_{m+\frac{1}{2}}^{i,j} + U_{m+\frac{1}{2}}^{i-1,j}}{\delta_r^2}, \\
\frac{\partial^2 F}{\partial u^2} &= \frac{U_m^{i,j+1} - 2U_m^{i,j} + U_m^{i,j-1}}{\delta_u^2}, \\
\frac{\partial^2 F}{\partial r \partial u} &= \frac{U_m^{i+1,j+1} - U_m^{i+1,j-1} - U_m^{i-1,j+1} + U_m^{i-1,j-1}}{4\delta_r \delta_u}.
\end{aligned}
$$

where the subscript denotes time and the superscripts denote the short rate and the USV respectively. Sorting all implicit terms to the LHS and all explicit terms to the

RHS then factorising gives

$$
\begin{aligned}
&U^{i,j}_{m+\frac{1}{2}}\left(-\frac{2}{\delta_\tau}-\frac{(r(\bar{r}-r)u)^2}{\delta_r^2}-\frac{r}{2}\right)+U^{i+1,j}_{m+\frac{1}{2}}\left(\frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r}+\frac{(r(\bar{r}-r)u)^2}{2\delta_r^2}\right)\\
&+U^{i-1,j}_{m+\frac{1}{2}}\left(-\frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r}+\frac{(r(\bar{r}-r)u)^2}{2\delta_r^2}\right)\\
&=U^{i,j}_{m}\left(-\frac{2}{\delta_\tau}+\frac{\sigma^2u}{\delta_u^2}+\frac{r}{2}\right)+U^{i,j+1}_{m}\left(-\frac{\kappa(\theta-u)}{2\delta_u}-\frac{\sigma^2u}{2\delta_u^2}\right)+U^{i,j-1}_{m}\left(\frac{\kappa(\theta-u)}{2\delta_u}-\frac{\sigma^2u}{2\delta_u^2}\right)\\
&-\left(\frac{U^{i+1,j+1}_m-U^{i+1,j-1}_m-U^{i-1,j+1}_m+U^{i-1,j-1}_m}{4\delta_r\delta_u}\right)\rho(r(\bar{r}-r)u)\sigma\sqrt{u}.
\end{aligned}
$$

Now, letting

$$
\begin{aligned}
\alpha_1&=-\frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r}+\frac{(r(\bar{r}-r)u)^2}{2\delta_r^2},\\
\beta_1&=-\frac{2}{\delta_\tau}-\frac{(r(\bar{r}-r)u)^2}{\delta_r^2}-\frac{r}{2},\\
\gamma_1&=\frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r}+\frac{(r(\bar{r}-r)u)^2}{2\delta_r^2},\\
a_1&=\frac{\kappa(\theta-u)}{2\delta_u}-\frac{\sigma^2u}{2\delta_u^2},\\
b_1&=-\frac{2}{\delta_\tau}+\frac{\sigma^2u}{\delta_u^2}+\frac{r}{2},\\
c_1&=-\frac{\kappa(\theta-u)}{2\delta_u}-\frac{\sigma^2u}{2\delta_u^2}.
\end{aligned}
$$

the problem can be expressed as the following tridiagonal matrix equation:

$$
\begin{aligned}
U^{\cdot,j}_{m+\frac{1}{2}}=G_1^{-1}\Bigg(&b^j_{m+\frac{1}{2}}+a_1U^{\cdot,j-1}_m+b_1U^{\cdot,j}_m+c_1U^{\cdot,j+1}_m\\
&-\frac{1}{4\delta_r\delta_u}\left(U^{\cdot+1,j+1}_m-U^{\cdot+1,j-1}_m-U^{\cdot-1,j+1}_m+U^{\cdot-1,j-1}_m\right)\rho(r(\bar{r}-r)u)\sigma\sqrt{u}\Bigg),
\end{aligned}
$$

for all $2\leq j\leq N_u-1$, and where

$$
G_1=\begin{bmatrix}
\beta_1+\alpha_1 & \gamma_1 & 0 & \cdots & 0 & 0\\
\alpha_1 & \beta_1 & \gamma_1 & 0 & \cdots & 0\\
0 & \alpha_1 & \beta_1 & \gamma_1 & \cdots & 0\\
0 & \vdots & \ddots & \ddots & \ddots & \vdots\\
0 & 0 & \cdots & \alpha_1 & \beta_1 & \gamma_1\\
0 & 0 & \cdots & 0 & \alpha_1 & \beta_1+\gamma_1
\end{bmatrix},\quad
b^j_{m+\frac{1}{2}}=\begin{bmatrix}0\\0\\\vdots\\0\\\delta_r\gamma_1f(T,S)\end{bmatrix}.
$$

Crépey (2013) prescribes this parametrisation of the triadiagonal matrix G_1 and the boundary vector $b^j_{m+\frac{1}{2}}$ when dealing with Neumann boundary conditions. The boundary vector is calculated using the Neumann boundary conditions given in

Chapter 3. The first and last columns of the grid — $U^{\cdot,1}_{m+\frac{1}{2}}$ and $U^{\cdot,N_u}_{m+\frac{1}{2}}$ — are handled separately since both the r and u boundary conditions need to be considered.

In the next half step, between $m+\frac{1}{2}$ and m, we solve implicitly in u and explicitly in r. The finite difference derivative approximations are as follows:

$$\begin{aligned}
\frac{\partial F}{\partial \tau} &= \frac{U^{i,j}_{m+1} - U^{i,j}_{m+\frac{1}{2}}}{\frac{\delta_\tau}{2}}, \\
\frac{\partial F}{\partial r} &= \frac{U^{i+1,j}_{m+\frac{1}{2}} - U^{i-1,j}_{m+\frac{1}{2}}}{2\delta_r}, \\
\frac{\partial F}{\partial u} &= \frac{U^{i,j+1}_{m+1} - U^{i,j-1}_{m+1}}{2\delta_u}, \\
\frac{\partial^2 F}{\partial r^2} &= \frac{U^{i+1,j}_{m+\frac{1}{2}} - 2U^{i,j}_{m+\frac{1}{2}} + U^{i-1,j}_{m+\frac{1}{2}}}{\delta_r^2}, \\
\frac{\partial^2 F}{\partial u^2} &= \frac{U^{i,j+1}_{m+1} - 2U^{i,j}_{m+1} + U^{i,j-1}_{m+1}}{\delta_u^2}, \\
\frac{\partial^2 F}{\partial r \partial u} &= \frac{U^{i+1,j+1}_{m+\frac{1}{2}} - U^{i+1,j-1}_{m+\frac{1}{2}} - U^{i-1,j+1}_{m+\frac{1}{2}} + U^{i-1,j-1}_{m+\frac{1}{2}}}{4\delta_r\delta_u}.
\end{aligned}$$

Sorting all implicit terms to the LHS and all explicit terms to the RHS then factorising gives

$$\begin{aligned}
&U^{i,j}_{m+1}\left(-\frac{2}{\delta_\tau} - \frac{\sigma^2 u}{\delta_u^2} - \frac{r}{2}\right) + U^{i,j+1}_{m+1}\left(\frac{\kappa(\theta-u)}{2\delta_u} + \frac{\sigma^2 u}{2\delta_u^2}\right) + U^{i,j-1}_{m+1}\left(-\frac{\kappa(\theta-u)}{2\delta_u} + \frac{\sigma^2 u}{2\delta_u^2}\right) \\
&= U^{i,j}_{m+\frac{1}{2}}\left(-\frac{2}{\delta_\tau} + \frac{(r(\bar{r}-r)u)^2}{\delta_r^2} + \frac{r}{2}\right) + U^{i+1,j}_{m+\frac{1}{2}}\left(-\frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r} + \frac{(r(\bar{r}-r)u)^2}{2\delta_r^2}\right) \\
&+ U^{i-1,j}_{m+\frac{1}{2}}\left(\frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r} - \frac{(r(\bar{r}-r)u)^2}{2\delta_r^2}\right) \\
&- \left(\frac{U^{i+1,j+1}_{m+\frac{1}{2}} - U^{i+1,j-1}_{m+\frac{1}{2}} - U^{i-1,j+1}_{m+\frac{1}{2}} + U^{i-1,j-1}_{m+\frac{1}{2}}}{4\delta_r\delta_u}\right)\rho(r(\bar{r}-r)u)\sigma\sqrt{u}.
\end{aligned}$$

Along the lower boundary of the USV state domain (where $u_t = 0$) the PDE in Equation (3.3) can reduced to the following:

$$F_\tau - (r_t - \lambda_1)(r_t - \lambda_2)F_r - \kappa\theta F_u + rF = 0.$$

We discretise the partial derivative with respect to the USV state variable using an explicit, forward finite difference as follows:

$$F_u = \frac{U^{i,j+1}_{m+\frac{1}{2}} - U^{i,j}_{m+\frac{1}{2}}}{\delta_u}.$$

The discretised reduced PDE is then given by

$$\frac{U^{i,j}_{m+1} - U^{i,j}_{m+\frac{1}{2}}}{\frac{\delta_\tau}{2}} - (r-\lambda_1)(r-\lambda_2)\frac{U^{i+1,j}_{m+\frac{1}{2}} - U^{i-1,j}_{m+\frac{1}{2}}}{2\delta_r} - \kappa\theta\frac{U^{i,j+1}_{m+\frac{1}{2}} - U^{i,j}_{m+\frac{1}{2}}}{\delta_u} + r\frac{U^{i,j}_{m+1} + U^{i,j}_{m+\frac{1}{2}}}{\frac{\delta_\tau}{2}} = 0.$$

Factorising gives the following expression for the value along the boundary:

$$U^{i,1}_{m+1} = \frac{2}{\delta_\tau(1+r)}\Bigg(\frac{(r-\lambda_1)(r-\lambda_2)}{2\delta_r}\big(U^{i+1,1}_{m+\frac{1}{2}} - U^{i-1,1}_{m+\frac{1}{2}}\big) + \frac{\kappa\theta}{\delta_u}U^{i,2}_{m+\frac{1}{2}} - U^{i,1}_{m+\frac{1}{2}}\big(\frac{2r}{\delta_\tau} + \frac{\kappa\theta}{\delta_u}\big)\Bigg), \tag{A.1}$$

noting that $u = 0$ corresponds to $j = 1$.

Now, letting

$$\begin{aligned}
\alpha_2 &= -\frac{\kappa(\theta-u)}{2\delta_u} + \frac{\sigma^2 u}{2\delta_u^2},\\
\beta_2 &= -\frac{2}{\delta_\tau} - \frac{\sigma^2 u}{\delta_u^2} - \frac{r}{2},\\
\gamma_2 &= \frac{\kappa(\theta-u)}{2\delta_u} + \frac{\sigma^2 u}{2\delta_u^2},\\
a_2 &= \frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r} - \frac{(r(\bar r - r)u)^2}{2\delta_r^2},\\
b_2 &= -\frac{2}{\delta_\tau} + \frac{(r(\bar r - r)u)^2}{\delta_r^2} + \frac{r}{2},\\
c_2 &= -\frac{(r-\lambda_1)(r-\lambda_1)}{2\delta_r} + \frac{(r(\bar r - r)u)^2}{2\delta_r^2},
\end{aligned}$$

the problem can be expressed as the following tridiagonal matrix equation:

$$U^{i,\cdot}_{m+1} = \Bigg(b^i_{m+1} + a_2 U^{i-1,\cdot}_{m+\frac{1}{2}} + b_2 U^{i,\cdot}_{m+\frac{1}{2}} + c_2 U^{i+1,\cdot}_{m+\frac{1}{2}} - \frac{1}{4\delta_r\delta_u}\big(U^{i+1,\cdot+1}_{m+\frac{1}{2}} - U^{i+1,\cdot-1}_{m+\frac{1}{2}} - U^{i-1,\cdot+1}_{m+\frac{1}{2}} + U^{i-1,\cdot-1}_{m+\frac{1}{2}}\big)\rho(r(\bar r - r)u)\sigma\sqrt{u}\Bigg)G_2^{-1},$$

for all $2 \le i \le N_r - 1$, and where

$$G_2 = \begin{bmatrix} \beta_2 & \gamma_2 & 0 & \cdots & 0 & 0\\ \alpha_2 & \beta_2 & \gamma_2 & 0 & \cdots & 0\\ 0 & \alpha_2 & \beta_2 & \gamma_2 & \cdots & 0\\ 0 & \vdots & \ddots & \ddots & \ddots & \vdots\\ 0 & 0 & \cdots & \alpha_2 & \beta_2 & \gamma_2\\ 0 & 0 & \cdots & 0 & \alpha_2 & \beta_2 \end{bmatrix}, \quad b^i_{m+1} = \begin{bmatrix} U^{i,1}_{m+1}\\ 0\\ \vdots\\ 0\\ 2U^{i,N_u}_{m+\frac{1}{2}} - U^{i,N_u-1}_{m+\frac{1}{2}} \end{bmatrix}^\top.$$

$U^{i,1}_{m+1}$ is calculated using Equation (A.1). We treat the boundary conditions as Dirichlet conditions, and parametrise G_2 and b^i_{m+1} according to Crépey (2013). The first and last rows of the grid — $U^{1,\cdot}_{m+\frac{1}{2}}$ and $U^{N_r,\cdot}_{m+\frac{1}{2}}$ — are handled separately since both the r and u boundary conditions need to be considered.