USLV: Unspanned Stochastic-Local Volatility Model *

Igor Halperin[1,2] and Andrey Itkin[3,2]

[1]MR&D, JPMorgan Chase, 270 Park Avenue, New York, NY 10172, USA,
igor.halperin@jpmorgan.com

[2]Polytechnic Institute of New York University, 6 Metro Tech Center, RH 517E, Brooklyn NY 11201, USA

[3]Numerix LLC, 150 East 42nd Street, 15th Floor, New York, NY 10017, USA,
aitkin@numerix.edu

January 15, 2013

Abstract

We propose a new framework for modeling stochastic local volatility, with potential applications to modeling derivatives on interest rates, commodities, credit, equity, FX etc., as well as hybrid derivatives. Our model extends the linearity-generating unspanned volatility term structure model by Carr et al. (2011) by adding a local volatility layer to it. We outline efficient numerical schemes for pricing derivatives in this framework for a particular four-factor specification (two "curve" factors plus two "volatility" factors). We show that the dynamics of such a system can be approximated by a Markov chain on a two-dimensional space (Z_t, Y_t), where coordinates Z_t and Y_t are given by direct (Kroneker) products of values of pairs of curve and volatility factors, respectively. The resulting Markov chain dynamics on such partly "folded" state space enables fast pricing by the standard backward induction. Using a nonparametric specification of the Markov chain generator, one can accurately match arbitrary sets of vanilla option quotes with different strikes and maturities. Furthermore, we consider an alternative formulation of the model in terms of an implied time change process. The latter is specified nonparametrically, again enabling accurate calibration to arbitrary sets of vanilla option quotes.

[*] Opinions expressed in this paper are those of the authors, and do not necessarily reflect the view of JPMorgan Chase and Numerix.

1 Introduction

1.1 Motivation

The present work is motivated by the desire to have a unified modeling methodology and shared implementation for derivatives pricing with a dynamic volatility smile for various asset classes, including interest rates (IR), commodities, equities, credit, foreign exchange (FX) etc., as well as for modeling hybrid derivatives such as equity-IR or equity-commodities hybrids. We present one possible approach, which extends a recently proposed class of stochastic volatility models.

1.2 Related Previous Work

Gabaix (2007) proposed a new class of asset price models, the so-called *linearity-generating processes* (LGP). Such processes are defined by the condition that the current prices of basic instruments (stock, bonds, futures, swaps etc.) are *linear* in a driving Markov process X_t. This stands in sharp contrast to popular *affine* models where, e.g., a zero-coupon bond price $P(t, T)$ is an exponentially-affine function of a Markov driver X_t.

On the theoretical side, the LGP processes appear very attractive. Indeed, typical ways we model basic instruments are drastically different between, say, IR and equity models[1]. In the equity world, basic instruments (equities) are linear in stochastic factors (usually taken to be equity prices themselves for purposes of modeling derivatives), and volatility is stochastic (SV) and unspanned (USV, see below for the definition of this term).

In the IR world, the mathematics are almost the same for the HJM-type models that model the entire yield curve. As the yield curve is in one-to-one correspondence with bond prices, it can be viewed as an "observable" basic instrument that again is linear in factors, and typically gives rise to USV.

But such linearity of HJM-like models has a high price, namely that the number of state variables needed for the Markovian dynamics turns out to be too high for use in a lattice-based setting in most cases of practical interest. Therefore, even with Markovian specifications, HJM-like models are typically employed within a Monte Carlo setup rather than on a lattice. On the other hand, an attempt to reduce the curve modeling to a short-rate modeling, as is done in affine models, leads to a nonlinear relation between bond prices and the factors, which produces undesirable side effects, such as a dependence of the instantaneous forward curve on the short-rate volatility.[2]

This problem is resolved in the LGP approach. By putting both equity and bonds on equal footing in terms of making them both linear functionals of the factors (and doing

[1] Both are taken to be examples of term-structure models vs. spot-models. Instead of IR and equity, we could, e.g. compare commodities and FX.

[2] This is the cost one has to pay for non-linearity. Clearly, nothing similar ever occurs in spot models: today's stock price S_t is obviously independent of the current volatility or current value of the volatility factor Y_t. Mathematically, this can be formulated as the statement that for spot stochastic volatility models (such as e.g. the Heston model), the pricing function $f(S_t, Y_t)$ of basic instruments (stocks) is an identity $f(S_t, Y_t) = S_t$, see also Table 1 in Sect. 2.

it in a different way from HJM), the LGP-based models play a role of "grand unification models," similar in a conceptual sense to "grand unification theories" (GUT) in physics. No proliferation of the number of Markov drivers occurs in LGP-type models as we move from one class of basic instruments (stocks) to other class (bonds).

Also on the practical side, linearity has profound consequences for tractability of asset pricing modeling within the LGP framework. In particular, if a zero-coupon bond price is linear in X_t, then so will be prices of a coupon bond or a swap. As a result, the swaption pricing, e.g. can be done in a semi-analytical form without additional approximations, such as those used by the Libor Market Models (LMM). It is also very helpful in calibration, as will be discussed in more detail below.

To summarize, the class of LGP-like models identified by Gabaix is a new interesting class that may develop into a viable competitor to both affine models, which are currently one of the main workhorses for derivatives modeling in credit, commodities, rates and other asset classes, and also HJM-type models. Yet this approach is in its infancy compared to the well-studied class of affine models.

In 2011, Carr, Gabaix and Wu (CGW) proposed a LGP-type stochastic volatility term structure model (Carr et al. (2011)). CGW, in particular, emphasize the point that stochastic volatility generated in LGP-type models is *unspanned* in the sense of the definition of Colin-Dufresne & Goldstein (2002), who coined the original term "unspanned volatility"[3]. The CGW model offers a number of attractive features. Most importantly, it is a low-dimension Markov model with unspanned stochastic volatility (USV), and an orthogonal set of model parameters with a separate calibration to the term structure and option volatilities.

The CGW model is a pure stochastic volatility model, as volatility is modeled as a superposition of CIR processes. To make it more practical, it would be very useful to add a local volatility layer to the model. Our extension of the CGW model amounts to introduction of such a local volatility factor, along with efficient numerical methods for calibration and pricing. To differentiate our framework from CGW, in what follows we will refer to it as the unspanned stochastic local volatility (USLV) model.

2 Overview of Our Framework

By construction, USLV preserves the linearity and USV properties of the CGW approach. Another property inherited from the CGW model is that USLV is formulated directly in the physical measure $\mathbb{P}$ (see below) rather than in the risk-neutral measure $\mathbb{Q}$, which makes it easier, e.g., to combine the historical and pricing data for model estimation, if desired.

The main theoretical construction that USLV adds to the CGW model is a local volatility layer. The resulting mixed stochastic/local volatility dynamics has a few important implications.

First, adding a local volatility layer enables nearly perfect matching of an arbitrary

[3] Following Colin-Dufresne & Goldstein (2002), the volatility is called unspanned if bond prices do not depend on the stochastic volatility driver.

number of European vanilla option quotes with different strikes and maturities.[4] Such an extension is clearly desirable in order to apply this approach for pricing of both vanilla and exotic derivatives, especially if vanilla options are used to hedge the exotics.

Second, the presence of a local volatility layer alongside a stochastic volatility part induces a decomposition of the option volatility into spanned and unspanned parts, rather than being of a pure unspanned type as in the CGW model. One could expect that such decomposition of volatility should translate into a decomposition of an option's vega into a delta-vega and a "genuine vega" part.

Because of the way our model is calibrated, it enables traders to incorporate their view on the relative weights of the spanned and unspanned parts in the option's vegas.[5] By viewing a trader's inputs as a prior model that does not necessarily match observed options, our model finds a minimal adjustment ("tweak") to the trader's prior model in order to reinforce an accurate match of the option quotes.

In contrast, the volatility in local volatility models would be 100% spanned. In local volatility models, matching vanilla pricing would fix the volatility surface for all strikes and maturities, and would not leave any flexibility for the model to match prices of more exotic options. The inclusion of stochastic volatility allows one to simultaneously have more realistic forward smile dynamics and additional parameters to match exotics' prices (if available). The ability of USLV to incorporate a possible trader's view is what sets it apart from both pure local volatility models and pure SV models of the CGW type.

On the implementation side, USLV concentrates on the most important low-dimensional specifications for practicality, e.g., two factors for the term structure (with N curve factors in general), and one or two factors for stochastic volatility (with M volatility factors in general). In particular, for a (2+2)-factor case, we show how to approximate the dynamics of the driving factors by a two-dimensional Markov chain on a space constructed by folding (see below) of the original four-dimensional state space. This enables fast pricing by standard backward induction on the chain.

It should be noted that while in this paper we concentrate on modeling term-structure dynamics (e.g., of futures, swap rates or credit spreads) with potential applications to "term structure asset classes," such as IR, commodities or credit, the same approach can be used for modeling spot prices, which would be a proper setting for "spot asset classes," such as equities or FX. Moreover, due to a symmetric treatment of "term-structure assets" and "spot assets" in the present framework, this approach is readily available for modeling hybrid derivative products (e.g., equity-IR or equity-commodity hybrids) using the same implementation. Changes from one asset class to another would amount to a proper reparametrization and reinterpretation of the Markov generator matrices while leaving the computational algorithm intact.[6] Furthermore, in the continuous limit, different parametrizations of the stochastic

[4] Note that while this property of USLV is shared by local stochastic volatility models as well, the key point here is that now we have an additional risk factor (volatility) to acknowledge, model and hedge.

[5] Technically, this is done by giving the end user the ability to input his/her own set of speed factors (SF), see below.

[6] In principle, this could produce a generic pricing engine, similar in a sense to Monte Carlo (MC). Indeed, the latter method is a "universal" method of derivatives pricing in the sense that in this framework, we only

volatility generator in our Markov chain model give rise to a rich class of (2+2)-factor models, including stochastic local volatility with jumps. Note that in this paper we primarily concentrate on specifications whose continuous limit is a two-dimensional diffusion with a two-dimensional diffusive stochastic local volatility. This case is covered in detail below in Sect. 6. However, in Sect. 7, we will present an alternative formulation that can give rise to jumps in both the underlying and stochastic volatility. Our approach is thus quite flexible in its ability to accommodate different specifications of the dynamics, including a four-factor stochastic local volatility model with jumps.

2.1 USLV vs. HJM vs. Affine Models

Our initial interest in using LGP-type models for a potential model for stochastic volatility was inspired by the observation that LGP-type models (and, by extension, USLV-like models) seem to combine the best features of both HJM-type models and affine models, while avoiding their disadvantages. Indeed, like the HJM-type models, the stochastic volatility is unspanned in USLV. Unlike the HJM-types, the model is Markov in dimension $N + M$ rather than $N + N(N+1)/2 + M$, as in HJM-type models. Conversely, both affine models and USLV have the same number of state variables ($N + M$). However, in USLV, volatility is always (partly) unspanned, while in affine models, volatility in general will be spanned unless some special constraints are imposed on parameters, which might be restrictive for calibration purposes. (See also Table 1 below.)

The above reasoning suggests that if we manage to generalize the pure stochastic volatility model of CGW to a stochastic local volatility model (i.e., to make a USLV out of CGW), and do it in a numerically efficient way, and if the resulting model demonstrates good parameters and hedges stability etc., then such a model can be considered a viable candidate for use in practice. This paper outlines the theoretical framework for USLV, leaving numerical experiments for future work.

A few more words of caution are in order here. Our outline of the USLV is generic and is not tied yet to any specific asset class. Each asset class makes its own demands on a model. For example, the ability to reproduce the Samuelson effect and asset cointegration are very important for commodities, alongside the ability to handle seasonality in asset levels and volatilities for certain commodities, such as gas or power. It has yet to be seen how (or whether) the USLV framework can accommodate such specific requirements. A discussion of this matter is planned for the second stage of the present theoretical work.

A brief summary of different model classes is presented in Table 1, where we compare the behavior of equity stochastic volatility models such as the Heston model, HJM-type, affine-type and LGP/CGW/USLV-type. The third column shows the functional form of

need to implement dynamic equations and payoff functions for a particular model-product combination in order to use a generic MC engine. Likewise, our Markov chain framework is “universal” in the same sense (within a class of all diffusive local stochastic volatility models in up to (2+2) dimensions). The only difference here is that while in MC we typically start with continuous space dynamics, which is then discretized for simulation through discretization of processes (e.g., Brownian motions) driving the dynamics, the dynamics in our approach are fundamentally defined in terms of discretized state variables.

Model	BI_t	BV_t	$BI_t = f(BV_t)$	USV	D
Equity	S_t	S_t, Y_t	$f(S_t, Y_t) = S_t$	Yes	$1 + M$
IR HJM	P_t^T	P_t^T, Y_t	$f(P_t^T, Y_t) = P_t^T$	Yes	$N + N(N+1)/2 + M$
IR Affine	P_t^T	X_t, Y_t	$f(X_t, Y_t)$	No	$N + M$
CGW/USLV	P_t^T	X_t, Y_t	$f(X_t, Y_t) = \alpha_{tT} X_t + \beta_{tT}$	Yes	$N + M$

Table 1: Model comparison summary. Note that "Yes" in column USV for IR HJM means "in general, yes," and likewise "No" for IR Affine means "in general, no, unless special 'knife-edge' constraints are imposed on parameters of the model." Here S_t, P_t^T and Y_t stand for the stock and bond prices and volatility factor, respectively, while BI_t and BV_t stand for basic instruments and basic variables, respectively. Finally, D stands for for the total number of state variables needed for a Markovian description.

conditional expectations arising in calculation of prices of elementary instruments.

3 The Carr-Gabaix-Wu Model

In this section, we provide a brief overview of the CGW model of Carr et al. (2011). The CWG model is then used as the first step in our setting. Simultaneously, in this section we set our notation, on which we largely follow Carr et al. (2011).

3.1 State-Price Processes and Martingale Pricing

The famous fundamental theorem of asset pricing (Harrison & Pliska (1981)) states that if the economy is arbitrage free, then there exists, under certain technical conditions such as positivity and time consistence, a strictly positive process M_t called the *state space deflator*, such that the deflated gain process associated with any admissible trading strategy is a martingale under the measure $\mathbb{P}$. In particular, for a contingent payoff Π_T at time $T > t$, its value at time t is given by the following $\mathbb{P}$-conditional expectation:

$$V(t,T) = \mathbb{E}_t \left[\frac{M_T}{M_t} \Pi_T \right]$$

The ratio M_T/M_t is sometimes referred to as the stochastic discount factor or the pricing kernel. The $\mathbb{P}$-measure SDE for M_t reads

$$\frac{dM_t}{M_t} = -r dt - \gamma(Z_t) dZ_t,$$

where Z_t is a vector of risk factors and $\gamma(Z_t)$ measures the market prices of risk for these factors. The formal solution to this SDE takes a multiplicative form

$$M_t = M_0 \exp\left(-\int_0^t r_s ds \right) \mathcal{E}\left(-\int_0^t \gamma(Z_s) dZ_s \right),$$

where $\mathcal{E}(\cdot)$ stands for the stochastic exponential martingale operator. The latter defines the Radon-Nikodým derivative $d\mathbb{Q}/d\mathbb{P}$ that transforms the physical measure $\mathbb{P}$ to the risk-neutral measure $\mathbb{Q}$ such that, under $\mathbb{Q}$, the contingent claim valuation reads

$$V(t,T) = \mathbb{E}_t^{\mathbb{Q}} \left[\exp\left(- \int_t^T r_s ds \right) \Pi_T \right]. \tag{1}$$

3.2 One-Factor Case

Assume that the state variable X_t is driven by the following SDE under the measure $\mathbb{Q}$:

$$dX_t = -\kappa X_t \left(1 - X_t\right) dt + dn_t, \tag{2}$$

where n_t stands for a martingale component that we will leave unspecified for a while.[7] The short rate r_t is obtained from X_t by a linear transformation:

$$r_t = \theta_r + \kappa X_t. \tag{3}$$

Note that in order to prevent exploding solutions of Eq.(2), X_t has to be constrained to live on the unit interval, $0 \leq X_t \leq 1$. We will return to this point below.

As defined by Gabaix (2007), a linearity-generating process (LGP) is characterized by two requirements: (i) The time-t zero-coupon bond price is linear in the state vector X_t, and (ii) the time-t conditional expectation of the deflated state vector X_{t+1} is linear in X_t:[8]

$$\begin{aligned} P(t,T) = \mathbb{E}_t \left[\frac{M_T}{M_t} \right] &= \alpha(t,T) + \delta(t,T) X_t, \\ \mathbb{E}_t \left[\frac{M_T}{M_t} X_T \right] &= \gamma(t,T) + \Gamma(t,T) X_t, \end{aligned} \tag{4}$$

where $\alpha(t,T), \delta(t,T)$ and $\Gamma(t,T)$ are some functions, while $\gamma(t,T)$ is set to zero in Gabaix (2007). To handle both constraints, Gabaix proposes the following compact description of dynamics in terms of the two-component vector

$$\mathbf{Y}_t = \begin{pmatrix} 1 - X_t \\ X_t \end{pmatrix} M_t. \tag{5}$$

The explicit calculation yields

$$\begin{aligned} \mathbb{E}_t \left[dY_t^{(1)} \right] &= \mathbb{E}_t \left[d\left((1 - X_t) M_t \right) \right] = \left(1 - X_t\right) M_t \mathbb{E}_t \left[\frac{dM_t}{M_t} \right] - M_t \left(\mathbb{E}_t \left[dX_t \right] + \left[dX, \frac{dM}{M_t} \right]_t \right) \\ &= -\left(1 - X_t\right) M_t r_t dt - M_t \mathbb{E}_t^{\mathbb{Q}} \left[dX_t \right] = -\left(1 - X_t\right) M_t r_t dt + \kappa M_t X_t \left(1 - X_t\right) dt \\ &= \left(1 - X_t\right) M_t \left[-(\theta_r + \kappa X_t) + \kappa X_t \right] dt = -\theta_r Y_t^{(1)} dt. \end{aligned} \tag{6}$$

[7] Note that our definition of X_t differs from that suggested in Gabaix (2007). For consistency of notation in the one factor and multi-factor cases, our definition rescales by κ.

[8] Note that in this paper all expectations assume the physical measure $\mathbb{P}$ unless stated otherwise.

A similar calculation produces

$$\mathbb{E}_t\left[dY_t^{(2)}\right] = -\left(\theta_r + \kappa\right) Y_t^{(2)} dt. \tag{7}$$

We emphasize that these results are independent of the specification of the martingale n_t in Eq.(2). Eq.(6) and Eq.(7) can be jointly written in a vector form

$$\mathbb{E}_t\left[d\mathbf{Y}_t\right] = -\mathbf{A}\mathbf{Y}_t dt, \tag{8}$$

where $\mathbf{A}$ is the Markov generator matrix

$$\mathbf{A} = \left(\begin{array}{cc} \theta_r & 0 \\ 0 & \kappa + \theta_r \end{array}\right).$$

The solution of Eq.(8) is then obtained by the matrix exponential:

$$\mathbb{E}_t\left[\mathbf{Y}_T\right] = e^{-\mathbf{A}(T-t)}\mathbf{Y}_t = e^{-\theta_r \tau}\left(\begin{array}{c} 1 - X_t \\ e^{-\kappa\tau} X_t \end{array}\right) M_t, \tag{9}$$

where $\tau = T - t$.

Now observe that the map $M_t \to \mathbf{Y}_t$ defined in Eq.(5) can be inverted by summing over the indices $i = 1, 2$ of $\mathbf{Y}_t$. This can be written in the equivalent form $M_t = \boldsymbol{\nu}\mathbf{Y}_t$, where $\boldsymbol{\nu} = (1, 1)$. Then, using Eq.(9), we obtain the following relation for the zero-coupon bond price:

$$P(t,T) = \mathbb{E}_t\left[\frac{M_T}{M_t}\right] = \frac{1}{M_t}\mathbb{E}_t\left[\boldsymbol{\nu}\mathbf{Y}_T\right] = \frac{1}{M_t}\boldsymbol{\nu} e^{-\mathbf{A}\tau}\mathbf{Y}_t = e^{-\theta_r\tau}\left(1 - \left(1 - e^{-\kappa\tau}\right) X_t\right), \tag{10}$$

which can be compared to Eq.(4). Note that Eq.(10) implies that $P(t,t) = 1$, as it should. We emphasize again that Eq.(10) holds for any specification of the martingale n_t in Eq.(2).

3.3 Z-Parametrization

Carr, Gabaix and Wu (CGW) find it convenient to reparametrize Eq.(5) as follows:

$$\mathbf{Y}_t = e^{-\theta_r t}\left(\begin{array}{c} \alpha_0 + \beta_0 Z_t \\ e^{-\kappa t}\left(\alpha_1 + \beta_1 Z_t\right) \end{array}\right) = e^{-\mathbf{A}t}\left(\begin{array}{c} \alpha_0 + \beta_0 Z_t \\ \alpha_1 + \beta_1 Z_t \end{array}\right) \tag{11}$$

where $(\alpha_0, \alpha_1, \beta_0, \beta_1)$ are scalar coefficients and Z_t is a nonnegative $\mathbb{P}$-martingale that starts at $Z_0 = 1$. We assume that parameters are chosen such that $\alpha_0 + \beta_0 Z_t \geq 0$ and $\alpha_1 + \beta_1 Z_t \geq 0$ for any $t \geq 0$. We postpone a specification of the dynamics of Z_t until Sect. 4, while in this section we concentrate on the results that are independent of this specification.

Equating Eq.(5) and Eq.(11), we impose the following relations between X_t, Z_t and M_t:

$$\begin{aligned} e^{-\theta_r t}\left(\alpha_0 + \beta_0 Z_t\right) &= \left(1 - X_t\right) M_t, \\ e^{-(\theta_r + \kappa)t}\left(\alpha_1 + \beta_1 Z_t\right) &= X_t M_t. \end{aligned} \tag{12}$$

Resolving this with respect to X_t , M_t in terms of Z_t, we obtain (here $\boldsymbol{\nu} = (1,1)$ as above)

$$M_t = \boldsymbol{\nu}\mathbf{Y}_t = e^{-\theta_r t}\left[\alpha_0 + \beta_0 Z_t + e^{-\kappa t}\left(\alpha_1 + \beta_1 Z_t\right)\right], \tag{13}$$

$$X_t = \frac{e^{-\kappa t}\left(\alpha_1 + \beta_1 Z_t\right)}{\alpha_0 + \beta_0 Z_t + e^{-\kappa t}\left(\alpha_1 + \beta_1 Z_t\right)} = \frac{1}{M_t} e^{-(\theta_r + \kappa)t}\left(\alpha_1 + \beta_1 Z_t\right). \tag{14}$$

We interpret these relations as follows. Eq.(13) produces the state price deflator M_t as a functional of the Markov driver Z_t. Note that this is a linear functional, which is exactly the reason we do not have any convexity corrections in Eq.(10). For any nonlinear functional, a resulting expression for $P(t,T)$ would depend on volatility of Z_t. Such behavior is intentionally avoided in the present framework.

The second equation, Eq.(14), computes a map $Z_t \to X_t$. The main property of this map is that, by construction, the bond price is linear in X_t according to Eq.(10), so that X_t can be viewed as a quasi-observable factor in the sense of being directly linked to bond prices.[9] Note that Eq.(14) implies that that X_t lives on the unit interval $0 \leq X_t \leq 1$ as long as $\alpha_0 + \beta_0 Z_t \geq 0$ and $\alpha_1 + \beta_1 Z_t \geq 0$, as was assumed above. This ensures that Eq.(2) is well behaved, and in particular does not give rise to exploding solutions.

The dynamics of the state price deflator M_t and the mapped factor X_t can now be obtained using Ito's lemma for Eq.(13) and Eq.(14). The $\mathbb{P}$-measure SDE for M_t reads

$$\frac{dM_t}{M_t} = -(\theta_r + \kappa X_t)dt - \gamma(Z_t)dZ_t \equiv -r_t dt - \gamma(Z_t, t)dZ_t, \tag{15}$$

where the market price of Z-risk is

$$\gamma(Z_t, t) = -\frac{\beta_0 + \beta_1 e^{-\kappa t}}{\alpha_0 + \beta_0 Z_t + e^{-\kappa t}(\alpha_1 + \beta_1 Z_t)}.$$

Now let us derive the SDE for X_t under the measure $\mathbb{Q}$. For the diffusion function, we use Eq.(14) to find the following expression:

$$\sigma(Z_t, t) \equiv \frac{\partial X_t}{\partial Z_t} = \frac{e^{-\kappa t}\left(\alpha_0 \beta_1 - \alpha_1 \beta_0\right)}{\left[\alpha_0 + \beta_0 Z_t + e^{-\kappa t}\left(\alpha_1 + \beta_1 Z_t\right)\right]^2}. \tag{16}$$

To get the risk-neutral drift of X_t, a "naive" use of Ito's lemma for Eq.(14) will not do the job, as we do not know the drift of Z_t under the measure $\mathbb{Q}$. All we know is that Z_t is a $\mathbb{P}$-martingale.

To proceed with the calculation, we start with the definition of the risk-neutral drift:

$$\mathbb{E}_t^{\mathbb{Q}}\left[dX_t\right] = \mathbb{E}_t\left[dX_t\right] + \left[dX_t, \frac{dM_t}{M_t}\right] = \mathbb{E}_t\left[\frac{d\left(M_t X_t\right)}{M_t}\right] - X_t \mathbb{E}_t\left[\frac{dM_t}{M_t}\right].$$

[9] One can notice here a certain conceptual similarity between the CGW and Markov functional models on the one hand, and a difference with the affine models on the other hand. For the latter, the driving SDE has affine drift and diffusion coefficients, while the function $f(X_t, Y_t)$ is *nonlinear* (nonaffine). For the CGW model and other LGP-type models, the situation is reversed: the state equation is now *nonaffine* but the pricing equation is *affine*.

Both terms entering here can be easily evaluated. Using Eq.(14) and Eq.(15), we obtain, respectively:

$$\begin{aligned}\mathbb{E}_t\left[\frac{d\left(M_t X_t\right)}{M_t}\right] &= \mathbb{E}_t\left[\frac{d\left(e^{-(\theta_r+\kappa)t}\left(\alpha_1+\beta_1 Z_t\right)\right)}{M_t}\right] = -(\theta_r+\kappa)X_t dt, \\ \mathbb{E}_t\left[\frac{dM_t}{M_t}\right] &= -(\theta_r+\kappa X_t)dt. \end{aligned} \tag{17}$$

Putting this together, we obtain the risk-neutral drift of X_t:

$$\mathbb{E}_t^{\mathbb{Q}}\left[dX_t\right] = \left[-(\theta_r+\kappa)X_t + X_t(\theta_r+\kappa X_t)\right]dt = -\kappa X_t\left(1-X_t\right)dt. \tag{18}$$

This is exactly the risk-neutral drift that enters the SDE Eq.(2), which demonstrates self-consistency of the formalism. To identify a diffusion term, we derive a SDE for X_t under the measure $\mathbb{P}$. This is obtained by using Itó's lemma for Eq.(14). To this end, we assume the following dynamics of Z_t under the measure $\mathbb{P}$:

$$\frac{dZ_t}{Z_t} = \hat{\sigma}(Z_t,t)dW_t, \tag{19}$$

where $\hat{\sigma}(Z_t,t)$ is a local volatility function that will be specified below. This yields

$$\begin{aligned} dX_t &= -\kappa X_t\left(1-X_t\right)dt + \sigma(Z_t,t)\left(dZ_t - \frac{\left(\beta_0+\beta_1 e^{-\kappa t}\right)Z_t^2\hat{\sigma}^2(Z_t,t)}{\alpha_0+\alpha_1 e^{-\kappa t}+\left(\beta_0+\beta_1 e^{-\kappa t}\right)Z_t}dt\right) \\ &\equiv -\kappa X_t\left(1-X_t\right)dt + \sigma(Z_t,t)dZ_t^{(Q)}, \end{aligned} \tag{20}$$

where

$$dZ_t^{(Q)} = dZ_t - \frac{\left(\beta_0+\beta_1 e^{-\kappa t}\right)Z_t^2\hat{\sigma}^2(Z_t,t)}{\alpha_0+\alpha_1 e^{-\kappa t}+\left(\beta_0+\beta_1 e^{-\kappa t}\right)Z_t}dt = Z_t\hat{\sigma}(Z_t,t)\left(dW_t + \gamma(Z_t)Z_t\hat{\sigma}(Z_t,t)dt\right). \tag{21}$$

Note that this is a $\mathbb{Q}$-martingale. Indeed, using Eq.(19), we can write the stochastic exponential operator in terms of the usual exponential $\mathbb{P}$-martingale:

$$\mathcal{E}\left(-\int_0^t \gamma(Z_s)dZ_s\right) = e^{\int_0^t b(s)dW_s - \frac{1}{2}\int_0^t (b(s))^2 ds}, \quad b(t) = -\gamma(Z_t)Z_t\hat{\sigma}(Z_t,t). \tag{22}$$

By Girsanov's theorem, the new Brownian motion $W_t^{(Q)}$ with $dW_t^{(Q)} = dW_t - b(t)dt$ is a $\mathbb{Q}$-martingale, which is exactly the combination that arises in Eq.(21).

Comparing Eq.(18) and Eq.(20), we obtain the SDE under the measure $\mathbb{Q}$:

$$dX_t = -\kappa X_t\left(1-X_t\right)dt + \sigma(Z_t,t)\,dZ_t^{(Q)}, \tag{23}$$

where the (parametric) local volatility $\sigma(Z_t,t)$ is defined in Eq.(16), and $Z_t^{(Q)}$ defined in Eq.(21) is a $\mathbb{Q}$-martingale.

While the SDE Eq.(23) may look complicated due to the presence of a local volatility function in Eq.(21) that defines the measure $\mathbb{Q}$, fortunately Eq.(23) is *not* used for pricing derivatives in the LGP model of Carr et al. (2011). Instead, it is the $\mathbb{P}$-measure dynamics Eq.(19) of Z_t that drives derivatives prices in this framework. We recall that Z_t is a $\mathbb{P}$-martingale with $Z_0 = 1$, which otherwise can be arbitrary in Eq.(23). We will deal with a specification of dynamics of Z_t in Sect. 4.

3.4 Extension to a Multi-Factor Case

The previous formulation treats a one-factor case $N = 1$. For an arbitrary number of factors $N \geq 1$, state variables are defined similarly to Eq.(11):

$$\mathbf{Y}_t = e^{-\mathbf{A}t}\left(\mathbf{C} + \mathbf{B}\mathbf{Z}_t\right) = e^{-\mathbf{A}t}\left(\begin{array}{c} C_0 + (\mathbf{B}\mathbf{Z}_t)_0 \\ C_1 + (\mathbf{B}\mathbf{Z}_t)_1 \\ \vdots \\ C_N + (\mathbf{B}\mathbf{Z}_t)_N \end{array}\right),$$

where $\mathbf{C} = (C_0, C_1, \ldots, C_N)^\top$ is a parameter vector, $\mathbf{B}$ is a factor loading matrix of size $(N+1) \times N$, and $\mathbf{Z}_t = \left(Z_t^{(1)}, \ldots, Z_t^{(N)}\right)^\top$ is a non-negative vector-valued $\mathbb{P}$-martingale that starts at $\mathbf{Z}_0 = (1, \ldots, 1)^T$.

Following Carr et al. (2011), we use a diagonal generator $\mathbf{A}$:

$$\boldsymbol{A} = \theta_r I_{N+1} + \operatorname{diag}\left(\kappa_0, \kappa_1, \ldots, \kappa_N\right),$$

where I_{N+1} stands for the identity matrix of size $(N+1) \times (N+1)$, and $\kappa_0 = 0$. The state vector Y_t now takes the form

$$\mathbf{Y}_t = e^{-\theta_r t}\left(\begin{array}{c} e^{-\kappa_0 t}\left(C_0 + (\mathbf{B}\mathbf{Z}_t)_0\right) \\ e^{-\kappa_1 t}\left(C_1 + (\mathbf{B}\mathbf{Z}_t)_1\right) \\ \vdots \\ e^{-\kappa_N t}\left(C_N + (\mathbf{B}\mathbf{Z}_t)_N\right) \end{array}\right) = M_t \left(\begin{array}{c} X_t^0 \\ X_t^1 \\ \vdots \\ X_t^N \end{array}\right) = M_t \mathbf{X}_t. \tag{24}$$

As can be easily checked, we recover the previous relation Eq.(11) from these formulae when $N = 1, C_i = \alpha_i$ and $B_{i1} = \beta_i$ with $i = 0, 1$ if we set $X_t^1 = X_t$ and $X_t^0 = 1 - X_t$.

Again, the map $M_t \to \mathbf{Y}_t$ defined by Eq.(24) can be inverted by introducing a $(N+1)$-component vector $\boldsymbol{\nu} = (1, \ldots, 1)$ such that $\boldsymbol{\nu}\mathbf{X}_t = 1$ for any t. This results in the following generalization of Eq.(13) for the state-price deflator M_t:

$$M_t = \boldsymbol{\nu}\mathbf{Y}_t = e^{-\theta_r t}\sum_{i=0}^{N} e^{-\kappa_i t}\left(C_i + (BZ_t)_i\right). \tag{25}$$

To resolve the constraint $\boldsymbol{\nu}\mathbf{X}_t = 1$ just introduced, we choose $X_t^0 = 1 - \sum_{i=1}^N X_t^i$. Now for a zero-coupon bond we obtain, similarly to Eq.(10),

$$\begin{aligned} P(t,T) &= \mathbb{E}_t\left[\frac{M_T}{M_t}\right] = \frac{1}{M_t}\boldsymbol{\nu}\mathbb{E}_t\left[\mathbf{Y}_T\right] = \frac{1}{M_t}\boldsymbol{\nu}e^{-\mathbf{A}T}\mathbf{Y}_t = \boldsymbol{\nu}e^{-\mathbf{A}\tau}\mathbf{X}_t \\ &= e^{-\theta_r\tau}\left(1 - \sum_{i=1}^N\left(1 - e^{-\kappa_i\tau}\right)X_t^i\right). \end{aligned} \tag{26}$$

Note that Eq.(26) implies that $P(t,t) = 1$ as long as $\boldsymbol{\nu}\mathbf{X}_t = 1$. Finally, for the short rate r_t, we obtain the following expression (compare to Eq.(3)):

$$r_t = \theta_r + \sum_{i=1}^N \kappa_i X_t^i. \tag{27}$$

4 The USLV Model

Until this point, our formalism was identical to that of the CGW model by Carr et al. (2011). Now it is time to part ways.

CGW investigate a parametric stochastic volatility (SV) specification of the dynamics of Z_t. In that specification, the stock volatility is obtained by a stochastic time change with an activity rate process given by a superposition of CIR processes.

Our plan is different. We want to stick to simple one- or two-factor specifications of the stochastic volatility process, while concentrating on modeling a nonparametric local volatility layer in Eq.(19) in such a way that all observable option quotes would be exactly matched. Furthermore, our approach is necessarily numerical and is based on a Markov chain approximation to the dynamics of the martingale Z_t.

We will construct our model in two steps. In the first step, we develop a discretized nonparametric local volatility version of USLV that corresponds to a zero vol-of-vol limit of the full-blown model. Calibration to option quotes in this framework is achieved via a set of multiplicative adjustment factors acting on elements of the Markov generator (see below). We will refer to these adjustment factors as the one-dimensional (1D) *Speed Factors* (SFs). Calibration of such a one-dimensional USLV model amounts to computing a set of 1D SFs.

In the second step, we move on to a full-blown USLV model with a nonzero vol-of-vol by turning stochastic volatility on. In the present discrete-space framework, this amounts to making the Markov chain generator stochastic.

To retain a near-perfect calibration to a set of option quotes obtained in the first (1D) step, we introduce another set of speed factors which we will refer to as 2D speed factors (2D SFs). The 2D SFs are then calibrated from the previously computed 1D SFs using a version of the Markovian projection method implemented in an efficient manner using forward induction on the Markov chain. Once calibrated, the resulting 2D Markov chain can be utilized to set up efficient pricing schemes for derivatives based on backward-induction algorithms.

We note that the resulting discrete-space continuous-time dynamics on a Markov chain with a stochastic generator arising in our approach resembles the BSLP model developed for portfolio credit derivatives in Arnsdorf & Halperin (2007). The two models are similar in that both use a two-step approach to calibration. In addition, both the definition and parametrization of our speed factors are similar to how analogously defined contagion factors are introduced and used in the BSLP model. The *difference* of the USLV model from the BSLP model is that while a discrete-space description is *exact* for credit,[10] it is used as an *approximation* to the dynamics of the underlying for USLV. Furthermore, while the BSLP model uses a non-linear *death* process as a model for a portfolio loss, in the present case we use a nonlinear *quasi-birth-death* (QBD) process as a discrete approximation to the dynamics of a two-dimensional Markov driver $\mathbf{Z_t} = \left(Z_t^{(1)}, Z_t^{(2)}\right)^\top$. Finally, we use a different method for an efficient computation of matrix exponentials arising in evaluation probabilities on Markov chains.

In what follows, we use the following compact notation for different flavors of our model. We denote one- and two-factor discretized local volatility models as USLV(1,0) and USLV(2,0), respectively. Versions with stochastic volatility are denoted as USLV(1,1), (2,1), or (2,2). We call a discretized process for $\mathbf{Z_t} = \left(Z_t^{(1)}, Z_t^{(2)}\right)^\top$ a 1D process, while the joint process of $\mathbf{Z_t}$ and stochastic volatility is referred to as a 2D process.

4.1 USLV(1,0): One-Factor Local Volatility

We consider the following SDE describing a local volatility dynamics of a one-dimensional Markov driver Z_t under the $\mathbb{P}$-measure:

$$\frac{dZ_t}{Z_t} = \hat{\sigma}(Z_t, t) dW_t, \tag{28}$$

where W_t is a Brownian motion and $\hat{\sigma}(Z_t, t)$ is a local volatility. Our objective is to discretize the dynamics corresponding to Eq.(28).

To this end, we first construct a nonuniform grid of possible values of the martingale $Z_t > 0$ with $Z_0 = 1$. Let p be the number of points on the grid and $0 < z_0 < z_1 < \cdots < z_{p-1}$ be the nodes on the grid. Irregularity of the grid allows making it denser in interesting regions, and sparser in uninteresting ones. Clearly, the fact that our process is a martingale helps as our grid should not be too large: as time passes, the underlying stays around the current value in the sense of expectations.

Let the current state be z_i and let $\Delta z_{i,i-1} = z_i - z_{i-1}$, $i \in [1, p-1]$ be the ith space interval on the grid. We construct the elements of the generator matrix A_t following the adaptive Markov chain approximation of Cerrato et al. (2011). Adapting their formulae to

[10] Provided some additional assumptions are made, such as a discrete spectrum of recovery values.

our case of a zero-drift diffusion Z_t, we obtain the Markov chain generator

$$A = \begin{pmatrix} a_{00} & a_{01} & 0 & \cdots & 0 \\ a_{10} & a_{11} & a_{12} & \cdots & 0 \\ \vdots & & & & \\ 0 & \cdots & a_{p-2,p-3} & a_{p-2,p-2} & a_{p-2,p-1} \\ 0 & \cdots & 0 & a_{p-1,p-2} & a_{p-1,p-1} \end{pmatrix} \tag{29}$$

with elements

$$\begin{aligned} a_{i,i-1} &= \frac{s_i(t)}{\Delta z_{i,i-1}\left(\Delta z_{i,i-1} + \Delta z_{i+1,i}\right)}, \quad a_{i,i+1} = \frac{s_i(t)}{\Delta z_{i+1,i}\left(\Delta z_{i,i-1} + \Delta z_{i+1,i}\right)}, \\ a_{ii} &= -a_{i,i-1} - a_{i,i+1}, \quad a_{i+j,i} = a_{i,i+j} = 0, \qquad j > 1. \end{aligned} \tag{30}$$

where we defined a grid-valued set of *speed factors* (SFs)

$$s_i(t) = \hat{\sigma}^2(z_i, t). \tag{31}$$

Now we have a Markov generator parametrized by the pointwise set of speed factors in Eq.(31). Next we will show how the SFs in Eq.(31) are turned into tunable parameters and used for calibration to option quotes.

4.2 Parametrization of Speed Factors in USLV(1,0)

As it stands, the parametrization in Eq.(29) is very general. The number of free parameters per a given time slice t is p, typically far exceeding the number of observed option quotes available for calibration for cases of practical interest.[11]

To achieve an exact match between the number of free parameters in our model and the number of available option quotes, we consider the following parametrization of our 1D speed factors Eq.(31). Our approach here is similar to how contagion factors are used in the BSLP model of Arnsdorf & Halperin (2007).

We assume that as a function of time t, the SFs $s_i(t)$ are piecewise constant between maturities of traded options. This considerably simplifies computation of matrix exponentials.

For the dependence of $s_i(t)$ on the grid index i (i.e., for the z *dependence*), we proceed as follows. Let $K_0, K_1, \ldots, K_{q-1}$ be a set of strikes for traded options across all maturities, expressed in terms of the Z-space as, e.g., in Proposition 3 of Carr et al. (2011). We assume that all these strikes correspond to q different nodes on our grid.[12] We then model the speed factors s_i for all values of i by picking exactly q free values $\hat{s}_0, \ldots, \hat{s}_{q-1}$ at locations $K_0, K_1, \ldots, K_{q-1}$, and using linear interpolation for points in between. In other words, our speed factors $s_i(t)$ are piecewise-linear in z_i, while the anchor points at q selected nodes serve

[11] Typical values of p that we have in mind for practical implementation are around 20 to 100; see also Cerrato et al. (2011).

[12] This is because our grid is constructed in such a way that it puts all strikes exactly at some nodes, plus add some nodes in between and beyond the range of quoted strikes.

as free parameters for calibration to option quotes. Any consistent set of option quotes can be exactly matched by the present method.[13]

Note that calibration of local volatility models without assuming availability of a complete set of option quotes (i.e., when the number of option quotes is less then the number of nodes on a grid) has been previously discussed in the literature. In particular, a recent paper by Lipton & Sepp (2011) analyzes a setting where the local volatility function is piecewise-flat between quoted strikes (or between mid-points) using Laplace-transform based methods. Unlike their method, which is exact in 1D, our approach is based on numerical optimization, but it is extendable to a multivariate setting (2D and higher) along the same lines as in 1D. In addition, a piecewise-linear volatility model appears to be a less drastic approximation than a piecewise-flat model.

4.3 Calibration of the USLV(1,0) Model

Calibration of the speed factors $s_i(t)$ in the above setting is straightforward. The anchor points introduced above serve as parameters of optimization. Given a multidimensional optimizer, at each iteration we first construct the generator matrix given the current set of the anchor points. After that, finite-time probability distributions are computed by taking matrix exponentials of the generator. As the mathematical structure of our model is essentially the same for the $N = 1$ and $N = 2$ cases (one or two factors for the term structure), we postpone presenting details of this procedure until the next section where we introduce an $N = 2$ version of our model. Theoretical option prices for a given set of model parameters are then computed using these probability distributions. Finally the optimizer adjusts the current set of free parameters to decrease the error between the model and the market.

5 USLV(2,0): Two Curve Factors

With two factors for the curve ($N = 2$), we assume the following vector-valued SDE for the dynamics of $\mathbf{Z}_t = (Z_t^1, Z_t^2)^\top$:

$$\begin{pmatrix} dZ_t^{(1)} \\ dZ_t^{(2)} \end{pmatrix} = \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix} \begin{pmatrix} dW_t^{(1)} \\ dW_t^{(2)} \end{pmatrix}, \tag{32}$$

where two Brownian motions $W_t^{(1)}, W_t^{(2)}$ are independent, and the volatility matrix $\Sigma = \Sigma(\mathbf{z})$ is defined as follows:

$$\Sigma(\mathbf{z}) = \sqrt{s(\mathbf{z},t)} \begin{pmatrix} \sigma_1\sqrt{1-\rho^2} & \sigma_1\rho \\ 0 & \sigma_2 \end{pmatrix}, \quad \Sigma(\mathbf{z})\Sigma(\mathbf{z})^T = s(\mathbf{z},t) \begin{pmatrix} \sigma_1^2 & \rho\sigma_1\sigma_2 \\ \rho\sigma_1\sigma_2 & \sigma_2^2 \end{pmatrix}. \tag{33}$$

[13] Alternatively, if the number of liquid option quotes per maturity is large while the calibration speed is an issue, then some strikes can be omitted from the set of anchor points at the price of giving up an exact calibration for these omitted strikes, while keeping an exact calibration for the other strikes. For example, one can be guided by the size of quoted bid-ask spreads in deciding which strikes could be skipped without much sacrifice to accuracy while gaining in performance. To preserve no-arbitrage for the omitted strikes, one should use a monotonic interpolation in the probability space.

Here $s(\mathbf{z},t) = s(\mathbf{Z}_t, t|\mathbf{Z}_t = \mathbf{z}) \geq 0$ is a scalar function of the state variable $\mathbf{Z}_t = \left(Z_t^{(1)}, Z_t^{(2)}\right)^\top$, and $\mathbf{z} = (z_1, z_2)$ are the values of $\mathbf{Z}_t$ at time t. An explicit specification of this function will be given below. Note that components $Z_t^{(1)}$ and $Z_t^{(2)}$ defined by Eq.(32) and Eq.(33) are correlated with correlation coefficient ρ.

Our first objective is to approximate the dynamics given by Eq.(32) by a 2D Markov chain. To this end, we start with a Markov generator corresponding to the 2D diffusion given by Eq.(32):

$$\mathcal{L}V(\mathbf{z}) = \frac{1}{2}\sum_{i,j=1}^{N=2} \left[\Sigma\Sigma^T\right]_{ij} \frac{\partial^2 V(\mathbf{z})}{\partial z_i \partial z_j}, \tag{34}$$

where $V(\mathbf{z}) = V(\mathbf{z}_T|\mathbf{z})$ is an arbitrary function (a value function or a transition density) of backward variables $\mathbf{z}$ (with forward variables $\mathbf{z}_T$ treated as parameters). The generator specifies the continuous-space backward equation

$$\frac{\partial V(\mathbf{z},t)}{\partial t} = -\mathcal{L}V(\mathbf{z},t).$$

Note that in order to be probabilistically interpretable as a generator of a Markov chain, a discrete version A of the generator $\mathcal{L}$ should have all nondiagonal elements nonnegative, and all diagonal elements negative, such that all rows sum up to one. These remarks are important as not any discretization of $\mathcal{L}$ gives rise to a valid Markov chain generator. For example, using a central divided difference to approximate the mixed derivatives in Eq.(34) would not preserve nonnegativity of nondiagonal elements of A.

With these remarks in mind, and given a two-dimensional grid[14] of values of $\left(Z^{(1)}, Z^{(2)}\right)$ with p nodes per dimension, we approximate second derivatives by divided differences. Derivatives V_{z_k,z_k}, $k = 1,2$, are represented using the central differences

$$\begin{aligned}
\left.\frac{\partial^2 V}{\partial z_1^2}\right|_{ij} &= \frac{V_{i+1,j} - 2V_{i,j} + V_{i-1,j}}{(\Delta z_1)^2} + O\left((\Delta z_1)^2\right), \\
\left.\frac{\partial^2 V}{\partial z_2^2}\right|_{ij} &= \frac{V_{i,j+1} - 2V_{i,j} + V_{i,j-1}}{(\Delta z_2)^2} + O\left((\Delta z_2)^2\right),
\end{aligned}$$

while for the mixed derivative we take uncentered differences which preserve nonnegativity:

$$\begin{aligned}
\left.\frac{\partial^2 V}{\partial z_1 \partial z_2}\right|_{ij} &= \frac{V_{i+1,j+1} - V_{i+1,j} - (V_{i,j+1} - 2V_{ij} + V_{i,j-1}) - (V_{i-1,j} - V_{i-1,j-1})}{2\Delta z_1 \Delta z_2} \\
&\quad + O(\Delta z_1 \Delta z_2) + O\left((\Delta z_1)^2\right) + O\left((\Delta z_2)^2\right), \quad \rho \geq 0, \\
\left.\frac{\partial^2 V}{\partial z_1 \partial z_2}\right|_{ij} &= \frac{V_{i+1,j} - V_{i+1,j-1} + (V_{i,j+1} - 2V_{ij} + V_{i,j-1}) - (V_{i-1,j+1} - V_{i-1,j})}{2\Delta z_1 \Delta z_2} \\
&\quad + O(\Delta z_1 \Delta z_2) + O\left((\Delta z_1)^2\right) + O\left((\Delta z_2)^2\right), \quad \rho < 0.
\end{aligned}$$

[14] For simplicity, in this section we assume that our one-dimensional grids are uniform with the same number p of grid points per each dimension. Therefore, $z_{i+1,j} - z_{i,j} = \delta z_1,\ \forall j = 1, N,\ i \in [1,p)$, $z_{i,j+1} - z_{i,j} = \delta z_2,\ \forall i = 1, N,\ j \in [1,p)$. For analysis of a nonuniform grid, see Appendix A.

Using this in Eq.(34) and regrouping terms, we obtain

$$(\mathcal{L}V(z))_{ij} = \sum_{k,m\in\{-1,0,1\}} a_{ij|i+k,j+m}, V_{i+k,j+m} \tag{35}$$

where we introduced the following compact notation:

$$\begin{aligned}
a_{ij|i+1,j} &= a_{ij|i-1,j} = \frac{1}{2}s_{ij}(t)\left(\frac{\sigma_1^2}{(\Delta z_1)^2} - \frac{|\rho|\sigma_1\sigma_2}{\Delta z_1\Delta z_2}\right),\\
a_{ij|i,j+1} &= a_{ij|i,j-1} = \frac{1}{2}s_{ij}(t)\left(\frac{\sigma_2^2}{(\Delta z_2)^2} - \frac{|\rho|\sigma_1\sigma_2}{\Delta z_1\Delta z_2}\right),\\
a_{ij|ij} &= -s_{ij}(t)\left(\frac{\sigma_1^2}{(\Delta z_1)^2} - \frac{|\rho|\sigma_1\sigma_2}{\Delta z_1\Delta z_2} + \frac{\sigma_2^2}{(\Delta z_2)^2}\right),\\
a_{ij|i+1,j+1} &= a_{ij|i-1,j-1} = \begin{cases} \frac{s_{ij}(t)}{2}\frac{\rho\sigma_1\sigma_2}{\Delta z_1\Delta z_2} & \text{if } \rho \geq 0,\\ 0 & \text{if } \rho < 0,\end{cases}\\
a_{ij|i+1,j-1} &= a_{ij|i-1,j+1} = \begin{cases} 0 & \text{if } \rho \geq 0,\\ \frac{s_{ij}(t)}{2}\frac{|\rho|\sigma_1\sigma_2}{\Delta z_1\Delta z_2} & \text{if } \rho < 0,\end{cases}
\end{aligned} \tag{36}$$

where $s_{ij}(t) = [s(Z_t,t)]_{ij}$.

To ensure that all parameters $a_{ij|i+k,j+m},\ k,m \neq 0$ are nonnegative, we impose the following constraint on the step size Δz_2 given a chosen step Δz_1:

$$|\rho|\frac{\sigma_2}{\sigma_1}\Delta z_1 \leq \Delta z_2 \leq \frac{\sigma_2}{|\rho|\sigma_1}\Delta z_1. \tag{37}$$

Assuming Eq.(37) is satisfied, we can interpret Eq.(35) as a generator matrix of a 2D Markov chain. We can write it in a matrix form as follows:

$$(\mathcal{L}V(z))_{ij} = [AV]_{ij} = \sum_{i',j'} A_{ij|i'j'}V_{i'j'}. \tag{38}$$

As we deal with a two-factor setting, the matrix elements of the generator A carry four indices rather than two. To sum over two indices corresponding to the $Z^{(1)}$- and $Z^{(2)}$-states in Eq.(38), it is convenient to group all transitions according to the change of one variable (e.g., $Z^{(1)}$). We obtain

$$\begin{aligned}
(\mathcal{L}V(z))_{ij} &= \sum_{i',j'} A_{ij|i'j'}V_{i'j'} = \sum_{j'} A_{ij|i-1,j'}V_{i-1,j'} + \sum_{j'} A_{ij|ij'}V_{ij'} + \sum_{j'} A_{ij|i+1,j'}V_{i+1,j'}\\
&= \big[\big\{a_{ij|i-1,j-1}V_{i-1,j-1} + a_{ij|i-1,j}V_{i-1,j} + a_{ij|i-1,j+1}V_{i-1,j+1}\big\}\\
&\quad + \big\{a_{ij|i,j-1}V_{i,j-1} + a_{ij|ij}V_{ij} + a_{ij|i,j+1}V_{i,j+1}\big\}\\
&\quad + \big\{a_{ij|i+1,j-1}V_{i+1,j-1} + a_{ij|i+1,j}V_{i+1,j} + a_{ij|i+1,j+1}V_{i+1,j+1}\big\}\big].
\end{aligned} \tag{39}$$

Here terms in the first, second, and third row correspond to transitions $i \to i-1$, $i \to i$, and $i \to i+1$ in the $Z^{(1)}$-dimension, respectively.

Mathematically, this is expressed via the following tridiagonal block-matrix form for the resulting "one-dimensional" Markov chain generator A:

$$A = \left(\begin{array}{ccccccc} L^{(0)} & F^{(0)} & 0 & 0 & \cdots & 0 & 0 \\ B^{(1)} & L^{(1)} & F^{(1)} & 0 & \cdots & 0 & 0 \\ 0 & B^{(2)} & L^{(2)} & F^{(2)} & \cdots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \cdots & B^{(p-1)} & L^{(p-1)} \end{array} \right), \tag{40}$$

where all matrices $B^{(i)}, L^{(i)}$, $F^{(i)}$ have dimension $p \times p$, i.e., the dimension of our one-dimensional grids.[15] Explicit expressions for these matrices can be found from Eq.(39):

$$\begin{array}{l} B^{(i)}_{j,j-1} = a_{ij|i-1,j-1}, \; B^{(i)}_{j,j} = a_{ij|i-1,j}, \; B^{(i)}_{j,j+1} = a_{ij|i-1,j+1}, \\ L^{(i)}_{j,j-1} = a_{ij|i,j-1}, \; L^{(i)}_{j,j} = a_{ij|ij}, \; L^{(i)}_{j,j+1} = a_{ij|i,j+1,}, \\ F^{(i)}_{j,j-1} = a_{ij|i+1,j-1}, \; F^{(i)}_{j,j} = a_{ij|i+1,j}, \; F^{(i)}_{j,j+1} = a_{ij|i+1,j+1}, \end{array} \tag{41}$$

while all other elements of these matrices vanish. Note that this implies that the generator Eq.(40) is "doubly" sparse, as matrices $B^{(i)}, L^{(i)}$ and $F^{(i)}$ are themselves sparse; see also a comment at the end of this section.

The block-tridiagonal matrix structure Eq.(40) of the Markov chain generator A is characteristic of so-called quasi-birth-death (QBD) processes. A QBD process is a bivariate Markov chain of a special type of dynamics of two components. The first component, $Z_t^{(1)}$, called the "level," follows a birth-and-death (BD) process on either a finite or infinite set of states. Conditional on the realization of the $Z_t^{(1)}$-component at a given step $[t, t + dt]$, the second component $Z_t^{(2)}$, called the "phase," follows another Markov process. For a short review of QBD processes, see, e.g., Kharoufeh (2011). Note that in our particular case, $Z_t^{(2)}$ follows another BD process, while the support of $Z_t^{(1)}$ is finite. QBD processes with finite support are called finite QBD processes.

The symbols L, B and F in Eq.(40) stand for local (without change of level), backward and forward (the level is changed by one unit up or down) moves, respectively. Note that as long as the matrices $L^{(i)}, B^{(i)}$ and $F^{(i)}$ depend on level i via the discretized local volatility function s_{ij}, our QBD process with generator Eq.(40) is a *level-dependent* (or *nonlinear*) finite QBD, denoted sometimes as a (finite) LDQBD in the literature.

It is easy to check that Eq.(40) with elements defined as in Eq.(36) is a valid Markov generator where all off-diagonal elements are positive, diagonal elements are negative and each row in A sums to zero.

After the QBD generator matrix is constructed according to Eq.(40), a matrix P of finite-time transition probabilities with matrix elements

$$P_{ij|i'j'}(t, T) \equiv P \left[(Z_T^{(1)}, Z_T^{(2)} = z_{i'}^{(1)}, z_{j'}^{(2)} | Z_t^{(1)}, Z_t^{(2)} = z_i^{(1)}, z_j^{(2)} \right] \tag{42}$$

[15] If grids in $z^{(1)}$ and $z^{(2)}$ have different lengths p_1 and p_2, then the size of these matrices will be $p_2 \times p_2$.

can be computed by solving the forward Kolmogorov equation

$$\frac{\partial P}{\partial T} = PA. \tag{43}$$

For a given interval $[t_1, t_2]$ where the generator A does not depend on time, the solution of the forward equation is

$$P = P_{t_1} e^{(t_2 - t_1)A},$$

where P_{t_1} is a state vector at time t_1, and e^X stands for the matrix exponential of X.

A few remarks on the complexity of the method just presented are in order here. We have managed to map the two-factor continuous-space dynamics Eq.(32) on the state space $Z_1 \times Z_2$ onto a QBD process with generator Eq.(40). The latter can formally be viewed as a one-dimensional Markov chain in an extended linear space whose basis is formed by elements of a Kroneker product of grid values $\mathbf{Z}_g^{(1)} \otimes \mathbf{Z}_g^{(2)}$ (and properly rearranged to form a QBD structure). Therefore, computation of transition probabilities Eq.(43) in our *two-factor* model is, at least formally, as simple as a corresponding calculation for a *one-factor* model, and reduces to computation of a single matrix exponential, albeit of a larger matrix.[16]

While naively the generator A has $O(p^4)$ free parameters, their actual number is much lower due to sparsity of the matrix. It is simple to find that the number of nonzero elements that need to be stored scales as $(3p-2)p + 2(2p-1)^2$. For example, for $p = 100$ our matrix A would be of size 10000×10000 with only 109,002 non-zero elements. Matrices of such sizes can well be handled by modern matrix exponentiation methods (see below).[17]

5.1 Transient Probabilities of QBD Chain by Randomization

It is well known that a direct computation of a matrix exponential e^{tA} with a Markov generator A via a straightforward use of a Taylor series expansion as $\sum_{n=0}^{\infty} (tA)^n/n!$ is in general not a good idea (see Moler & van Loan (2003)). The main reason for this is that severe roundoff errors might accumulate (especially when the matrix is large) due to the fact that the generator has both positive and negative entries. In addition, matrices $(tA)^n$ become nonsparse even if the original matrix A is sparse, as is the case for the QBD process.

An efficient method of choice for dealing with matrix exponentials for large matrices is known as *Jensen's randomization*; see, e.g., Gross & Miller (1984), or Haverkort (2001) for a more recent review. The method proceeds as follows. We start with choosing a parameter $\lambda \geq \max_i \{|A_{ii}|\}$, and define the matrix

$$\mathbf{P} = \mathbf{I} + \frac{\mathbf{A}}{\lambda} \;\Rightarrow \mathbf{A} = \lambda \left(\mathbf{P} - \mathbf{I}\right). \tag{44}$$

[16] Here in addition to various efficient algorithms for computing a matrix exponential of a sparse matrix, one could use splitting in different dimensions that would take into account a block-diagonal form of the generator matrix A.

[17] While a randomization method that we describe in the following section may not be the most efficient method when the L_2 norm of matrix A is large (Sidje & Stewart (1999)), it is very convenient for introducing stochastic volatility via a stochastic time change. We therefore stick to this approach in what follows.

With our choice of λ, all entries of $\mathbf{P}$ are between 0 and 1, and all rows sum to 1. This means that $\mathbf{P}$ is a stochastic matrix that describes a *discrete-time* Markov chain "related" to the original *continuous-time* Markov chain with generator A. We now want to discuss in more detail the sense in which these two Markov chains are "related."

To this end, we substitute A as given by Eq.(44) into the solution of the forward equation:

$$P(t) = P_0' e^{tA} = P_0' e^{\lambda t(\mathbf{P}-\mathbf{I})} = P_0' e^{-\lambda t} e^{\lambda \mathbf{P} t}.$$

Using a Taylor series expansion for the matrix exponential in this expression, we obtain

$$P(t) = P_0' e^{-\lambda t} \sum_{n=0}^{\infty} \frac{(\lambda t)^n \mathbf{P}^n}{n!} = P_0' \sum_{n=0}^{\infty} \psi(\lambda t, n) \mathbf{P}^n, \tag{45}$$

where

$$\psi(\lambda t, n) = e^{-\lambda t} \frac{(\lambda t)^n}{n!}, \quad n \in \mathbb{N},$$

are Poisson probabilities, i.e., probabilities of observing n events by time t for a Poisson process with intensity λ. Note that a naive Taylor expansion of the matrix exponential e^{tA} behaves badly, but the new expansion is much better behaved: roundoff errors are now largely eliminated as all entries of matrix $\mathbf{P}$ are between 0 and 1. Moreover, different terms are weighted by the Poisson probabilities, so that the expansion is expected to converge fast when the product λt is not too large.

We note that the construction given by Eq.(45) can be interpreted as a discrete-time Markov chain (DTMC) Y_n ($n = 0, 1, \ldots, p$) with transition matrix $\mathbf{P}$ subordinated to a Poisson process N_t where the latter serves as a randomized "operational time" for Y_n, so that the subordinated process is now defined as $X_t = Y_{N_t}$ (see, e.g., Feller (1968)). We will return to the topic of subordinated processes in Sect. 7.1.

It is important to point out that the method just presented can be used without a matrix-matrix multiplication (as Eq.(45) would naively suggest). Let $\hat{\pi}_n$ be the probability distribution vector in the DTMC with transition matrix $\mathbf{P}$ after n epochs. This vector can be computed recursively:

$$\begin{aligned} \hat{\pi}_0 &= \mathbf{P}_0, \\ \hat{\pi}_n &= \hat{\pi}_{n-1} \mathbf{P}, \quad n \in \mathbb{N}^+. \end{aligned} \tag{46}$$

Using the vector $\hat{\pi}_n$, the state probabilities Eq.(45) in the original CTMC are computed as follows:

$$P(t) = \sum_{n=0}^{\infty} \psi(\lambda t, n) \left(P_0' \mathbf{P}^n \right) = \sum_{n=0}^{\infty} \psi(\lambda t, n) \hat{\pi}_n.$$

Therefore, computationally the algorithm amounts to a series of vector-matrix multiplications that can be done very efficiently for matrices of sizes typical for our problem. Moreover, the recursive procedure of Eq.(46) preserves the sparsity of $\mathbf{P}$, thus enabling a substantial acceleration of the vector-matrix multiplication.

In practice, the infinite sum in Eq.(45) is truncated at some value n_{max}. This value can be adaptively controlled within the algorithm itself, as a theoretical upper bound for an error resulting from the truncation is available as discussed, e.g., by Gross and Miller (1984).

5.2 Calibration of Speed Factors in USLV(2,0)

Summarizing our results for the USLV(2,0) model so far, we see that the mathematical structure of the (2,0) model is similar to that of the (1,0) model. Indeed, for the latter our Markov chain construction gives rise to a *nonlinear birth-death* (BD) process modulated by 1D speed factors (SFs) $s_i(t)$. After a proper parametrization as described in Sect. 4.2, SFs are calibrated to market option quotes. For the (2,0) case (two factors for the term structure), the resulting Markov chain is a *nonlinear quasi-birth-death* process, again modulated by a set of SFs s_{ij}. In terms of computational complexity, the two cases are essentially the same, as both amount to calculation of matrix exponentials of a Markov chain generator (albeit in different dimensions).

Calibration of the (2,0) model is done by the fitting function $s(z, t)$ in Eq.(32). To this end, we proceed similarly to the one-factor case. We assume that function $s(z, t) = s(z_1, z_2, t)$ is a function of single argument, $s(z_1, z_2, t) = s(\alpha_1 z_1 + \alpha_2 z_2, t)$, where $\alpha_1, \alpha_2 \geq 0$ are some weights (e.g., $\alpha_1 = \alpha_2 = 0.5$). For calibration purposes, we could parametrize this function in a piecewise-linear way, i.e., in exactly the same way as we did before for the $N = 1$ model. The number of free parameters (anchor points) and their locations would be chosen based on a particular set of instruments available for calibration.

To continue with our theoretical construction of the model, in what follows we assume that the stage of construction of a (2,0) (or (1,0)) version of the USLV model is completed along the lines described here. In what follows, we refer to these SFs as 1D SFs, in order to differentiate them from another set of speed factors (2D SFs) that will be introduced below when we add stochastic volatility to the model.

Finally, we note that while the main purpose of the USLV(2,0) model for our purposes is to use it as a building block in the construction of a full-blown USLV(2,2) model with stochastic volatility, the pure local volatility USLV(2,0) model can also be useful in its own right, e.g., as a way of pricing European vanilla options with illiquid strikes in terms of prices of liquidly traded options.

6 USLV(2,2): Two Curves and Two Volatility Factors

Once we have a calibrated USLV(2,0) model, introduction of stochastic volatility in this framework amounts to two things: (i) introducing new dynamics for volatility drivers, and (ii) making sure the model still calibrates to available option prices. This produces a calibrated USLV(2,2) model.

Let us note that stochastic volatility dynamics can be introduced in our framework in two ways. In the first approach, we follow the formulation of a continuous-time Markov chain (CTMC) dynamics, which we now augment by 2D dynamics of “spot” variance factors. For numerical implementation, the model is then put on a time grid $[t_0, t_1, \ldots, t_n]$ with a uniform time step Δt. All calculations (see below) are done to $O(\Delta t^2)$ accuracy, which assumes that Δt should be sufficiently small.[18] With this method, we solve the forward and backward

[18] For example, we might need to use daily or more frequent steps, depending on the level of volatility,

Kolmogorov equations one step Δt at a time, similarly to finite difference methods.

In the second approach, we deal with arbitrary time lines which do not necessarily have small time steps. For example, we may want to model the values of underlying factors only on a sparse set of "interesting" dates (e.g., coupon dates, call dates etc.). Essentially, by taking matrix exponentials of the generator, we aim here to achieve a functionality similar to the USLV(1,0) and USLV(2,0) models (or any CTMC model, for that matter), which are capable of computing finite-time transition probabilities directly.[19]

Respectively, in what follows we present two versions of the USLV(2,2) model. In the first version, we assume a Markov dynamics in the pair $(\mathbf{Z}_t, \mathbf{Y}_t)$ where $\mathbf{Y}_t = \left(Y_t^{(1)}, Y_t^{(2)}\right)$ is a bivariate "spot" variance driver. In the second version, we instead assume a Markov dynamics in a pair of $\mathbf{Z}_t$ and an *integrated* bivariate variance, or, more generally, a bivariate stochastic time subordinator (see below). We will use the notation $\mathbf{T}_t = \left(T_t^{(1)}, T_t^{(2)}\right)$ for the latter in what follows. For reasons that will become clear shortly, we will refer to these two versions of the model as the *activity-rate*-based model (AR-USLV), or the *implied-time-change*-based model (ITC-USLV), respectively.

On the *theoretical* side, it turns out that both approaches can be viewed in a unified way by interpreting them as particular realizations of a stochastic time change of the original QBD Markov chain. We will give more details on this below in Sect. 7.1.

On the *practical* side, we can choose between two numerical methods. With the first method, we can implement both the AR- and ITC-versions of our model in a similar way using a version of the Markovian projection method. The latter reduces calibration of USLV(2,2) to a fast forward induction method in what is essentially a 1D problem, without a need for a forward induction on a full 2D Markov chain.[20] No new optimization in addition to one performed at the stage of calibration of the USLV(2,0) model is involved here. Therefore, the method is very fast on each given time step, the only potential bottleneck being the necessity to perform such computation repeatedly on a dense time grid. The method is *nonparametric* in that it solves the problem of calibration of the full-blown USLV(2,2) model via a judicious choice of 2D speed factors (SFs) that are computed off the calibrated 1D SFs of the USLV(2,0) model.

The second method, which is applied below for the ITC-USLV version of our model but in principle could be used for both versions, is to "break the symmetry," and make the process $\mathbf{T}_t = \left(T_t^{(1)}, T_t^{(2)}\right)$ *parametric* in one dimension (e.g., $T^{(2)}$), while keeping it *nonparametric* in another dimension (resp., $T^{(1)}$). The idea here is that for the purpose of calculation of finite-

with this approach.

[19] To the extent that one-step methods, such as the Runge-Kutta method, can be viewed as particular ways to compute matrix exponentials (see Moler & van Loan (2003)), what we mean here by "direct" calculations are other methods of computing matrix exponentials that might in some cases be more efficient than one-step methods.

[20] Recall that by 1D and 2D, we mean linearized spaces obtained from the pairs $\left(Z_t^{(1)}, Z_t^{(2)}\right)$ and $\left(Y_t^{(1)}, Y_t^{(2)}\right)$ by taking elements of pairwise Kroneker products as new 1D bases. In terms of factor counting, our 1D and 2D Markov chains correspond to the two- and four-factor model specifications, respectively.

time transition probabilities in the $\mathbf{Z}$-space, we can perform averaging over the randomness due to $T_t^{(2)}$ *analytically* (or semi-analytically) once a tractable model for subordinator $T_t^{(2)}$ is specified. The averaging over the residual randomness due to $T_t^{(1)}$ is performed numerically. Similarly to the previous case, this calculation can be done in a nonparametric setting, where at each step on our sparse time grid, we introduce just enough free parameters to match observed quotes for options maturing at this time. *Differently* from the previous case, the recalibration to option quotes in the present setting amounts to a (convex) optimization problem in the dimension equal to the number of option quotes for this maturity.

The two flavors (AR and ITC) of our USLV(2,2) model outlined above thus offer a certain trade-off in terms of complexity. For the AR-USLV(2,2) model, the recalibration is fast for one step, but complexity scales linearly with the number N_d of time steps on a dense grid; i.e., the complexity is $O(N_d)$. For the ITC-USLV(2,2) model, the complexity is $O(N_s N_c)$, where N_s is the number of nodes on a sparse time line, and N_c is the number of option quotes per node, independently of Δt, but the $O(N_c)$ part above involves convex optimization in dimension N_c. Based on previous experience with similar models, we expect a compatible performance from the two versions of the USLV(2,2) model, at least for typical cases (e.g., $N_c = 5$, $N_t = 40$). Therefore, in what follows we will present both versions of the model.

Our plan for the reminder of this paper is as follows. In the rest of this section, we describe the AR-USLV(2,2) version of the model, where the Markov pair is $(\mathbf{Z}_t, \mathbf{Y}_t)$, with a bivariate spot variance driver $\mathbf{Y}_t = \left(Y_t^{(1)}, Y_t^{(2)}\right)$. In Sect. 6.1, we provide a qualitative overview of this version of the model. The following subsections of Sect. 6 provide details of our approach. The ITC-USLV(2,2) version of the model, where the Markov pair is $(\mathbf{Z}_t, \mathbf{T}_t)$ with $\mathbf{T}_t = \left(T_t^{(1)}, T_t^{(2)}\right)$ being a bivariate subordinator, is presented in Sect. 7. As will be shown below, calibration to observed option prices amounts, in this approach, to a construction of an *implied time change* (ITC) process. Within a particular approach presented in Sect. 7, the ITC process is defined in terms of a bivariate exponential-Lévy process $\mathbf{L}_t = (T_t, \theta_t)$ where T_t is a *parametric* subordinator (e.g., an exponential gamma process), and θ_t is a *nonparametric* subordinator. The latter will be referred to as a *time dilaton* process, for reasons explained below.

6.1 Overview of AR-USLV(2,2)

As was mentioned above, a model obtained from our USLV(2,0) model by adding new state variables (in this case, spot volatility drivers $\mathbf{Y}_t = \left(Y_t^{(1)}, Y_t^{(2)}\right)$) would not in general match observed option prices, even if our initial USLV(2,0) model does. Moreover, for any particular parametric model for the dynamics of the pair $\left(Y_t^{(1)}, Y_t^{(2)}\right)$, we are still not guaranteed that the full model could accurately fit available option quotes even after calibration of parameters of the Y-process.

In order to reinforce a nearly exact calibration to options for all consistent sets of quotes, we introduce 2D speed factors (SFs) $S(z, y, t)$ in the full (2,2) model, that play an analogous role to 1D SFs Eq.(31) in the (2,0) version of the model. We then provide a fast scheme to compute 2D SFs based on solving the forward equation on the Markov chain. Our method is similar to one used by Britten-Jones & Neuberger (2000) (BJN), see also Rossi (2002), for a lattice-based stochastic local equity model in a (1+1) setting. A similar method was used in the BSLP model by Arnsdorf & Halperin (2007) for modeling dynamics of credit portfolios. For a similar method used for equity option pricing, see Ren et al. (2007).

A peculiar feature of a BJN-like forward induction method (to be presented in detail below) is that it tries to adjust the Z-process for *any* Y-process. It does not address the problem of calibration of parameters of the Y-process itself. In certain situations, it might make sense to try to calibrate parameters of the Y-process as accurately as we can *before* adding a local volatility layer (so that our change to a parametric model due to introduction of a local volatility would be a minimal tweak of the model). Alternatively, we could try to fit parameters of the Y-process *after* we introduce the local volatility layers, but *before* we compute the 2D SFs. This might be an attractive option for a practical method of model calibration in our setting. The reason is that if such parametric calibration of the Y-process produces a good but not perfect fit to the data, the role of non-parametric 2D SFs of the $(N = 2, M = 2)$ model would be to perfect quality of calibration at the price of adding some nonparametricity.[21]

Note that while the BJN-like approach does not by itself address the problem of calibration of parameters of the Y-process for a parametric specification of the dynamics of $\mathbf{Y}_t$, this is where we could use Laplace transform based methods for stochastic subordinators, similar to the method presented in Sect. 7 in a slightly different setting. This implies that the calibration method presented later in this section and a method presented in Sect. 7 can in practice be used together for a joint parametric/nonparametric calibration of the AR-USLV(2,2) model.

Yet another possible way to calibrate our model would be as follows. If a trader has a strong view on the relative weights of a spanned (delta-driven) and unspanned (genuine vega) contribution to options' vegas, and wants the model to behave accordingly, this could be achieved as follows. Assuming that we are able to map constraints like those onto some typical behavior of the set of SFs,[22] we first fix some set of 1D SFs, and then calibrate parameters of the Y-process given these SFs. After parameters of the Y-process are specified in this way, we proceed in the regular way of calibrating the model, by first computing the "true" (market-implied) set of 1D SFs, and then following the forward calibration of 2D SFs in the full-blown model with parameters of the Y-process just computed at the previous step. Again, a combination of various methods presented below can be used to implement such a calibration strategy.

As a brief summary, our QBD Markov chain stochastic-local volatility offers substantial flexibility in how the model can be calibrated to available market and/or historical data. Different steps/versions of the calibration procedure can be combined (or skipped), depending on the specific needs of an end user. We now proceed with describing our framework.

[21] We hold a view that nonparametricity is "evil," but it is a "common evil" in the sense that it is used everywhere (for term structure calibration, local volatility models etc.).

[22] Such dependence can be established either theoretically, or empirically on the basis of behavior of the model as a function of model parameters.

6.2 QBD Processes and Stochastic Time Changes

We prefer to think of stochastic volatility in terms of a stochastic time change of some "base" process such as a Brownian motion. (See Sect. 7 for more details and relevant references.) As our original two-factor ($N = 2$) diffusion equation, Eq.(32), has two Brownian drivers, $W_t^{(1)}$ and $W_t^{(2)}$, we can use two *different* stochastic clocks on them. This would amount to having a stochastic local volatility model with $N = 2$ and $M = 2$.

Such formulation can be useful for asset classes where the short- and long-term option volatilities typically behave differently (e.g., have different typical levels or vol-of-vol), in addition to a different behavior of Z_t-factors driving short- and long-term prices of basic instruments (bonds, futures etc.). For example, for modeling commodity derivatives, we might want to have one long-term factor $Z_t^{(1)}$ driven by a Brownian driver $W_t^{(1)}$ with its own stochastic clock (stochastic volatility) driver $Y_t^{(1)}$, and another, short-term factor $Z_t^{(2)}$ driven by another (possibly correlated) Brownian driver $W_t^{(2)}$, with its own stochastic clock driver $Y_t^{(2)}$.[23] This results in a four-factor scenario with correlated long- and short-term factors, each having its own stochastic volatility driver.

The above picture of two curve drivers each having its own stochastic clock is not lost in our discrete-space Markov chain construction. As we will show next, the structure of our QBD Markov chain Eq.(40) for the $N = 2$ case enables introducing two stochastic clocks in the model in an internally consistent way, and without any need of introducing additional *ad hoc* constraints on the model dynamics. These stochastic clocks will modulate two Markov chain generators. As the latter play the role of stochastic drivers in the discrete-space setting, the resulting "ecosystem" of (discrete-valued) curve and volatility factors bears a strong structural similarity to its continuous-space counterpart.

To explain our construction, we start with representing the Markov generator Eq.(40) in the following form:

$$\begin{aligned} A &= \begin{pmatrix} -\hat{F}^{(0)} & F^{(0)} & 0 & \cdots & 0 \\ B^{(1)} & -\hat{F}^{(1)} & F^{(1)} & \cdots & 0 \\ 0 & B^{(2)} & -\hat{F}^{(2)} & \cdots & 0 \\ \vdots & & & & \\ 0 & 0 & \cdots & B^{(p)} & -\hat{F}^{(p)} \end{pmatrix} + \begin{pmatrix} \hat{L}^{(0)} & 0 & 0 & \cdots & 0 \\ 0 & \hat{L}^{(1)} & 0 & \cdots & 0 \\ 0 & 0 & \hat{L}^{(2)} & \cdots & 0 \\ \vdots & & & & \\ 0 & 0 & 0 & \cdots & \hat{L}^{(p)} \end{pmatrix} \\ &\equiv A_1 + A_2, \end{aligned} \tag{47}$$

where $\hat{F}^{(i)} = \text{diag}\left(F^{(i)}\mathbf{1}\right) + \text{diag}\left(B^{(i)}\mathbf{1}\right)$ and $\hat{L}^{(i)} = L^{(i)} + \hat{F}^{(i)}$, with $\mathbf{1}$ being a vector of ones. Using Eq.(36) and Eq.(41), it can be readily checked that both A_1 and A_2 defined in Eq.(47) are valid generators in the sense that for both, all off-diagonal elements are positive, all diagonal elements are negative and all rows sum up to zero.

This can be interpreted as follows. The second generator A_2 corresponds to an idiosyncratic component of the Z-dynamics that is independent of the rest of the system, and can be

[23] Alternatively, correlated dynamics of two stochastic drivers with each one having its own stochastic volatility factor can be used for pricing hybrid derivatives.

thought of as describing idiosyncratic jumps of $Z_t^{(2)}$ that occur without a simultaneous jump of $Z_t^{(1)}$ on the same time interval.[24] In terms of representation of stochastic dynamics of our system, this generator is an avatar of the idiosyncratic Brownian driver $W_t^{(2)}$ in Eq.(32). The first generator A_1 describes joint jumps of $Z_t^{(1)}$ and $Z_t^{(2)}$, and can be thought of as an avatar of the common Brownian driver $W_t^{(1)}$ in Eq.(32).

As shown in Appendix B, a random time change of a continuous-time Markov chain amounts, in terms of the resulting Markov generator for the chain, to scaling all elements of the original Markov chain generator by a common factor given by the value of the activity rate (intensity of the time change) process.

As they conserve probability separately from each other, two generators A_1 and A_2 can be seen as representing two different subsystems of our dynamic system in the $(Z_t^{(1)}, Z_t^{(2)})$-space. As a time change acts as a common scaling factor on a generator matrix (see Appendix B), this implies that we can subject two generators, A_1 and A_2, to *different* stochastic clock changes without any problems with laws of probability: after separate time changes, probabilities are still all nonnegative, and sum up to one in each subsystem separately. This gives rise to a discrete-space version of a continuous $M = 2$ stochastic volatility model.

To summarize, the (2+2)-factor structure of the original continuous-space system Eq.(32) (i.e. two factors for the term structure and two factors for the volatility) is now naturally mapped onto a corresponding structure in our Markov chain model, where a QBD process is an avatar of a two-dimensional Brownian motion $\left(W_t^{1)}, W_t^{(2)}\right)$, and two volatility factors are separately introduced as two stochastic clocks for two generators A_1 and A_2 as explained above. We now discuss specific realizations of this scenario in our model.

6.3 Forward Equation and Transition Probabilities in USLV(2,2)

We assume discrete dynamics of stochastic intensity (stochastic volatility) drivers $\mathbf{Y}_t$, with an odd number $N_y = 2q+1$ of discrete states for each driver $Y_t^{(1)}, Y_t^{(2)}$. Points on a two-dimensional Y-grid are denoted as $\left\{Y_{\alpha_1}^{(1)}, Y_{\alpha_2}^{(2)}\right\}_{\alpha_1,\alpha_2=0}^{\alpha_1,\alpha_2=2q}$. The initial values $(Y_{t=0}^{(1)}, Y_{t=0}^{(2)})$ correspond to the midpoints $(Y_q^{(1)}, Y_q^{(2)})$ on the grid.

In practice, we prefer to keep a low number of states (say 3 to 11) per volatility factor. As volatility is unobservable, we feel that maintaining a low number of states might suffice to reproduce most important stylized facts about stochastic volatility such as mean reversion and/or volatility clustering (persistence), alongside its role in providing a better behavior of a forward smile (a non-flattening smile for longer maturities) than typical behavioral patterns observed with local volatility models.

To ease the notation, in this section we use Latin indices i, j, k to enumerate states ($\mathbf{Z}_t = \mathbf{Z}_i, \mathbf{Z}_{t+dt} = \mathbf{Z}_j$ etc.), and Greek indices α, β to enumerate values of $\mathbf{Y}_t$, $\mathbf{Y}_{t+dt}$. However, because we deal with a two-factor setting, both the indices and factor values are now two-

[24] This can also be viewed as a one-dimensional orthogonal projection of two-dimensional dynamics of the pair $(Z_t^{(1)}, Z_t^{(2)})$ onto a subspace where no jumps in variable $X = Z_t^{(1)}$ are allowed.

component vectors rather than scalars; for example,

$$\mathbf{Z}_i = \left(Z^{(1)}_{i_1}, Z^{(2)}_{i_2} \right) \ , \ \ i = (i_1, i_2) \ , \ i_1, i_2 \in \mathbb{Z}^+. \tag{48}$$

A similar representation is used for volatility drivers $\mathbf{Y}_\alpha = \left(Y^{(1)}_{\alpha_1}, Y^{(2)}_{\alpha_2} \right)$. In what follows we use both the vectorized and component notations.

We postulate that the 2D dynamics of the pair $(\mathbf{Z}_t, \mathbf{Y}_t)$ (where both factors $\mathbf{Z}_t$ and $\mathbf{Y}_t$ are two-dimensional) in the USLV model is Markovian. The system is defined in terms of the joint marginal probabilities

$$\pi(j, \alpha, t) \equiv P \left[\mathbf{Z}_t = \mathbf{Z}_j, \mathbf{Y}_t = \mathbf{Y}_\alpha \right]$$

and conditional transition probabilities

$$p_{i\alpha|j\beta}(t, t+dt) \equiv P \left[\mathbf{Z}_{t+dt} = \mathbf{Z}_j, \mathbf{Y}_{t+dt} = \mathbf{Y}_\beta | \mathbf{Z}_t = \mathbf{Z}_i, \mathbf{Y}_t = \mathbf{Y}_\alpha \right].$$

The forward equation takes the form

$$\pi(j, \beta, t+dt) = \sum_{j,\alpha} p_{i\alpha|j\beta}(t, t+dt) \pi(i, \alpha, t). \tag{49}$$

The transition probabilities have the following expansion:

$$p_{i\alpha|j\beta}(t, t+dt) = \delta_{ij}\delta_{\alpha\beta} + A_{i\alpha|j\beta}(t) dt + O\left(dt^2\right), \tag{50}$$

where $A_{i\alpha|j\beta}(t)$ is the Markov generator, and $\delta_{ij} = \delta_{i_1 j_1}\delta_{i_2 j_2}$ is the 2D Kroneker symbol (with a similar definition for $\delta_{\alpha\beta}$).

To proceed, we introduce the following conditional probabilities:

$$\begin{aligned} P^{(\alpha)}_{ij}(t, t+dt) &= P \left[\mathbf{Z}_{t+dt} = \mathbf{Z}_j | \mathbf{Z}_t = \mathbf{Z}_i, \mathbf{Y}_t = \mathbf{Y}_\alpha \right], \\ \hat{P}^{(ij)}_{\alpha\beta}(t, t+dt) &= P \left[\mathbf{Y}_{t+dt} = \mathbf{Y}_\beta | \mathbf{Y}_t = \mathbf{Y}_\alpha, \mathbf{Z}_t = \mathbf{Z}_i, \mathbf{Z}_{t+dt} = \mathbf{Z}_j \right]. \end{aligned} \tag{51}$$

The joint probability $p_{i\alpha|j\beta}$ can now be written as follows:

$$p_{i\alpha|j\beta}(t, t+dt) = P^{(\alpha)}_{ij}(t, t+dt) \hat{P}^{(ij)}_{\alpha\beta}(t, t+dt). \tag{52}$$

Using Eq.(50), we obtain

$$P^{(\alpha)}_{ij}(t, t+dt) = \sum_\beta p_{i\alpha|j\beta}(t, t+dt) \equiv \delta_{ij} + \hat{A}^{(\alpha)}_{ij}(t) dt + O\left(dt^2\right), \tag{53}$$

where

$$\hat{A}^{(\alpha)}_{ij}(t) = \sum_\beta A_{i\alpha|j\beta}(t) \, , \ \ \sum_j \hat{A}^{(\alpha)}_{ij}(t) = 0 \tag{54}$$

is the conditional generator of the Z-Markov chain.

It is convenient to write the second conditional probability in Eq.(51) in the following form:

$$\begin{aligned}
\hat{P}^{(ii)}_{\alpha\beta}(t,t+dt) &= \delta_{\alpha\beta} + \hat{Q}^{(i)}_{\alpha\beta}(t)dt\,, \quad \sum_{\beta}\hat{Q}^{(i)}_{\alpha\beta}(t) = 0, \qquad (55)\\
\hat{P}^{(i,i+m)}_{\alpha\beta}(t,t+dt) &= \delta_{\alpha\beta} + \tilde{Q}^{(m)}_{\alpha\beta}(t)\,, \quad \sum_{\beta}\tilde{Q}^{(m)}_{\alpha\beta}(t) = 0\,, \quad (m_1,m_2) \neq (0,0).
\end{aligned}$$

Note that the term $\tilde{Q}^{(m)}_{\alpha\beta}(t)$ in the second equation is *not* multiplied by dt because the second relation in Eq.(51) is not a transition probability but rather a conditional probability where we condition, in particular, on $\mathbf{Z}_{t+dt}$. If $d\mathbf{Z}_t \neq 0$, then dt cancels out in the calculation of the conditional probability. This means that $\tilde{Q}^{(m)}_{\alpha\beta}(t)$ is not a real generator, but rather a “pseudo generator” introduced here to simplify formulae to follow. In its turn, this means that diagonal elements of $\tilde{Q}^{(m)}_{\alpha\beta}(t)$ cannot be made arbitrarily negative, as otherwise we would end up with probabilities reaching outside of the unit interval $[0,1]$.

Now we plug Eq.(50) and Eq.(53) into Eq.(52). This produces the following relation (here we omit $O\left(dt^2\right)$ terms):

$$\delta_{ij}\delta_{\alpha\beta} + A_{i\alpha|j\beta}(t)dt = \hat{P}^{(ij)}_{\alpha\beta}(t,t+dt)\left[\delta_{ij} + \hat{A}^{(\alpha)}_{ij}(t)dt\right]. \qquad (56)$$

A more explicit expression for the generator $A_{i\alpha|j\beta}$ in terms of auxiliary generators $\hat{A}^{(\alpha)}_{ij}$, $\hat{Q}^{(i)}_{\alpha\beta}$ and $\tilde{Q}^{(m)}_{\alpha\beta}$ can be obtained using the following identity (which can be checked by inspection):

$$A_{i\alpha|j\beta} = (1-\delta_{ij})A_{i\alpha|j\beta} + (1-\delta_{\alpha\beta})A_{i\alpha|j\beta} - (1-\delta_{ij})(1-\delta_{\alpha\beta})A_{i\alpha|j\beta} + \delta_{ij}\delta_{\alpha\beta}A_{i\alpha|i\alpha}. \qquad (57)$$

Using Eq.(56) to evaluate different terms in the right-hand side of Eq.(57) in terms of the auxiliary generators, we obtain, after some algebra, the following general representation of generator $A_{i\alpha|j\beta}(t)$ of USLV(2,2):

$$A_{i\alpha|j\beta} = \delta_{ij}\hat{Q}^{(i)}_{\alpha\beta} + (1-\delta_{ij})\tilde{Q}^{(j-i)}_{\alpha\beta}\hat{A}^{(\alpha)}_{ij} + \delta_{\alpha\beta}\hat{A}^{(\alpha)}_{ij}. \qquad (58)$$

Different terms in this expression are interpreted as follows.[25]

The first term $\delta_{ij}\hat{Q}^{(i)}_{\alpha\beta}$ is a generator of idiosyncratic jumps of $\mathbf{Y}_t$ that proceed without a simultaneous jump of $\mathbf{Z}_t$ in the interval $[t,t+dt]$. Various continuous-space models can be used as a means to parametrize this generator via discretization of the state space. For example, starting with a diffusive model for $\mathbf{Y}_t$, we end up with a tridiagonal generator matrix $\hat{Q}^{(i)}_{\alpha\beta}$. More details and examples will be given below in Sect. 6.6.

The second term in Eq.(58) is a generator of joint jumps of $(\mathbf{Z}_t,\mathbf{Y}_t)$. Note that it is a valid generator on its own as long as $\sum_{\beta}\tilde{Q}^{(m)}_{\alpha\beta} = 0$. Again, different specifications of this generator can be considered within our general framework. This will be discussed in some detail below in Sect. 6.7.

[25] We thank Leonid Malyshkin for proposing a decomposition of the Markov generator in such form, as well as for discussions that helped improve the presentation in this section.

Finally, the last term in Eq.(58) is a generator of idiosyncratic jumps of $\mathbf{Z}_t$ that proceed without a simultaneous jump of $\mathbf{Y}_t$. It is determined by the conditional Markov chain generator $\hat{A}^{(\alpha)}_{i|j}(t)$. This generator plays a special role in our construction. It is special because the conditional Markov chain generator $\hat{A}^{(\alpha)}_{i|j}(t)$ is the *only* generator in Eq.(58) that impacts prices of European vanilla options, while prices of exotic options will in general depend on all generators that enter Eq.(58). As will be explained in more detail below, this is due to the following relations that follow as long as $\sum_\beta \hat{Q}^{(i)}_{\alpha\beta} = 0$ and $\sum_\beta \tilde{Q}^{(m)}_{\alpha\beta} = 0$:

$$\begin{aligned} \sum_\beta A_{i\alpha|j\beta}(t) &= \hat{A}^{(\alpha)}_{ij}(t), \\ \sum_\beta p_{i\alpha|j\beta}(t) &= \delta_{ij} + \hat{A}^{(\alpha)}_{ij}(t)dt. \end{aligned} \tag{59}$$

Note that the fact that theoretical prices of vanilla options computed in USLV(2,2) do not depend on specification of the other generators $\hat{Q}$ and $\tilde{Q}$ in Eq.(58) has a few interesting implications.

First, it suggests a nice "orthogonality" property of model parameters determining various generators that enter Eq.(58), such that parameters driving prices of exotic options can be tuned (or picked) without impacting calibration to vanillas. If prices of some exotic options are available in the marketplace, this can be used to calibrate these two generators, after the model is calibrated to available vanillas.

Second, in a scenario where no reliable pricing information is available for exotic options, we could use this property of the model in order to specify a measure of "exoticness" as, e.g., the amount the price of the given exotic derivative moves under certain functional or parametric tweaks of the first two generators in Eq.(58). Given two exotic options and given tweaks to be performed on the generators in Eq.(58) (such as a common rescaling of all elements) while pricing both options, one of the options from the pair would in general end up being "more exotic" than the other. While these issues will be addressed in a future work, here we concentrate on the problem of calibrating the model to European vanilla option prices.

6.4 Conditional Markov Chain Generator

Clearly, prices of European vanilla options on a given underlying $\mathbf{Z}_t$ for a set of options maturing at times $T_1, T_2, \ldots$ are only determined by marginal distributions of $\mathbf{Z}_t$ at these times. An equation driving evolution of marginal $\mathbf{Z}$-distributions in the full USLV(2,2) model can be obtained by summing over $\beta = (\beta_1, \beta_2)$ in the forward equation Eq.(49). We obtain

$$\pi(j, t+dt) = \sum_\beta \sum_{i,\alpha} p_{i\alpha|j\beta}(t, t+dt)\pi(i,\alpha,t) = \sum_{i,\alpha} \left(\delta_{ij} + \hat{A}^{(\alpha)}_{ij}(t)dt\right) \pi(i,\alpha,t), \tag{60}$$

where we used Eq.(59) for the last equality. This justifies the claim we made above: observed prices of European vanilla options impose certain constraints on the conditional Markov chain generator $\hat{A}_{ij}^{(\alpha)}$, but not on the other generators appearing in Eq.(58). Rearranging terms in Eq.(60), we obtain

$$\frac{d\pi(j,t)}{dt} = \sum_{i,\alpha} \hat{A}_{ij}^{(\alpha)}(t)\pi(i,\alpha,t). \tag{61}$$

An explicit expression for the conditional Markov chain generator $\hat{A}_{ij}^{(\alpha)}(t)$ can be obtained using Eq.(52):

$$\hat{A}_{ij}^{(\alpha)} = \frac{1}{dt}\left[P\left[\mathbf{Z}_{t+dt} = \mathbf{Z}_j | \mathbf{Z}_t = \mathbf{Z}_i, \mathbf{Y}_t = \mathbf{Y}_\alpha\right] - \delta_{ij}\right] + O\left(dt^2\right). \tag{62}$$

From this point onwards, we reserve the notation $\hat{A}_{ij}^{(\alpha)}$ for a *calibrated* generator of USLV(2,2), while using the notation $A_{ij}^{(\alpha)}$ for an *initial guess* for $\hat{A}_{ij}^{(\alpha)}$. The latter is assumed to be a valid generator (obtained from some consistent model) that is not necessarily accurately calibrated to the observed market. In what follows, we will refer to the latter as a *prior* conditional Markov chain generator. While a particular functional relation between the two generators $\hat{A}_{ij}^{(\alpha)}$ and $A_{ij}^{(\alpha)}$ will be considered in the next section, in the reminder of this section we concentrate on specifying the second, "prior" generator $A_{ij}^{(\alpha)}$.

As was outlined above (see also Appendix B), we define $A_{ij}^{(\alpha)}$ as a combination of generators A_1 and A_2 (see Eq.(47)), scaled by two components of $\mathbf{Y}_t = \left(Y_t^{(1)}, Y_t^{(2)}\right)$:

$$A_{ij}^{(\alpha)} = Y_{\alpha_1}^{(1)} A_1 + Y_{\alpha_2}^{(2)} A_2. \tag{63}$$

Recalling the original definition Eq.(40) of the Markov chain generator, we can write this in a matrix form:

$$A_{ij}^{(\alpha)} = \left(\begin{array}{ccccccc} L_Y^{(0)} & F_Y^{(0)} & 0 & 0 & \cdots & 0 & 0 \\ B_Y^{(1)} & L_Y^{(1)} & F_Y^{(1)} & 0 & \cdots & 0 & 0 \\ 0 & B_Y^{(2)} & L_Y^{(2)} & F_Y^{(2)} & \cdots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \cdots & B_Y^{(p)} & L_Y^{(p)} \end{array}\right), \tag{64}$$

where all matrices $B_Y^{(i)}$, $F_Y^{(i)}$ are obtained by scaling of $B^{(i)}$, $F^{(i)}$ by $Y_{\alpha_1}^{(1)}$:

$$B_Y^{(i)} = Y_{\alpha_1}^{(1)} B^{(i)}\,, \quad F_Y^{(i)} = Y_{\alpha_1}^{(1)} F^{(i)}, \tag{65}$$

while elements of $L_Y^{(0)}$ are scaled by $Y_{\alpha_2}^{(2)}$, except for the diagonal elements:

$$\left(L_Y^{(i)}\right)_{jk} = \begin{cases} Y_{\alpha_2}^{(2)} L_{jk}^{(i)} & \text{if } k = j \pm 1, \\ Y_{\alpha_2}^{(2)}\left(L_{jj}^{(i)} + \hat{F}_{jj}^{(i)}\right) - Y_{\alpha_1}^{(1)} \hat{F}_{jj}^{(i)} & \text{if } k = j, \end{cases} \tag{66}$$

where $\hat{F}^{(i)}$ is defined in Eq.(47). Using Eq.(36) in Eq.(66), we obtain the explicit expression:

$$\left(L_Y^{(i)}\right)_{jk} = \begin{cases} Y_{\alpha_2}^{(2)} \frac{s_{ij}(t)}{2}\left(\frac{\sigma_2^2}{\Delta z_2^2} - \frac{|\rho|\sigma_1\sigma_2|}{\Delta z_1 \Delta z_2}\right) & \text{if } k = j \pm 1, \\ -Y_{\alpha_2}^{(2)} s_{ij}(t)\left(\frac{\sigma_2^2}{\Delta z_2^2} - \frac{|\rho|\sigma_1\sigma_2|}{\Delta z_1 \Delta z_2}\right) - Y_{\alpha_1}^{(1)} s_{ij}(t)\frac{\sigma_1^2}{\Delta z_1^2} & \text{if } k = j. \end{cases} \tag{67}$$

Clearly, conditional on values $Y_{\alpha_1}^{(1)}, Y_{\alpha_2}^{(2)} \geq 0$, diagonal elements of the conditional Markov chain generator Eq.(64) given by the second line of Eq.(66) are negative (as long as Eq.(37) holds), while all off-diagonal elements are positive, and the row-wise sums of elements in Eq.(64) are all zeros. Therefore, Eq.(64) is a valid conditional generator for any fixed values of $Y_{\alpha_1}^{(1)}, Y_{\alpha_2}^{(2)} \geq 0$. The first component $Y^{(1)}$ modulates transitions between $Z^{(1)}$-states (which may or may not be accompanied by transitions between $Z^{(2)}$-states), while the second component $Y^{(2)}$ modulates transitions between $Z^{(2)}$-states without simultaneous transitions between $Z^{(1)}$-states.

6.5 Fast Calibration of USLV(2,2) by 1D Forward Induction

In this section, we present a fast calibration algorithm that enables a recalibration of the full 2D USLV(2,2) model starting from a calibrated 1D USLV(2,0) without using a forward induction on a full-blown 2D Markov chain. It uses a recursive procedure of "integrating in" the stochastic volatility process. Our method is similar to Arnsdorf & Halperin (2007). In its turn, a fast calibration method on a 2D Markov chain used in Arnsdorf & Halperin (2007) is similar to an algorithm originally developed by Britten-Jones and Neuberger (BJN)[26].

Recalling our previous notation where we used symbols $\hat{A}$ and A for the calibrated and "prior" conditional Markov chain generator, we assume the following relation between them:

$$\hat{A}_{ij}^{(\alpha)}(t) = (1 - \delta_{ij})\, q_{ij}(\mathbf{Y}_t, t) A_{ij}^{(\alpha)}(t) - \delta_{ij} \sum_{m \neq 0} q_{jm}(\mathbf{Y}_t, t) A_{jm}^{(\alpha)}(t). \tag{68}$$

Here $q_{ij}(\mathbf{Y}_t, t) \geq 0$ are adjustment factors that will be used below to calibrate the full-blown USLV(2,2) model.[27] Note that Eq.(68) defines a valid generator as long as $q_{ij}(\mathbf{Y}_t, t) \geq 0$ and $A_{jm}^{(\alpha)}(t)$ is a valid generator, as all nondiagonal elements of $\hat{A}_{ij}^{(\alpha)}(t)$ are non-negative, all diagonal elements are negative, and all rows sum up to zero.

The purpose of introducing the adjustment factors $q_{ij}(\mathbf{Y}_t, t)$ in Eq.(68) is to provide degrees of freedom needed for calibration to option prices in the (2,2) model in a way similar to the way the 1D speed factors were used above to calibrate the (2,0) model without stochastic volatility. As will be shown below, such calibration can be done in a numerically efficient way by reutilizing results of a previous calibration in a local volatility USLV(2,0) model. Note that after calibration of USLV(2,2) is done via a choice of multiplicative adjustment factors $q_{ij}(\mathbf{Y}_t, t)$, the latter can be combined with the 1D SFs $s_{ij}(t)$ that appear in the "prior" generator $A_{jm}^{(\alpha)}(t)$ to produce 2D SFs $S_{ij}(Y_t, t)$.

[26] This approach was later popularized by Piterbarg (2006) under the name "Markovian projection." Note that both BJN and Peterbarg cite the work by Dupire on the link between stochastic and local volatility models. Dupire's approach seems to provide a common basis for both the BJN and Markov projection methods.

[27] The theoretical interpretation of adjustment factors $q_{ij}(\mathbf{Y}_t, t)$ is that they provide "risk-neutralizing" drift corrections to the dynamics in the presence of stochastic volalitity; see a related discussion in Britten-Jones & Neuberger (2000) and Rossi (2002).

To proceed, we first plug Eq.(68) into Eq.(61). This yields

$$\frac{d\pi(j,t)}{dt} = \sum_{i\neq j,\alpha} \left[q_{ij}(\mathbf{Y}_t,t) A_{ij}^{(\alpha)}(t)\pi(i,\alpha,t) - q_{ji}(\mathbf{Y}_t,t) A_{ji}^{(\alpha)}(t)\pi(j,\alpha,t) \right]. \tag{69}$$

This can be compared to the forward equation obtained in the USLV(2,0) model where we have the following definition of the generator:

$$p_{ij}(t,t+dt) = \delta_{ij} + A_{ij}dt + O\left(dt^2\right), \tag{70}$$

while the forward equation has the form

$$\frac{d\pi(j,t)}{dt} = \sum_{i\neq j} \left[A_{ij}(t)\pi(i,t) - A_{ji}(t)\pi(j,t) \right]. \tag{71}$$

Comparing Eq.(69) and Eq.(71), we find that marginal distributions $\pi(j,t+dt)$ in both the USLV(2,2) and USLV(2,0) models are matched at each node $j = (j_1, j_2)$ provided we make the following choice for adjustment factors $q_{ij}(t)$ in Eq.(68):

$$q_{ij}(\mathbf{Y}_t,t) = \frac{A_{ij}(t)\pi(i,t)}{\sum_\alpha A_{ij}^{(\alpha)}(t)\pi(i,\alpha,t)} = \frac{A_{ij}(t)\sum_\alpha \pi(i,\alpha,t)}{\sum_\alpha A_{ij}^{(\alpha)}(t)\pi(i,\alpha,t)}. \tag{72}$$

We now use our key result Eq.(72) to set up a convenient and fast forward-induction method for the calibration of USLV(2,2) that utilizes the results of the 1D calibration of the Z_t-Markov chain of the USLV(2,0) model with a calibrated generator $A_{ij}(t)$, starting with a "prior" conditional generator $A_{ij}^{(\alpha)}(t)$ of the USLV(2,2) model.

We assume that the 1D calibration of the Z_t-Markov chain is performed as discussed above. We start with the initial conditions for the 2D and 1D probability distributions, correspondingly,

$$\pi(i,\alpha,0) = \delta_{i\hat{i}}\delta_{\alpha\hat{\alpha}}\,, \quad \pi(i,0) = \delta_{i\hat{i}}, \tag{73}$$

where $\hat{i} = (\hat{i}_1, \hat{i}_2)$ and $\hat{\alpha} = (\hat{\alpha}_1, \hat{\alpha}_2)$ are indices corresponding to the initial values of Z_0 and Y_0 (which we assume to be known), respectively. Using Eq.(72), we solve for $q_{\hat{i}j}(\mathbf{Y}_0, 0)$:

$$q_{\hat{i}j}(\mathbf{Y}_0,0) = \frac{A_{\hat{i}j}(0)}{A_{\hat{i}j}^{(\hat{\alpha})}(0)}. \tag{74}$$

Note that for $i \neq \hat{i}$, the correction factors at time $t = 0$ are undefined. However, this does not pose any problem as such states are unachievable at time $t = 0$, and therefore play no role in the dynamics. If desired, these parameters can be assigned some dummy values that would not have any impact on any numerical results produced with the model.

Next we use the forward equation on interval $[0, dt]$ to compute the joint probability $\pi(j, \beta, dt)$, which is then used to compute the adjustments for all nodes at time $t = dt$, and so on. As a result, we have a full 2D USLV(2,2) Markov chain calibrated to the set of option quotes using a fast and effective algorithm.

6.6 Generator of Idiosyncratic Dynamics of $\mathbf{Y}_t$

In this section, we provide some examples of specification of the first generator, $\delta_{ij}\hat{Q}^{(i)}_{\alpha\beta}$, in Eq.(58). We recall that by construction of the model, the choice of this generator in USLV(2,2) has no impact on the quality of calibration of the model to prices of European vanilla options in a calibration set, while in general it does impact prices of exotic options produced by the model.

As was mentioned above, for practical applications we typically have in mind a low (e.g., 3 to 11) number of possible states per dimension of the stochastic volatility factor. This has two implications.

First, we prefer to view continuous-space models as a convenient and compact way to parametrize the generator $\delta_{ij}\hat{Q}^{(i)}_{\alpha\beta}$ in Eq.(58). This generator corresponds to idiosyncratic moves of $\mathbf{Y}_t$ without simultaneous moves of $\mathbf{Z}_t$. Such parametrization is clearly preferred to directly specifying $\sim 2(2q+1)^2$ free parameters defining a discrete-state generator $\delta_{ij}\hat{Q}^{(i)}_{\alpha\beta}$.

Second, as long as the number of volatility states is low, the generator matrix for $\mathbf{Y}_t$ should not necessarily be sparse. This remark is important as nonsparse matrices arise while discretizing jump-diffusion processes.

For simplicity, in this paper we restrict ourselves to a particular bivariate mean-reverting diffusion process as a continuous-space model that produces generator $\hat{Q}^{(i)}_{\alpha\beta}$ after a proper discretization.[28] More specifically, we consider a bivariate Ornstein-Uhlenbeck (OU) process for the logarithmic variables $y^{(i)}_t = \log Y^{(i)}_t$:

$$
\begin{aligned}
dy^{(1)}_t &= k_1\left(\eta_1 - y^{(1)}_t\right)dt + \nu_1 dW^{(1)}_t, \\
dy^{(2)}_t &= k_2\left(\eta_2 - y^{(2)}_t\right)dt + \nu_2 dW^{(2)}_t,
\end{aligned}
\tag{75}
$$

where the two Brownian motions $W^{(1)}_t$ and $W^{(2)}_t$ are correlated with correlation ρ_y.

The continuous-space Markov generator corresponding to Eq.(75) reads

$$
\mathcal{L}_y V(\mathbf{y}) = b_1(y)\frac{\partial V(\mathbf{y})}{\partial y_1} + b_2(y)\frac{\partial V(\mathbf{y})}{\partial y_2} + \frac{1}{2}\nu_1^2\frac{\partial^2 V(\mathbf{y})}{\partial y_1^2} + \frac{1}{2}\nu_1^2\frac{\partial^2 V(\mathbf{y})}{\partial y_1^2} + \rho_y\nu_1\nu_2\frac{\partial^2 V(\mathbf{y})}{\partial y_1 \partial y_2}, \tag{76}
$$

where

$$
b_i(y) = k_i\left(\eta_i - y^{(i)}_t\right), \quad i = 1, 2. \tag{77}
$$

Note that parameters of generator $\mathcal{L}_y$ can be made dependent on the value of $\mathbf{Z}_t$ if desired.

The Y-generator can be discretized in a similar way to a procedure used above in Sect. 5. We use central differences for the second derivatives and a noncentral difference for the mixed derivative in Eq.(76). In addition, we use the upwind-difference discretization for the first derivatives in Eq.(76). Regrouping different terms in the discretized generator much as it was done above in Sect. 5, the resulting discrete Markov chain generator for $\mathbf{Y}_t$ can be cast

[28] If any other model of stochastic volatility is chosen instead of the one presented below, the only change needed in the present framework would be to construct different transition matrices (or generators) for the Y-states, while the computational part of calibration and pricing would stay the same.

in the form of another QBD process, in an analogous way to our construction of the QBD process for $\mathbf{Z}_t$. The resulting generator takes the familiar QBD form (compare, e.g., with Eq.(64)):

$$\hat{Q}^{(i)} = \begin{pmatrix} L^{(0)} & F^{(0)} & 0 & 0 & \cdots & 0 & 0 \\ B^{(1)} & L^{(1)} & F^{(1)} & 0 & \cdots & 0 & 0 \\ 0 & B^{(2)} & L^{(2)} & F^{(2)} & \cdots & 0 & 0 \\ \vdots & & & & & & \\ 0 & 0 & 0 & 0 & \cdots & B^{(2q)} & L^{(2q)} \end{pmatrix}, \tag{78}$$

where all matrices $B^{(i)}, L^{(i)}$, $F^{(i)}$ have dimension $(2q+1) \times (2q+1)$, i.e., the dimension of our one-dimensional grids.

We would like to conclude this section with a few remarks. Using a diffusive prototype is certainly not the only way of constructing the idiosyncratic generator $\hat{Q}^{(i)}$ in Eq.(58). Alternatively, we could consider more general jump-diffusion or Lévy processes as continuous-space prototypes of the generator. For such more general specifications, generator $\hat{Q}^{(i)}(t)$ would not in general be sparse. Note, however, that a potential nonsparsity of the generator in the latter case would not be a major concern if we have a small number of states for $\mathbf{Y}_t$.

Furthermore, as was mentioned above, a BJN-like fast 1D calibration procedure for AR-USLV(2,2) presented below in Sect. 6.5 can also be applied when instead of *spot* variance factors $\mathbf{Y}_t$, we use *integrated* variance drivers/subordinators $\mathbf{T}_t$. The only difference from the case of spot variance factors just presented would be that generators for subordinators should have all zeros below the diagonals (as a subordinator should be a non-decreasing function of time). However, both the calibration and pricing algorithms would stay the same.

6.7 Generator of Joint Jump Dynamics

Here we consider the second term $(1-\delta_{ij})\tilde{Q}^{(j-i)}_{\alpha\beta}\hat{A}^{(\alpha)}_{ij}$ in Eq.(58). We recall that this expression defines the generator of joint jumps whose matrix elements give intensities of simultaneous jumps of $\mathbf{Z}_t$ and $\mathbf{Y}_t$.

The motivation for introducing joint jumps of $\mathbf{Z}_t$ and $\mathbf{Y}_t$ is to incorporate the asset-volatility codependence[29] in our framework. We note that in a discrete-time setting, both Britten-Jones & Neuberger (2000) and Rossi (2002) introduce different transition matrices for the Y-states for different transitions between the Z-states in an *ad hoc* way. Our approach, which starts with a continuous time dynamics and the decomposition Eq.(58) of the Markov generator, is hopefully a bit more systematic and easier to relate to one's intuition.

As the conditional Markov chain generator $\hat{A}^{(\alpha)}_{ij}$ is fixed by the above procedure of calibration to vanilla options, the joint jump generator is specified by defining the pseudo-generator $\tilde{Q}^{(m)}_{\alpha\beta}(t)$ (see Eq.(55)). Much as we did in our approach above for the idiosyncratic Y-generator, here we settle for a very simple and parsimonious choice. Alternative and more complicated specifications of the pseudo-generator $\tilde{Q}^{(m)}_{\alpha\beta}(t)$ could clearly be considered instead without significantly affecting complexity or performance of the model.

[29] Such as the well-known leverage effect (a negative spot-volatility correlation) for equity markets.

Specifically, we assume that conditionally on $\left(\Delta Z_t^{(1)}, \Delta Z_t^{(2)}\right)$, jumps $\Delta Y_t^{(1)}$ and $\Delta Y_t^{(2)}$ are independent. We further specify that jump $\Delta Y_t^{(1)}$ depends only on $\Delta Z_t^{(1)}$ rather than on the pair $\left(\Delta Z_t^{(1)}, \Delta Z_t^{(2)}\right)$, and similarly $\Delta Y_t^{(2)}$ depends only on $\Delta Z_t^{(2)}$. To control the amount of the asset-volatility codependence in our model, we introduce two parameters, γ_1 and γ_2, that determine the codependence for the joint moves $\left(\Delta Z_t^{(1)}, \Delta Y_t^{(1)}\right)$ and $\left(\Delta Z_t^{(2)}, \Delta Y_t^{(2)}\right)$, respectively, such that $\gamma_i > 0$ and $\gamma_i < 0$ correspond to positive and negative codependences, respectively. We then define the pseudo-generator $\tilde{Q}_{\alpha\beta}^{(m)}(t)$ as follows:[30]

$$\begin{aligned} \tilde{Q}_{\alpha\beta}^{(m)}(t) &= \left[\delta_{\alpha_1\beta_1} + (\gamma_1 m_1)^+ A_{\alpha_1\beta_1}^{(1,+)} + (\gamma_1 m_1)^- A_{\alpha_1\beta_1}^{(1,-)}\right] \\ &\times \left[\delta_{\alpha_2\beta_2} + (\gamma_2 m_2)^+ A_{\alpha_2\beta_2}^{(2,+)} + (\gamma_2 m_2)^- A_{\alpha_2\beta_2}^{(2,-)}\right] - \delta_{\alpha_1\beta_1}\delta_{\alpha_2\beta_2}\,, \quad (m_1, m_2) \neq (0,0), \end{aligned} \tag{79}$$

where for any real number x we defined $x^+ = \max(0, x)$ and $x^- = \max(0, -x)$. Matrices $A^{(+)}$ and $A^{(+)}$ in Eq.(79) stand for some upper- and lower-triangular generator matrices, respectively. For example, a nonsparse specification of generator Eq.(79) could be provided using two parameters, $q_1, q_2 \leq 1$, that determine the speed of decay of the generator away from the diagonal:

$$\begin{aligned} A_{\alpha\beta}^{(i,+)} &= \theta(\beta - \alpha) q_i^{\beta-\alpha-1} - \delta_{\alpha\beta} \sum_{\beta>\alpha} q_i^{\beta-\alpha-1}\,, \quad i = 1, 2, \\ A_{\alpha\beta}^{(i,-)} &= \theta(\alpha - \beta) q_i^{\alpha-\beta-1} - \delta_{\alpha\beta} \sum_{\beta<\alpha} q_i^{\alpha-\beta-1}\,, \quad i = 1, 2, \end{aligned} \tag{80}$$

where $\theta(x)$ stands for a Heavyside step-function. Alternatively, if we want to keep the generator sparse, we could consider bi-diagonal specifications for matrices $A^{(i,\pm)}$.

7 ITC-USLV(2,2): Implied Time Change Process

The algorithm of forward induction-based calibration of the USLV model just presented is simple and intuitive; however, it is not ideal from a practical viewpoint, as it assumes that the time steps are small enough to justify the use of a trinomial (birth-and-death) approximation for the diffusion process in $\mathbf{Z}_t$. In practice, this means we should take daily (possibly hourly) steps, which may slow down calibration and pricing. If we want to be able to have a lattice with larger steps (e.g., monthly), or work with irregular large steps, we need a different method.

Several alternatives of different complexity can be considered at this point. One approach would be to generalize Eq.(68) to the case of larger time steps Δt by viewing the left- and right-hand sides of Eq.(68) as leading terms in expansions of finite-time matrix exponentials of the conditional (on a realization of $\mathbf{Y}_t$) generator of the QBD Markov chain for the calibrated and "prior" model, respectively. While it can be shown that the recursive forward

[30] This parameterization was proposed by Leonid Malyshkin.

calibration can be carried over in such framework as in the one-step BJN method described above, practical uses of such an approach may be constrained by our ability to compute conditional multi-step transition probabilities for **Z**-states in a numerically efficient way. We expect that splitting methods can be efficiently used to this end, but we leave research in this direction for a future work, and instead concentrate on alternative approaches. The latter are based on stochastic time change techniques, which is what we present next.

7.1 Modeling Stochastic Time Changes

Let ξ_t be a (random) matrix-valued value of the QBD process with generator Eq.(40) at time t. Consider a right-continuous nondecreasing process T_t with $\tau_0 = 0$ with independent and homogeneous increments.[31] In the present context, such process T_t is called a *Bochner subordinator*; see, e.g., Feller (1968).

Now consider a new process $\eta_t \equiv \xi_{T_t}$ given by our QBD process Eq.(40) evaluated at a random (business) time T_t instead of the calendar time t. This produces a QBD Markov chain *subordinated* to the Bochner subordinator T_t.

Note that the idea of a subordinated Markov chain has already been used above in Sect. 5.1 (see also a discussion on this point in Gross & Miller (1984)) as a computational tool for evaluation of matrix exponentials of generator Eq.(40). A more general subordinator is given by a nondecreasing Lévy process; see, e.g., Carr et al. (2003) and references therein. It can be written as

$$T_t = \int_0^t Y_s ds + T_t^{(jump)}, \tag{81}$$

where Y_t is a nonnegative process called the *activity rate*, and $T_t^{(jump)}$ stands for a jump component of the time change.

Many specifications of a time change process can be described by the general formula Eq.(81). For example, subordination by a Poisson process was used above in Sect. 5.1, which corresponds to the Bochner subordinator being a pure jump process with a finite jump activity. Another simple choice for a pure jump time change would be a gamma process (incidentally, this process has a particularly simple Laplace transform). Alternatively, one can consider purely diffusive time changes where $T_t^{(jump)}$ vanishes, with the activity rate Y_t specified, e.g., by a CIR process or a positive OU process.

Let us assume that the stochastic time change is independent of the QBD Markov chain, and that its Laplace transform

$$\mathcal{L}_{T_t}(u) = \mathbb{E}\left[e^{-uT_t}\right] \tag{82}$$

is known in closed form, or can be computed numerically at a low cost. Consider first a one-factor stochastic time change for a single Markov chain with generator A. Recall that finite-time transition probabilities can be computed using the randomization method as in

[31] That is, $\tau_{t+s} - \tau_t$ is independent of the filtration $\mathcal{F}_t$ and has the same distribution as τ_s.

Eq.(45), which we write here as

$$P(t) = P_0' e^{-\lambda t} \sum_{n=0}^{\infty} \frac{(\lambda t)^n \mathbf{P}^n}{n!} = \sum_{n=0}^{\infty} \frac{(-1)^n}{n!} P_0' \mathbf{P}^n \left(\lambda \frac{d}{d\lambda} \right)^n e^{-\lambda t}. \tag{83}$$

Substituting here $t \to T_t$, taking the expectation with respect to future scenarios of the time change process T_t, and interchanging the summation and expectations in Eq.(83), we obtain

$$\mathbb{E}\left[P(T_t) \right] = \sum_{n=0}^{\infty} \frac{(-1)^n}{n!} P_0' \mathbf{P}^n \left(\lambda \frac{d}{d\lambda} \right)^n \mathcal{L}_{T_t}(\lambda). \tag{84}$$

Note that in practice, randomization methods truncate an infinite series in Eq.(83) at some finite n_{max} determined by a needed accuracy level ε. If we first truncate the series for fixed t given ε, and *then* do a stochastic time change, this might lead to a substantial loss of accuracy. One possible way to achieve a fixed-ε calculation for a given t in the model with stochastic time T_t is as follows. We first find a maximum value of $\tau_{max} = \max_{\tau} T_t$ that can be reached at some confidence level, and then truncate the sum at some value n'_{max} that provides needed accuracy for Eq.(83) where t is replaced by τ_{max}.

This means that if derivatives of the Laplace transform of the time change are easy to compute, and the number of terms we need to keep in Eq.(84) for given tolerance is reasonably small, then the problem of parametric calibration of parameters of the Y-process in the USLV(1,1) (or USLV(2,1), see below) models can be solved, in the zero-correlation limit, in one step, with no need for a forward induction algorithm.

Note that while the assumption of zero correlation might be restrictive, the above approach can in fact be generalized to the case of nonzero correlation, at the price of introducing a complex-valued measure (see Carr & Wu (2004)). This could be used to calibrate parameters of the Y process for a given set of option quotes while keeping the 1D SFs fixed or flat.[32] Further improvements of calibration quality (if desired) could then be achieved using a forward-induction-based calibration method described in Sect. 6.5.

For the most interesting *two-factor* time change specification (i.e., for USLV(2,2)), the situation is more tricky because of correlations between different drivers, as well as because of noncommutativity of generators A_1 and A_2. However, as will be shown in the next section, it turns out that a tractable framework can be obtained with a proper construction of a two-factor time change.

7.2 Time Change with a Hierarchical Bivariate Subordinator

It might be tempting to try to extend our framework by incorporating stochastic time changes with a nonvanishing jump component $T_t^{(jump)}$ in Eq.(81), with a two-factor time change $\mathbf{T}_t = (T_t^{(1)}, T_t^{(2)})$. If $T_t^{(1)}, T_t^{(2)}$ include jumps, this leads to jumps in the underlyings $\mathbf{Z}_t =$

[32] Alternatively, we could use the zero-correlation limit as a "quick and dirty" way to estimate the parameters of the full model with nonzero correlation, except of course those parameters that are critically dependent on the level of correlation.

$\left(Z_t^{(2)}, Z_t^{(2)}\right)$. If both the nonvanishing activity rate $\mathbf{Y}_t$ and a jump component $\hat{T}_t^{(jump)}$ are retained in Eq.(81), this results (in our setting) in a four-factor stochastic-local volatility dynamics with jumps in both the underlyings and volatility.

Now we introduce such a two-factor stochastic time change with a bivariate subordinator and show how to evaluate finite-time transition probabilities in a resulting subordinated Markov chain using a generalization of the randomization method presented in Sect. 5.1.

Recall from Sect. 6.2 that the Markov chain generator A in our problem has the decomposition Eq.(47), where both A_1 and A_2 are valid generators that can be subject to individual stochastic time changes. Consider a bivariate subordinator of the following form

$$\mathcal{T}_t = \left(\begin{array}{c} T_t^{(1)} \\ T_t^{(2)} \end{array} \right) = \left(\begin{array}{c} \theta_t T_t \\ T_t \end{array} \right). \tag{85}$$

Here $\theta_t > 0$ is a stochastic process with $\mathbb{E}[\theta_t] = 1$ that will be specified in more detail below. We assume that θ_t is independent of T_t. As θ_t acts as a time dilation factor on top of the random time T_t, we will refer to θ_t as a *time-dilaton* process. Note that correlation between $T_t^{(1)}$ and $T_t^{(2)}$ is now driven by the variance of θ_t:

$$\rho_{T_t^{(1)}, T_t^{(2)}} = \sqrt{\frac{Var\left(T_t\right)}{Var\left(T_t\right) + Var\left(\theta_t\right)\left(Var\left(T_t\right) + \left(\mathbb{E}\left[T_t\right]\right)^2\right)}} \tag{86}$$

so that we can fit any nonnegative correlation between $T_t^{(1)}$ and $T_t^{(2)}$ by a proper choice of $Var(\theta_t)$.

We assume that (T_t, θ_t) is a 2D Markov process with independent and time-homogeneous increments. Furthermore, we assume that both T_t and θ_t are non-decreasing exponential-Lévy processes. As a product $\theta_t T_t$ of two (non-decreasing) exponential-Lévy processes T_t and θ_t is another (non-decreasing) exponential-Lévy process, Eq.(85) defines a valid subordinator that can be used to time-change our QBD process with generator Eq.(47).

The interpretation of the bivariate subordinator Eq.(85) is as follows. The second component $T_t^{(2)} = T_t$ provides a common time change that modulates all transitions on the chain; i.e., it acts on both generators A_1 and A_2. The first component $T_t^{(1)} = \theta_t T_t$ can be thought of as a *hierarchical* time change. In this hierarchical scheme, we first apply a common time change T_t , and then time-change it again using a linear time change function $T_t^{(1)}(T_t) = \theta_t T_t$. This time change will be applied below to generator A_1 alone.[33] Note that if both T_t and θ_t are non-decreasing exponential Lévy processes, this implies that the stochastic clock runs faster for transitions involving changes of both $Z^{(1)}$ and $Z^{(2)}$ than for transitions that only involve changes of $Z^{(2)}$. Also note that while θ_t is a stochastic *process*, in the context of calculation of finite-time transition probabilities on a fixed interval $t \in [0, t]$ (where t is some

[33] Note that the order of the time changes indicated above is very important: if we reversed it, this would be equivalent to allowing future events of transitions driven by A_1 to impact the dynamics of the whole system at the present time. For a recent application of such hierarchical time changes, see, e.g., Puzanova (2011).

"interesting" time, e.g., a coupon date) what matters is only a terminal value θ_t (see below in Eq.(87)). Therefore, for such calculation we can treat the terminal value θ_t as a random variable,[34] which certainly simplifies an approach presented below.

The finite-time transition probability for a time-changed QBD Markov chain can now be computed by conditioning on the realization of θ_t:

$$\begin{aligned} P(t) &= \mathbb{E}\left[P_0' e^{T_t^{(1)} A_1 + T_t^{(2)} A_2}\right] = \mathbb{E}\left[P_0' e^{\theta_t T_t A_1 + T_t A_2}\right] = \mathbb{E}\left[\mathbb{E}\left[P_0' e^{T_t(\theta_t A_1 + A_2)}\right] \middle| \, \theta_t\right] \\ &\equiv \mathbb{E}\left[\mathbb{E}\left[P_0' e^{T_t A_\theta} \middle| \, \theta_t\right]\right]. \end{aligned} \tag{87}$$

Here the outside expectation corresponds to averaging with respect to the randomness due to θ_t, while the inner expectation averages over the randomness due to T_t for a fixed value of θ_t. In the last equation, we have defined $A_\theta = \theta_t A_1 + A_2$ for any fixed $\theta_t = \theta$.

Now consider the inner expectation in Eq.(87). For any fixed $\theta_t = \theta$, we proceed as follows. First we specify a nonnegative parameter $\Lambda_\theta \geq \max_n |(A_\theta)_{nn}|$. Next, we use the idea of the randomization method of Sect. 5.1 to define a DTMC with transition matrix

$$\mathbf{P}_\theta = \mathbf{I} + \frac{\mathbf{A}_\theta}{\Lambda_\theta} \;\Rightarrow \mathbf{A}_\theta = \Lambda_\theta \left(\mathbf{P}_\theta - \mathbf{I}\right). \tag{88}$$

Substitute A as given by Eq.(88) into the solution of the forward equation, which we write here as $P_\theta(t)$ to emphasize that the whole calculation is done for a fixed $\theta_t = \theta$:

$$P_\theta(t) = \mathbb{E}\left[P_0' e^{T_t A_\theta} \middle| \, \theta_t\right] = \mathbb{E}\left[P_0' e^{T_t \Lambda_\theta (\mathbf{P}_\theta - \mathbf{I})} \middle| \, \theta_t\right] = \mathbb{E}\left[e^{-\Lambda_\theta T_t} P_0' e^{T_t \Lambda_\theta \mathbf{P}_\theta} \middle| \, \theta_t\right].$$

Using a Taylor series expansion for the matrix exponential in this expression and interchanging the summation and expectation, we obtain

$$P_\theta(t) = \sum_{n=0}^{\infty} \frac{(P_0' \mathbf{P}_\theta^n)}{n!} \mathbb{E}\left[(\Lambda_\theta T_t)^n e^{-\Lambda_\theta T_t} \middle| \, \theta_t\right] = \sum_{n=0}^{\infty} \frac{(-1)^n}{n!} \left(P_0' \mathbf{P}_\theta^n\right) \left(\lambda \frac{d}{d\lambda}\right)^n \mathcal{L}_{T_t}(\lambda) \Big|_{\lambda = \Lambda_\theta}, \tag{89}$$

where $\mathcal{L}_{T_t}(u)$ stands for a Laplace transform Eq.(82) of the time change T_t.

Eq.(89) is a "semi-analytical" expression for a finite-time transition probability in our Markov chain *after* the first time change driven by T_t, but *before* the second time change driven by θ_t. The product $P_0' \mathbf{P}_\theta^n$ can be efficiently implemented via a recursive vector-matrix multiplication as in Eq.(46). The derivatives $\left(\lambda \frac{d}{d\lambda}\right)^n \mathcal{L}_{T_t}(\lambda)$ can be easily computed if the Laplace transform $\mathcal{L}_{T_t}(\lambda)$ is known in closed form (or can be computed numerically at a low cost). Note that this part of the calculation is independent of any pricing data—the latter only impacts the matrix $\mathbf{P}_\theta$ through a calibrated set of 1D SFs s_i.

As an example, consider an exponential-gamma subordinator specification for $T_t^{(1)} = T_t$ with $T_t = \exp(X_t)$, where X_t is a Gamma process. Recall that an univariate (homogeneous) Gamma process $X_t \geq 0$ with $X_0 = 0$ and parameters $a, b \in \mathbb{R}^+$ is a process with independent

[34] This is due to the Markov property of the θ_t-dynamics which was assumed above: For a continuous-time Markov process, a time line can be chosen in an arbitrary way, while the corresponding finite-time transition probabilities would be related by the Chapman-Kolmogorov equations; see, e.g., Feller (1968).

increments such that X_t is Gamma-distributed $\Gamma(at, b)$ with the following probability density function (pdf):

$$f_{at,b}(x) = \frac{b^{at}}{\Gamma(at)} x^{at-1} e^{-bx} 1_{\mathbb{R}^+}(x), \tag{90}$$

with

$$\mathbb{E}\left[X_t\right] = \frac{at}{b}, \quad Var\left[X_t\right] = \frac{at}{b^2}. \tag{91}$$

Here the parameter a is called the shape parameter, and b is the rate parameter. In what follows, we set $a = b = 1/\nu$. The Laplace transform of Eq.(90) reads

$$\mathcal{L}_{X_t}(u) = (1 + \nu u)^{-\frac{t}{\nu}}. \tag{92}$$

Given this expression, we can approximately compute the Laplace transform of $T_t = \exp(X_t)$. All rescaled derivatives $\left(\lambda \frac{d}{d\lambda}\right)^n \mathcal{L}_{T_t}(\lambda)$ would then have to be computed from that latter Laplace transform.

After the condtional time-t probabilities are computed, the unconditional probabilities are obtained by averaging over a marginal probability density $p_t(\theta)$ of θ_t:

$$P(t) = \mathbb{E}\left[P_\theta(t)\right] = \int d\theta p_t(\theta) P_\theta(t). \tag{93}$$

In practice, this integral should be computed by discretization of the range of θ_t onto a finite grid $[\theta_0, \theta_1, \ldots, \theta_{q-1}]$.

Note that we can proceed in two different ways with a computation involved in Eq.(93). The first way would be to specify a process for θ_t, discretize it, and then compute a discrete approximation to Eq.(93). We could tune parameters of this process to fit a given set of option quotes. However, a particular parametric model of θ_t might be too restrictive for such task, especially if the number of option prices to fit grows larger. For this reason, in the next section we present a more flexible nonparametric approach that is able to fit any arbitrage-free set of option quotes.

7.3 Implied Time-Dilaton Process

For a particular parametric specification of a (discretized) time dilaton process θ_t, Eq.(93) produces some transitions probabilities for the Z-states in the time-changed QBD Markov chain. In general, these transition probabilities would be different from marginal probabilities in the local volatility USLV(2,0) calibrated to observed option prices. This means that a nearly perfect calibration to options achieved in the USLV(2,0) model would in general be lost once we add a stochastic time change to our model.

However, we can rematch the prices of options in our calibration set after a time change if we treat the distribution $p_t(\theta)$ of realizations of different values of θ_t as a distribution *implied* by option prices (given the specification of a process for T_t). Furthermore, for a multi-period setting specified by a particular time grid $t_0, t_1, \ldots$ (where t_i can be, e.g., swaption maturities in the calibration set), we can construct an implied *process* for θ_t if we impose a Markovian structure on it. In this section, we show how such process can be constructed using a Minimum Cross Entropy (MCE) method (see, e.g., Cover & Thomas (2006)). Our construction is similar to Halperin (2009) where analogous ideas were used in a different context.

Let F_{ijk} be a payoff function of the option with maturity t_i and strike K_j (with $j = 1, \ldots, K$) in a scenario where the terminal value of the underlying at the option maturity is given by values $\left(Z_{k_1}^{(1)}, Z_{k_2}^{(2)}\right)$, and let C_{ij} be the corresponding market prices of options in our calibration set. Using Eq.(93), we can express this scenario as a set of constraints[35]

$$\int d\theta p_i(\theta) G_{ij}(\theta) = C_{ij} \,, \quad G_{ij}(\theta) \equiv \sum_k F_{ijk} \left[P_\theta\right]_{\hat{k},k}(t_i) \,, \quad j = 0, \ldots, K, \tag{94}$$

where $\hat{k} = (\hat{k}_1, \hat{k}_2)$ is an index corresponding to the initial value $\mathbf{Z}_0$. Note that Eq.(94) with $j = 0$ corresponds to the constraint $\mathbb{E}[\theta_t] = 1$ implied above in Eq.(86), which is here enforced as an additional artificial option quote with $j = 0$, $F_{i0k} = \theta_t$ and $C_{i0} = 1$.

We can now find a probability density $p_i(\theta)$ that satisfies these constraints using the MCE approach. With this method, given a *reference* (*"prior"*) model $q_i(\theta)$ (given, e.g., by another exponential-gamma process), we minimize the Kullback-Leibler (KL) distance between the two distributions $p_i(\theta)$ and $q_i(\theta)$ (see Cover & Thomas (2006)):

$$D\left[p_i(\theta) || q_i(\theta)\right] = \int d\theta p_i(\theta) \log \frac{p_i(\theta)}{q_i(\theta)} \tag{95}$$

subject to constraints of Eq.(94). This produces a least biased (relatively to the reference measure $q_i(\theta)$) distribution $p_i(\theta)$ that satisfies the constraints of Eq.(94).

For the first node on the time grid, minimization of Eq.(95) with constraints Eq.(94) is done using the method of Lagrange multipliers. Using Eq.(95) with $i = 1$, the corresponding Lagrange function is

$$L = \int d\theta p_1(\theta) \log \frac{p_1(\theta)}{q_1(\theta)} - \sum_j \xi_j^{(1)} \left(\int d\theta p_1(\theta) G_{1j}(\theta) - C_{1j} \right), \tag{96}$$

where $\xi_j^{(1)}$ are Lagrange multipliers. Minimizing this expression with respect to $p_1(\theta)$, we obtain

$$p_1(\theta) = \frac{1}{Z_1} q_1(\theta) e^{\sum_j \xi_j^{(1)} G_{1j}(\theta)} \,, \quad Z_1 = \int d\theta q_1(\theta) e^{\sum_j \xi_j^{(1)} G_{1j}(\theta)}. \tag{97}$$

The Lagrange multipliers can now be computed by plugging Eq.(97) back into Eq.(96), and maximizing the resulting expression as a function of $\{\xi_j^{(1)}\}$. This amounts to a convex optimization problem in dimension equal to the number of option quotes. (For more details on the MCE method in both the one- multi-period settings, see, e.g., Halperin (2009) and references therein.)

[35] Note that we use the continuous notation here for simplicity of presentation only. For implementation, all stochastic processes are discretized within our approach.

For the second maturity, instead of minimizing the *unconditional* KL distance Eq.(95), we minimize a *conditional* KL distance for the next interval. This is done as follows. Using the Markov property we can write the pricing constraints as

$$C_{2,j} = \int_0^\infty d\theta_2\, p_2(\theta_2) G_{2j}(\theta_2) = \int_0^\infty d\theta_2\, G_{2j}(\theta_2) \int d\theta_1\, p_1(\theta_1)\, p(\theta_2|\theta_1). \tag{98}$$

Assuming that the density $p_1(\theta_1)$ is fixed at the previous step, the conditional transition density $p(\theta_2|\theta_1)$ can be found by minimization of the expected conditional KL cross entropy[36]

$$H\left[p(\theta_2|\theta_1)||q(\theta_2|\theta_1)\right] = \int d\theta_1\, p_1(\theta_1) \int d\theta_2 p(\theta_2|\theta_1) \log \frac{p(\theta_2|\theta_1)}{q(\theta_2|\theta_1)} \tag{99}$$

subject to pricing constraints Eq.(98). Here $q(\theta_2|\theta_1)$ is a prior transition probability. As the time change T_t should be nondecreasing, it should satisfy the condition $q(\theta_2|\theta_1) = 0$ for $\theta_2 < \theta_1$. Again, a natural choice for the prior transition density could be the transition density of an exponential-gamma process.

The corresponding Lagrange function for the second interval is

$$\begin{aligned} L &= \int d\theta_1\, p_1(\theta_1) \int d\theta_2 p(\theta_2|\theta_1) \log \frac{p(\theta_2|\theta_1)}{q(\theta_2|\theta_1)} \\ &- \sum_j \xi_j^{(2)} \left(\int_0^\infty d\theta_2\, G_{2j}(\theta_2) \int d\theta_1\, p_1(\theta_1)\, p(\theta_2|\theta_1) - C_{2,j} \right) \end{aligned} \tag{100}$$

where $\xi_j^{(2)}$ are Lagrange multipliers enforcing the constraints in Eq.(98). Minimizing this expression with respect to $p(\theta_2|\theta_1)$, we obtain the conditional transition probability

$$\begin{aligned} p(\theta_2|\theta_1) &= \frac{1}{Z_2(\theta_1, \xi^{(2)})} q(\theta_2|\theta_1) e^{\sum_j \xi_j^{(2)} G_{2j}(\theta_2)}, \\ Z_2(\theta_1, \xi^{(2)}) &= \int_0^\infty d\theta_2\, q(\theta_2|\theta_1) e^{\sum_j \xi_j^{(2)} G_{ij}(\theta_2)}. \end{aligned} \tag{101}$$

Note that $p(\theta_2|\theta_1) < 0$ if $\theta_2 < \theta_1$ (i.e., our "true" time-dilation process is a valid subordinator) as long as our prior model $q(\theta_2|\theta_1)$ is a valid subordinator.

Substituting Eq.(101) into Eq.(100) (and flipping the sign to convert a maximization problem to a minimization problem), we obtain the following function $U(\xi^{(2)})$ (sometimes referred to as a potential function):

$$U(\xi^{(2)}) = \int d\theta_1\, p(\theta_1)\, \log Z_2(\theta_1, \xi^{(2)}) - \sum_j \xi_j^{(2)} C_{2j} \tag{102}$$

The problem of computation of the Lagrange multipliers $\xi_j^{(2)}$ is now reduced to minimizing Eq.(102), which again amounts to a convex optimization problem in dimension equal to the number of option quotes for maturity t_2.

[36] The conditional KL cross entropy is a measure of the difference between two conditional transition probabilities, averaged over the position of the initial point; see, e.g., Cover & Thomas (2006).

For a multi-period setting with more than two nodes on a time line, the above scheme is applied recursively. Let $t_1, t_2, \ldots, t_N$ be nodes on the time line. We first solve the problem for the pair t_1, t_2 as described above. Using these results, we next calculate marginal probabilities $f(\theta_2)$ using the Chapman-Kolmogorov equations. Now the problem for the pair of times t_2, t_3 is treated in the exact same manner as above. We then move to the pair t_3, t_4, etc. As a result, we end up with an implied discrete-valued process for θ_t on a discrete timeline $t_1, t_2, \ldots, t_N$. Derivatives pricing with this framework can be done using the standard backward induction method.

Acknowledgments

We thank Peter Carr, Gene Cohler and Leonid Malyshkin for useful discussions, and Nic Trainor for editing the manuscript.

References

Arnsdorf, M, & Halperin, I. 2007. BSLP: Bivariate Spread-Loss Model for Portfolio Credit Derivatives, *Journal of Computational Finance* **12**, 77-107; `http://arxiv.org/pdf/0901.3398.pdf`.

Bielecki, T.R., Crepey, S., & Herbetsson, A. 2009. Markov Chain Models of Portfolio Credit Risk, in *Handbook of Credit Derivatives*, eds. A. Lipton and A. Rennie, Oxford University Press (2009).

Britten-Jones, M. & Neuberger, A. 2000. Option Prices, Implied Price Processes, and Stochastic Volatility, *Journal of Finance* **55**, 839–866.

Carr, P., Gabaix, X., & Wu, L. 2011. Linearity-Generating Processes, Unspanned Stochastic Volatility, and Interest-Rate Option Pricing, working paper.

Carr, P., Geman, H., Madan, D., & Yor, M. 2003. Stochastic Volatility for Lévy Processes, *Mathematical Finance* **13**(3), 345-382.

Carr, P., & Wu, L. 2004. Time-Changed Lévy Processes and Option Pricing, *Journal of Financial Economics* **71**, 113-141.

Cerrato, M., Lo, C.C. & Skindilias, K. 2011. Adaptive Continuous Time Markov Chain Approximation Model to General Jump-Diffusion, working paper.

Colin-Dufresne, P. & Goldstein, R.S. 2002. Do Bonds Span the Fixed Income Market? Theory and Evidence for Unspanned Stochastic Volatility, *Journal of Finance* **57**(4), 1685-1730.

Cover, T.M., & Thomas, J.A. 2006. *Elements of Information Theory*, and ed., Wiley 2006.

Gabaix, X. 2007. Linearity-Generating Processes: a Modeling Tool Yielding Closed Forms for Asset Prices, working paper, New York University.

Gross, D. & MIller, D.R. 1984. The Randomization Technique as a Modeling Tool and Solution Procedure for Transient Markov Processes, *Operations Research* **32**(2), 343-361.

Halperin, I. 2009. Implied Multi-Factor Model for Bespoke CDO Tranches and Other Portfolio Credit Derivatives, *Journal of Credit Risk* **6** (3), 3-52; http://arxiv.org/pdf/0910.2696.pdf.

Harrison, J. & Pliska, S. 1981. Martingales and Stochastic Integrals and in Theory of Continuous Trading, *Stochastic Processes and their Applications* **11**, 215–260.

Haverkort, B.R. 2001. Markovian Models for Performance and Dependability Evaluation, *FMPA 2000* ed. E. Brinksma et. al., Springer-Verlag (2001).

Feller, W. 1968. An Introduction to Probability Theory and Its Applications, vol 1, Wiley (1968).

Kharoufeh, J.P. 2011. Level-dependent quasi-birth-and-death processes, in *Wiley Encyclopedia of Operations Research and Management Science*, eds. J. Cochran, T. Cox, P. Keskinocak, J.P. Kharoufeh and J.C. Smith, John Wiley & Sons, New York, NY.

Lipton, A., & Sepp, A. 2011. Filling the Gaps, *RISK* October 2011, 78-83.

Moler, C. & van Loan, C.F. 2003. Nineteen Dubious Ways to Compute the Exponential of a Matrix, Twenty-Five Years Later, *SIAM Review* **45**(1), 1-46.

Ren, Y., Madan, D., & Qian, M.Q. 2007. Calibrating and Pricing with Embedded Local Volatility Models, *RISK* September 2007, 138-143.

Piterbarg, V. 2006. Markovian projection method for volatility calibration, available at http://ssrn.com/abstract=906473.

Puzanova, N. 2011. A Hierarchical Model of Tail Dependent Asset Returns For Assessing Portfolio Credit Risk, available at http://www.bundesbank.de/Redaktion/EN/Downloads/Publications/Discussion_Paper_2/2011/2011_12_30_dkp_16.pdf?__blob=publicationFile.

Rossi, A. 2002. The Britten-Jones and Neuberger Smile-Consistent with Stochastic Volatility Option Pricing Model: a Further Analysis, *Int. Journal of Theor. Appl. Finance* **5**(1), 1-31.

Sidje, R.B , Stewart, W.J. 1999. A numerical study of large sparse matrix exponentials arising in Markov chains *Computational Statistics & Data Analysis* **29**(3), 345–368.

Toivanen, J. 2010. A Componentwise Splitting Method for Pricing American Options Under the Bates Model. *Pages 213–227 of: Applied and Numerical Partial Differential Equations.* Computational Methods in Applied Sciences, vol. 15. Springer.

Appendices

A 2D Markov Generator on a Nonuniform Grid

In this appendix, we discuss how to construct the 2D Markov chain generator A in Eq.(29) using a nonuniform grid in the Eq.(30).

To simplify notation, we introduce $h_{k,i} = \Delta z_{k,(i,i-1)}, h^+_{k,i} = \Delta z_{k,(i+1,i)},\ k = 1,2,\ i = 0,\dots,p_k$, where p_1, p_2 are the upper boundary of our discrete grid in the first and second dimensions. The central difference approximation of the second derivatives reads

$$\left.\frac{\partial^2 V}{\partial z_1^2}\right|_{ij} = \delta^-_{1,i} V_{i-1,j} + \delta^0_{1,i} V_{i,j} + \delta^+_{1,i} V_{i+1,j} + O\left(h^2_{1,i})\right) + O\left((h^+_{1,i})^2)\right) + O\left(h_{1,i} h^+_{1,i})\right), \tag{103}$$

$$\left.\frac{\partial^2 V}{\partial z_2^2}\right|_{ij} = \delta^-_{2,j} V_{i,j-1} + \delta^0_{2,j} V_{i,j} + \delta^+_{2,j} V_{i,j+1} + O\left(h^2_{2,i})\right) + O\left((h^+_{2,j})^2)\right) + O\left(h_{2,j} h^+_{2,j})\right),$$

where

$$\delta^-_{k,i} = \frac{2}{h_{k,i}(h_{k,i} + h^+_{k,i})}, \quad \delta^0_{k,i} = -\frac{2}{h_{k,i} h^+_{k,i}}, \quad \delta^+_{k,i} = \frac{2}{h^+_{k,i}(h_{k,i} + h^+_{k,i})}.$$

At the boundaries these coefficients are $\delta^-_{k,i} = 0,\ i = 1$ and $\delta^+_{k,i} = 0,\ i = p_k$.

For the mixed derivative we take noncentral differences to preserve nonnegativity

$$\left.\frac{\partial^2 V}{\partial z_1 \partial z_2}\right|_{ij} = \sum_{m=j-1}^{j+1} \left[\gamma^+_{i,m} V_{i+1,m} + \gamma^0_{i,m} V_{i,m} + \gamma^-_{i,m} V_{i-1,m}\right] + R_{ij}.$$

For $\rho \geq 0$ one has $R_{ij} = O(h_{1,i} h^+_{2,j}) + O(h_{1,i} h^+_{1,i})$ and

$$\begin{aligned}
&\gamma^-_{i,j-1} = \frac{1}{h_{1,i} h_{2,j}}, \quad \gamma^-_{i,j} = -\gamma^-_{i,j-1} - \gamma^-_{i,j+1}, \quad \gamma^-_{i,j+1} = \frac{1}{(h_{1,i} + h^+_{1,i}) h^+_{2,j}}, \\
&\gamma^0_{i,j-1} = -\frac{1}{h_{1,i} h_{2,j}}, \quad \gamma^0_{i,j} = -\gamma^0_{i,j-1} - \gamma^0_{i,j+1}, \quad \gamma^0_{i,j+1} = -\frac{1}{h^+_{1,i} h^+_{2,j}}, \\
&\gamma^+_{i,j-1} = 0, \quad \gamma^+_{i,j} = -\gamma^+_{i,j+1}, \quad \gamma^+_{i,j+1} = \frac{h_{1,i}}{h^+_{1,i}\left(h_{1,i} + h^+_{1,i}\right) h^+_{2,j}}.
\end{aligned}$$

For $\rho < 0$ we find $R_{ij} = O(h_{1,i}h^+_{2,j}) + O(h^+_{1,i}h^+_{2,j}) + O(h_{1,i}h^+_{1,i})$ and

$$\begin{aligned}
&\gamma^-_{i,j-1} = 0, \quad \gamma^-_{i,j} = -\gamma^-_{i,j+1}, \quad \gamma^-_{i,j+1} = -\frac{h^+_{1,i}}{h_{1,i}\left(h_{1,i} + h^+_{1,i}\right)h^+_{2,j}}, \\
&\gamma^0_{i,j-1} = \frac{1}{h_{2,j}h^+_{1,i}}, \quad \gamma^0_{i,j} = -\gamma^0_{i,j-1} - \gamma^0_{i,j+1}, \quad \gamma^0_{i,j+1} = \frac{1}{h_{1,i}h^+_{2,j}}, \\
&\gamma^+_{i,j-1} = -\frac{1}{h_{2,j}h^+_{1,i}}, \quad \gamma^+_{i,j} = -\gamma^+_{i,j-1} - \gamma^+_{i,j+1}, \quad \gamma^+_{i,j+1} = -\frac{1}{\left(h_{1,i} + h^+_{1,i}\right)h^+_{2,j}}.
\end{aligned}$$

Using this in Eq.(34) and regrouping terms, we obtain

$$\left(\mathcal{L}V(z)\right)_{ij} = \sum_{k,m=\{-1,0,1\}} a_{ij|i+k,j+m}V_{i+k,j+m},$$

where the following notation is used :

$$\begin{aligned}
a_{ij|i+1,j} &= \frac{1}{2}s_{ij}\sigma_1\left(\sigma_1\delta^+_{1,i} + \rho\sigma_2\gamma^+_{i,j}\right), & a_{ij|i-1,j} &= \frac{1}{2}s_{ij}\sigma_1\left(\sigma_1\delta^-_{1,i} + \rho\sigma_2\gamma^-_{i,j}\right) \\
a_{ij|i,j+1} &= \frac{1}{2}s_{ij}\sigma_2\left(\sigma_2\delta^+_{2,j} + \rho\sigma_1\gamma^0_{i,j+1}\right), & a_{ij|i,j-1} &= \frac{1}{2}s_{ij}\sigma_2\left(\sigma_2\delta^-_{2,j} + \rho\sigma_1\gamma^0_{i,j-1}\right) \\
a_{ij|ij} &= -\frac{1}{2}s_{ij}\left(\sigma_1^2\delta^0_{1,i} + \rho\sigma_1\sigma_2\gamma^0_{i,j} + \sigma_2^2\delta^0_{2,j}\right), \\
a_{ij|i+1,j+1} &= \rho\sigma_1\sigma_2 s_{ij}\gamma^+_{i,j+1}, & a_{ij|i-1,j-1} &= \rho\sigma_1\sigma_2 s_{ij}\gamma^-_{i,j-1}, \\
a_{ij|i+1,j-1} &= \rho\sigma_1\sigma_2 s_{ij}\gamma^+_{i,j-1}, & a_{ij|i-1,j+1} &= \rho\sigma_1\sigma_2 s_{ij}\gamma^-_{i,j+1}.
\end{aligned}$$

where $s_{ij} = [s(Z_t)]_{ij}$.

To construct a valid Markov generator, we have to make sure that all off-diagonal elements are positive and all rows sum to zero. It is also necessary to obey the following property: if f^n and f^{n+1} are the state vectors at time moment n and $n+1$, and A is the transition matrix (i.e., $f^{n+1} = Af^n$), then to preserve positiveness of f, the matrix A must be diagonally dominant. When applied to the above equations, these three conditions give rise to tricky dependencies between the grid steps $h_{1,i}, h^+_{1,i}, h_{2,i}, h^+_{2,i}$, which could be hard to reconcile with the usual approach of building a nonuniform grid based on expected values of model parameters. One possible approach that escapes the need to deal with exceedingly complicated constraints on the grid steps could be to use a nonuniform grid in one direction and a uniform grid in the other direction.[37]

[37] This is similar to building space grids as a part of an FD approach to solving 2D PDEs that determine the option price under some stochastic volatility models. For more details, see, e.g., Toivanen (2010).

B Random Time Change of a Continuous-Time Markov Chain

Consider a homogeneous Markov chain with a diagonalizable generator A such that

$$A = UDU^{-1} \ \ , \ \ D \equiv \mathrm{diag}(d_1, d_2, \ldots, d_N),$$

where the eigenvalues $\{d_i\}$ are assumed to be in a descending order. The matrix U consists of eigenvectors stored column-wise. For a finite-time transition matrix, we then have

$$P(t,T) = Ue^{(T-t)D}U^{-1}.$$

Next we make the transition matrix stochastic by introducing the random time change $t \to T_t$ driven by a nonnegative stochastic process (activity rate) Y_t such that

$$T_t = \int_0^t Y_s ds. \tag{104}$$

By viewing T_t as a "true" "business" or "trading" time as opposed to the calendar time t, the transition matrix becomes stochastic as it now depends explicitly on Y_t:

$$P_X(t,T) = Ue^{D\int_t^T Y_s ds}U^{-1}. \tag{105}$$

Consider now a Markov chain obtained by conditioning on a path of Y_t. By taking the derivative of Eq.(105) with respect to t and comparing with the Kolmogorov equation

$$\frac{\partial P_X(t,T)}{\partial t} = -A_X(t)P_X(t,T),$$

we see that the conditional on the realization of the path of Y_t, our process is given by an inhomogeneous Markov chain with generator

$$A_X(t) = Y_t UDU^{-1} = Y_t A.$$