C. R. Acad. Sci. Paris, Ser. I ••• (••••) •••–•••

Contents lists available at ScienceDirect

C. R. Acad. Sci. Paris, Ser. I

www.sciencedirect.com

Partial differential equations/Numerical analysis

# A new method for solving Kolmogorov equations in mathematical finance

## *Une nouvelle méthode pour résoudre les équations de Kolmogorov en finance mathématique*

Philippe G. LeFloch [a], Jean-Marc Mercier [b]

[a] *Laboratoire Jacques-Louis-Lions & Centre national de la recherche scientifique, Université Pierre-et-Marie-Curie, 4, place Jussieu, 75252 Paris cedex 05, France*
[b] *MPG Partners, 136, bd Haussmann, 75008 Paris, France*

ARTICLE INFO

*Article history:*
Received 30 June 2016
Accepted after revision 15 May 2017
Available online xxxx

Presented by Olivier Pironneau

ABSTRACT

For multi-dimensional Fokker–Planck–Kolmogorov equations, we propose a numerical method which is based on a novel localization technique. We present extensive numerical experiments that demonstrate its practical interest for finance applications. In particular, this approach allows us to treat calibration and valuation problems, as well as various risk measure computations.



RÉSUMÉ

Nous proposons une nouvelle méthode numérique, utilisant une technique de localisation originale, pour résoudre des équations de Fokker–Planck–Kolmogorov multi-dimensionnelles. Nous présentons des tests numériques extensifs qui démontrent l'intérêt pratique de cette approche pour les applications en finance. En particulier, cette approche nous permet de traiter les problèmes de calibration et de valorisation, ainsi que le calcul de mesures de risque variées.



### Version française abrégée

Nous présentons une nouvelle méthode de calcul numérique pour les équations de Kolmogorov provenant de la finance mathématique. Cette méthode permet de résoudre le problème de la valorisation et le problème de la calibration de manière homogène. Tout d'abord, nous reformulons les équations de Kolmogorov grâce à un changement de variable motivé par la théorie du transport optimal. Un deuxième ingrédient essentiel dans notre méthode est la construction d'une grille de calcul dont la discrépance est optimale : les points de la grille sont obtenus à partir d'une suite de points de type (quasi-)Monte-

*E-mail addresses:* contact@philippelefloch.org (P.G. LeFloch), jean-marc.mercier@mpg-partners.com (J.-M. Mercier).

http://dx.doi.org/10.1016/j.crma.2017.05.003
1631-073X/

Please cite this article in press as: P.G. LeFloch, J.-M. Mercier, A new method for solving Kolmogorov equations in mathematical finance, C. R. Acad. Sci. Paris, Ser. I (2017), http://dx.doi.org/10.1016/j.crma.2017.05.003

Carlo, qui est transformée en une suite optimale par un système dynamique adéquat. Nos résultats numériques mettent en évidence la pertinence de cette approche pour des applications pratiques en finance.

## 1. Introduction and main strategy

### 1.1. Fokker–Planck–Kolmogorov equations

We denote by $\mu = \mu(t,x)$ ($t \geq 0$, $x \in \mathbb{R}^D$) the probability density measure of a stochastic process, in which the number of dimensions $D$ is usually the number of risk sources (also called underlyings) in the applications in mathematical finance. We assume that this process defines, up to a variable change, a martingale process governed by a Fokker–Planck equation

$$\partial_t \mu - \mathcal{L}\mu = 0, \qquad \mathcal{L}\mu := \sum_{1 \leq i,j \leq D} \partial_i \partial_j \big(a_{ij}(t,.)\mu\big), \tag{1}$$

where $A(t,x) := \big(a_{ij}(t,x)\big)_{i,j=1,\dots,D} \in \mathbb{R}^{D\times D}$ is a (possibly unknown) field of symmetric positive definite matrices. We also consider the adjoint of the above equation, that is, a backward Kolmogorov equation with unknown $P = P(t,x) \in \mathbb{R}^M$ (defined for $0 \leq t \leq T$ and $x \in \mathbb{R}^D$)

$$\partial_t P - \mathcal{L}^* P = 0, \quad P(T,\cdot) = P_T \in \mathbb{R}^M, \tag{2}$$

where $\mathcal{L}^* P := \sum_{1\leq i,j\leq D} a_{ij}(t,.)\partial_i \partial_j P$ denotes the adjoint operator and a final Cauchy data $P_T = P_T(x)$ is prescribed at some (future) time $T > 0$. This equation typically is enjoyed by flows (of cash or asset) in mathematical finance. We can also treat nonlinear versions of (2), such as the optimal-stopping problem in which the mapping $\mathcal{S} = \mathcal{S}(t,x,P) : \mathbb{R}^{D+1+M} \to \mathbb{R}^E$ is interpreted as a "strategy":

$$\text{either } \partial_t P - \mathcal{L}^* P = 0 \text{ and } \mathcal{S}(t,x,P) \geq 0 \quad \text{hold, or else} \quad \partial_t P - \mathcal{L}^* P \geq 0 \text{ and } \mathcal{S}(t,x,P) = 0. \tag{3}$$

In this context, the following two rather distinct problems arise.

#### 1.1.1. The calibration problem

This problem consists in solving for the probability measures $\mu(t,\cdot)$ satisfying (1) and we distinguish between three setups. First of all, the *SDE-driven problem* corresponds to a case where the matrix $A = A(t,x)$ is known, for instance when the equation (1) arises from a stochastic process $t \mapsto X_t \in \mathbb{R}^D$ driven by a stochastic differential equation (SDE) $\mathrm{d}X_t = \sigma(t,X_t)\,\mathrm{d}W_t$, where $t \mapsto W_t$ is an uncorrelated Brownian motion and, in this case, one has $A = \frac{1}{2}\sigma^{\mathrm{T}}\sigma$. In the second setup of interest, referred to as the *model-free calibration*, the matrix $A = A(t,x)$ itself is an unknown of the problem and is determined such that $\mu(t,\cdot)$ satisfies a list of integral constraints at some (future) times $T_1, T_2, \dots, T_I > 0$:

$$\int_{\mathbb{R}^D} P_i(\cdot)\,\mu(T_i,\cdot) = C_i, \qquad i = 1,\dots,I, \tag{4}$$

in which $P^i$ and $C_i$ (referred to as observables) are prescribed. From a local volatility model "à la Dupire", several methods are available in order to handle this (undetermined) problem and we refer to [4] for a review of the most commonly used methods. In a third possible setup, referred to as a *scenario*, the map $t \mapsto \mu(t,\cdot)$ is in fact prescribed.

#### 1.1.2. The valuation problem

The probability measure $\mu(t,\cdot)$ being known at this stage, the second problem of interest consists in computing the operator $\mathcal{L}$ (for the model-free calibration) and solving the equation (3), which leads one to the output measure computed from the surface $\{t,x,P(t,x)\}$. There exist several numerical methods devoted to (2) and (3); see for instance [2] for a survey of Monte-Carlo methods (which are able to handle *linear* problems like (2) only) and PDE (partial differential equation) lattice-based methods (which are able to handle also nonlinear problems (3)). We emphasize that the whole "surface" $\{t,x,P(t,x)\}$ must be computed in order to solve the nonlinear problem (3). Furthermore, even for linear problems like (2), this surface also contains the necessary information that is required for risk measurement applications (of operational or regulatory nature).

### 1.2. Application to mathematical finance

For instance, one could treat the scalar strategy $\mathcal{S}(t,x,P) = P - (x-K)^+$ (with $Q^+ := \max(Q,0)$) corresponding to an optimal strategy for options having strike $K \in \mathbb{R}$. The vector-valued function $P = P(t,x)$ in general determines the *fair* value of contracts under the strategy $\mathcal{S}$, when the underlying stochastic process is worth $x \in \mathbb{R}^D$ at a future time $t \geq 0$. The method we now propose provides a general framework since all the stochastic models usually defined by prescribing specific stochastic processes (such as normal, log-normal, stochastic volatility, local volatility or local correlation models) fit into the Fokker–Planck setting (1). Moreover, the applications in mathematical finance involve constraints of the form (4), which we can handle here. As stated above, all classical risk measurements can be deduced from the knowledge of the solutions to

Please cite this article in press as: P.G. LeFloch, J.-M. Mercier, A new method for solving Kolmogorov equations in mathematical finance, C. R. Acad. Sci. Paris, Ser. I (2017), http://dx.doi.org/10.1016/j.crma.2017.05.003

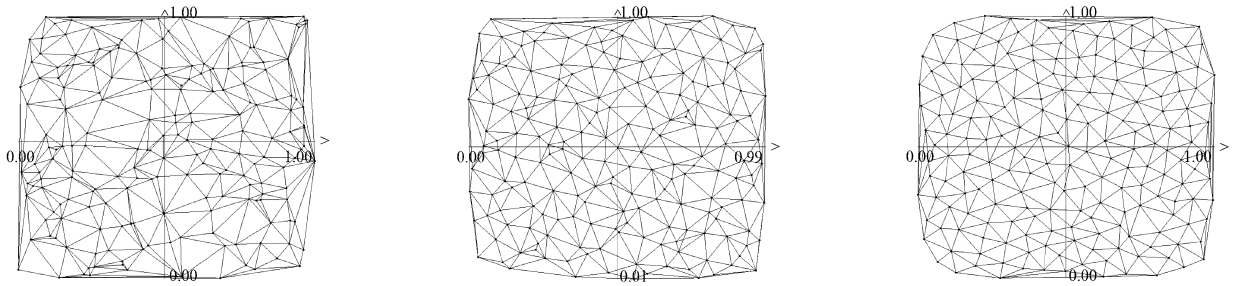


**Fig. 1.** Different grids with Delaunay meshes.

(3). For instance, for operational measures, $P^t := \int_{\mathbb{R}^D} P(t,\cdot)\mu(t,\cdot)$ corresponds to the *fair* value if $t = 0$ (and future value if $t > 0$) of contracts viewed from today ($t = 0$), since $\int_{\mathbb{R}^D} \nabla P(t,\cdot)\mu(t,\cdot)$ computes its hedge, where $\nabla := \left(\frac{\partial}{\partial x_d}\right)_{d=1,\dots,D}$ is the gradient operator. Similarly for regulatory type measures, a Value At Risk (VaR) or a Credit Value Adjustment (CVA) can be expressed from the knowledge of the function $t \mapsto P(t,\cdot)$.

### 1.3. Purposes and main ideas

The CoDeFi algorithm, as we call it, is based on several novel numerical techniques, which lead us to a framework for handling the calibration problem as well as the valuation problem possibly in high dimensions. Our algorithm is based on two main ingredients. First of all, a **localization principle** is defined from a change of variable which localizes the system of equations (3) to the unit cube, denoted $\Lambda = [0,1]^D$. To properly introduce this change of variable, we recall a standard result from the theory of optimal transport (cf. Villani [10]), going back to Brenier [1], allowing one to regard the quantile of a probability measure as a change of variable of the form:

$$\mu(t,\cdot) = S(t,\cdot)_{\#}m, \qquad S(t,\cdot) = \nabla h : \Lambda \mapsto \mathbb{R}^D, \quad h \text{ convex}, \tag{5}$$

where $S(t,\cdot)_{\#}m$ stands for the pull-back of the Lebesgue measure $m$ on $\Lambda$. For our purpose, we think of $S(t,\cdot)$ as a map inducing a relevant change of variable for Monte-Carlo methods. Together with a random generator, it allows us to sample any process at time $t$, by writing

$$\mathbb{E}^t(P,\mu) := \int_{\mathbb{R}^D} P(\cdot)\mu(t,\cdot) \simeq \frac{1}{N}\sum_{1\le n\le N} P\circ S(t,Y_n), \qquad n = 1,\dots,N, \tag{6}$$

where $Y = \{Y_n \in \Lambda\}_{n=1,\dots,N} \in \mathbb{R}^{N\times D}$ is made of random vectors. Our localization principle then consists in considering the following version of (3) (*transported* into the unit cube):

$$\text{either } \left(\partial_t P - \mathcal{L}^* P\right)\circ S = 0 \text{ and } \mathcal{S}(t,S,P\circ S) \ge 0 \text{ hold, or else } \left(\partial_t P - \mathcal{L}^* P\right)\circ S \ge 0 \text{ and } \mathcal{S}(t,S,P\circ S) = 0. \tag{7}$$

One important incentive for using the change of variables above is to transform the Fokker–Planck equation (1) and Kolmogorov equation (2) into *self-adjoint* problems. Indeed, one can check that [6]

$$\partial_t S = -\nabla\cdot\Big((\nabla S)^{-1}A\circ S(\nabla S)^{-1}\nabla\Big)S, \quad \partial_t(P\circ S) = \nabla\cdot\Big((\nabla S)^{-1}A\circ S(\nabla S)^{-1}\nabla\Big)(P\circ S), \tag{8}$$

where $\nabla\cdot$ denotes the divergence operator. In particular, for the model-free calibration, we observe that, once $S(t,\cdot) = \nabla h(t,\cdot)$ is determined, then $A\circ S := (\partial_t h)(\nabla^2 h)^{-1}$ provides a solution to the latter equation in (8). Notice that we are assuming in this Note that the Jacobian $\nabla^2 h$ is strictly positive definite. However, we can also cope with degenerate cases, corresponding to singular probability measures $\mu(t,\cdot)$, see [6] for details and, for earlier related work by the authors, see [5].

Our second main ingredient is a **meshfree technique** (cf. [3] for an introduction) in order to solve (7) using (8). Our motivation is that random sampling methods (such as Monte-Carlo ones (6)) are not sensitive to increasing the dimension of the problem. Our PDE-based method will use the sampling vectors $Y$ as a mesh. In the rest of this Note, we build upon these two main ideas and we provide further details on the CoDeFi algorithm.

## 2. Grid generation, calibration and valuation algorithms

### 2.1. Grid generation

Throughout, we have two essentials parameters, that is, the number $D$ of dimensions (i.e. the number of underlyings or risk sources in financial applications) and the number $N$ of grid points. Let us discuss first the choice of the sampling set $Y = \{Y_n \in \Lambda\}_{n=1,\dots,N} \in \mathbb{R}^{N\times D}$, used as a mesh for the PDE computation. The CoDeFi algorithm is primarily a Monte-Carlo method and, in fact, could rely on any sampling set of the uniform law over $\Lambda$ as a grid. For instance, Fig. 1 displays the plot

Please cite this article in press as: P.G. LeFloch, J.-M. Mercier, A new method for solving Kolmogorov equations in mathematical finance, C. R. Acad. Sci. Paris, Ser. I (2017), http://dx.doi.org/10.1016/j.crma.2017.05.003

of a two-dimensional grid with 200 points, based on a Delaunay mesh and using the Mersenne Twister generator mt19937 (cf., for instance, [7]). Another alternative is to use quasi-random sequences: Fig. 1 displays a grid with 200 points using a Sobol low-discrepancy generator (cf. for instance [9]).

On the other hand, since the properties of the underlying mesh is essential in PDE methods, we propose here to use sequences that we call *optimal discrepancy sequences,* in order to generate the grid. See the right-most plot in Fig. 1. Such sequences reach the rate of convergence $\epsilon = \mathcal{O}(1/N)\%$ for sufficiently regular functions $P$ in (6). Our sequences are determined by considering (for instance) the discrepancy functional $\mathcal{D}(Y) = \sum_{i,j=1,\dots,N} \ln|Y_i - Y_j|_2$ and the dynamical system $\frac{\mathrm{d}}{\mathrm{d}t} Y_i = \sum_{j=1,\dots,N} \frac{Y_j - Y_i}{|Y_j - Y_i|_2}$, corresponding to a steepest descent algorithm for this functional. The corresponding discrete scheme converges at exponential rate (up to a rescaling factor) toward (what we call) an optimal discrepancy sequence for the unit ball $B(0,1) \subset \mathbb{R}^D$. The numerical results presented in this Note were obtained by transporting such sequences into the unit cube $[0,1]^D$.

### 2.2. The model-free calibration algorithm

Our calibration algorithm search for a probability measure $\mu(t,\cdot)$ and, by standard optimal transport results, this is equivalent to finding the quantile $S(t,\cdot)$ in (5). When the matrix $A = A(t,x)$ is prescribed, we can solve directly the left-hand-side equation in (8); therefore, we can now focus on the model-free calibration problem, i.e. the case when the quantile is required to satisfy the constraints (4).

Let $\mu_0(t,\cdot)$ be a smooth "prior" probability measure (in practice, we use normal or log-normal processes as priors) and let $S_0(t,\cdot) = \nabla h_0(t,\cdot)$ be its quantile, which is known explicitly. From the numerical standpoint, we solve the calibration problem by finding a matrix $S(t) = \nabla_Y h(t) \in \mathbb{R}^{ND}$, where $\nabla_Y \in \mathbb{R}^{ND\times N}$ denotes a discretization of the nabla operator $\nabla$, satisfying for each relevant time $T_i$ (cf. our notation in (4)):

$$h(T_i) := \arg\inf_{h\in\mathbb{R}^N} \sum_{n=1,\dots,N} |\nabla_Y h - \nabla_Y h_0|_2^2, \qquad \frac{1}{N} \sum_{n=1,\dots,N} P^i(T_i, \nabla_Y h) = C^i, \quad i = 1,\dots,I. \tag{9}$$

This optimization problem with constraints does not guarantee that the solution $h$ will be convex, that is, $\left(\nabla_Y^2 h(t)\right)_{n=1,\dots,N}$ need not be a field of symmetric positive definite matrices. However, $h$ can always be made convex by a permutation; that is, we also need here to apply a *reordering algorithm* in order to compute the relevant permutation using optimal transport and we refer to [6] for the details. Consequently, starting from a set of calibrated times $T_i$, standard bootstrap and interpolation arguments allow us to recover the whole surface $t \mapsto S(t) \in \mathbb{R}^{ND}$ for any desired times, as required. Note, moreover, that the constraints (4) may not always hold true but, since we solve a constrained minimization problem, the calibration step remains always stable in practice. We conclude this section by pointing out that the constraints (4) provide us with a quite general setup. In particular, we can use a set of constraints to match classical statistical measures exactly:

- *expectations*: we can match any of the $D$ marginal expectations $T \mapsto F_d(T) := \int x_d \mu(T,\cdot)$ of any distributions using the constraint $F_d(T,x) = x_d$ in (4);
- *variances*: we can match any of the $D$ marginal variances $T \mapsto V_d(T)$ of any distributions using the constraints $V_d(T,x) = \big(x_d - F_d(T)\big)^2$ in (4);
- *correlations*: we can match any correlation matrix $T \mapsto C_{d_1,d_2}(T)$ with $d_1, d_2 = 1,\dots,D$, using the constraints $C_{d_1,d_2}(T,x) = \frac{(x_{d_1} - F_{d_1}(T))(x_{d_1} - F_{d_2}(T))}{V_{d_1}(T) V_{d_2}(T)}$, $1 \le d_1 < d_2 \le D$ in (4).

### 2.3. The valuation algorithm

It remains to present the discretization of the transported Kolmogorov equation (i.e. the right-hand-side equation in (8)). This valuation phase arises once the calibration step has been performed, hence the discrete quantile $S(t) = \nabla_Y h(t) \in \mathbb{R}^{ND}$ is known at this stage: it is either computed as described in Section 2.2 or prescribed a priori. We suppose also at this stage that the matrix $A \circ S(t)$ is also known. In particular, we can pick-up $A \circ S(t) := \partial_t h (\nabla_Y^2 h)^{-1}$ in the model-free calibration. Denote $S(t) = \nabla_Y h$ with $h \simeq h(t, Y^n)_{n=1,\dots,N} \in \mathbb{R}^N$ numerically convex (in the sense $(\nabla_Y^2 h)_{n=1,\dots,N}$ is symmetric positive definite at each point of the mes) and denotes $P(t) \simeq \Big(P(t, Y_n)\Big)_{n=1,\dots,N} \in \mathbb{R}^{N\times M}$ its numerical approximation. Then writing a semi-discrete scheme leads us to a scheme of the form

$$\frac{\mathrm{d}}{\mathrm{d}t} P = -\Big(\nabla_Y \cdot (\nabla_Y^2 h)^{-1} A \circ S (\nabla_Y^2 h)^{-1} \nabla_Y\Big) P = -B(t) P, \quad P(T) \in \mathbb{R}^{N\times M} \text{ being prescribed,} \tag{10}$$

where $\nabla_Y \cdot := \nabla_Y^{\mathrm{T}}$ approximates the divergence operator, $B(t) = C^{\mathrm{T}}(t) C(t) \in \mathbb{R}^{N\times N}$ symmetric positive definite with $C(t) := \sqrt{A \circ S} (\nabla_Y^2 h)^{-1} \nabla_Y \in \mathbb{R}^{ND^2\times N}$. Our scheme is dissipative in the sense that $\frac{\mathrm{d}}{\mathrm{d}t} \|P\|_{\mathbb{R}^{N\times M}}^2 \le 0$, hence a global existence result

Please cite this article in press as: P.G. LeFloch, J.-M. Mercier, A new method for solving Kolmogorov equations in mathematical finance, C. R. Acad. Sci. Paris, Ser. I (2017), http://dx.doi.org/10.1016/j.crma.2017.05.003

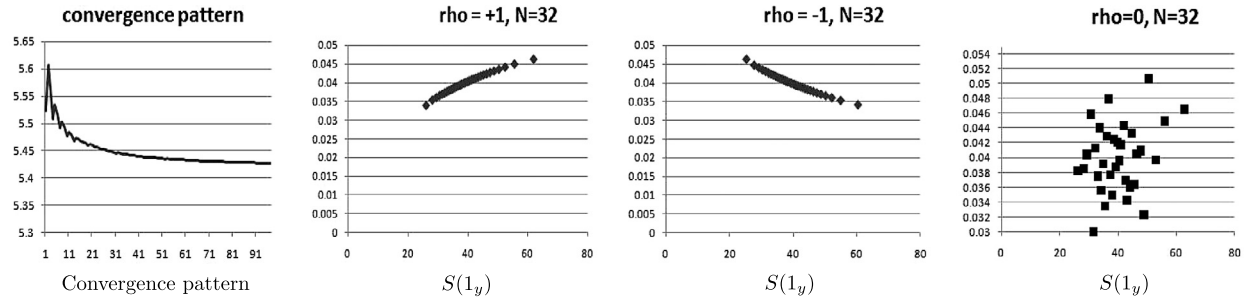


**Fig. 2.** Qualitative and quantitative property of the SDE driven algorithm – Heston case.

**Table 1**
$\rho = 0$: Comparison with the modified Craig–Sneyd method (results provided using Quantlib), grids of size $N \times N$. The theoretical value is 5.435503.

| | CoDeFi $N = 6$ | $N = 9$ | $N = 12$ | Craig–Sneyd $N = 30$ | $N = 60$ | $N = 120$ |
|---|---|---|---|---|---|---|
| Comp. | 5.441709 | 5.43749 | 5.435855 | 5.446631 | 5.4380253 | 5.4361311 |
| Err % | 0.0571% | 0.0183% | 0.0032% | 0.10226% | 0.0232% | 0.00578% |
| Conv. rate | 4.168 | 3.917173 | 4.1603 | 2.0244 | 2.0440 | 2.0384 |

holds true. Let $s \leq t$ two consecutive times of this time grid (recall that time is reversed). Then the solution $P(s) \in \mathbb{R}^N$ is computed accordingly to

$$P(s) = \Pi^{(t,s)} P(t), \quad \Pi^{(t,s)} := \left(\pi_{n,m}^{(t,s)}\right)_{n,m=1,\dots,N} \in \mathbb{R}^{N\times N}, \tag{11}$$

where the matrix $\Pi^{(t,s)}$ (the generator to the discrete transported Kolmogorov equation (11)) is computed explicitly using a $\theta$ scheme. This matrix $\Pi^{(t,s)}$ can be interpreted in a Markov-chaining process setting: $\pi_{n,m}^{(t,s)}$ is the probability that the stochastic process jumps from the state $S_n(t) \in \mathbb{R}^D$ to the state $S_m(t) \in \mathbb{R}^D$. Hence, an important property of this matrix is to be *stochastic*, reflecting the fact that the underlying process defines a *martingale* process. Obviously, this property is obtained after projecting the quantile $t \mapsto S(t, \cdot)$ into a proper space of *martingale* mappings. Note that this *stochastic* property implies the following discrete conservation of expectations for any time, for any solution $P \circ S$ to the transported Kolmogorov equation

$$C = \sum_{n=1,\dots,N} P^i(t, S^n(t)) := \langle P(t), 1_N \rangle_{\mathbb{R}^N}, \quad 0 \leq t \leq T_i. \tag{12}$$

Solutions to the nonlinear problem (3) are approximated adding nonlinear terms to the linear solution (11) at each time, as follows: $P(s) = \max\left(\Pi^{(t,s)} P(t), \mathcal{S}(s, S(s), \Pi^{(t,s)} P(t))\right)$. Hence, once the transition matrices $\Pi^{(t,s)}$ computed, computing (3) is mainly a matter of matrix–matrix multiplication.

## 3. Numerical experiments

### 3.1. SDE-driven algorithm example

Let us investigate our method for a bi-dimensional Heston process $(\kappa, \theta, r, \rho, \xi)$ defined by the SDE $\mathrm{d}X_t = r(t, X_t) + \sigma(t, X_t)\mathrm{d}W_t$, with $r(t,x) := (rx_1, \kappa(\theta - x_2))$, $\sigma(t,x) = \begin{pmatrix} \sqrt{x_2}x_1 & 0 \\ \rho\xi\sqrt{x_2} & \sqrt{1-\rho^2}\xi\sqrt{x_2} \end{pmatrix}$ – a combination between a log-normal process and a CIR (Cox–Ingelson–Ross) process. This stochastic process is well-defined for $2\kappa\theta > \xi^2$, that is the condition for which the CIR process remains positive. The numerical test is as follows: consider the Heston process defined by $(\kappa = 0.5, \theta = 0.04, r = 0, \rho = 0, \xi = 0.1)$, with initial data $X_0 = (40, 0.04)$ and the function (call payoff) $P(1_y, x) := (x_1 - K)^+$, with $K = 36$ (where $1_y$ stands for 'one year'). We recall that there exists a closed formula to value this particular Heston Kolmogorov equation, giving benchmarks. These data are input in our algorithm, that computes the left-hand equation in (8) as a discrete quantile $t \mapsto S(t) \in \mathbb{R}^{N\times 2}$. For instance, the three right-most figures in Fig. 2 show $S(1_y)$ for three different values of the correlation $\rho = -1, 0, +1$, $N = 32$ (hence including the two degenerate cases $\rho = \pm 1$). With this quantile, the Kolmogorov equation (right-hand-side equation in (8)) is solved using the scheme (10), and our result (line "Comp." in Table 1) is compared with the modified Craig–Sneyd method, using $N$ time steps of a $N \times N$ grid, computing the observed convergence rate of both methods in the last column (in this table, the total number of points being $N \times N$). This table shows that our algorithm performs particularly well.

### 3.2. Model-free calibration algorithm – examples

The first example is one dimensional: consider Table 2, containing (few but) real constraints, corresponding to quotes of call options written over index $SX5E$ having maturity 3 months (denotes as $3_m$), that is $P^i(T_i, x) = (x - K_i)^+$, $T_i = 3_m$,

Please cite this article in press as: P.G. LeFloch, J.-M. Mercier, A new method for solving Kolmogorov equations in mathematical finance, C. R. Acad. Sci. Paris, Ser. I (2017), http://dx.doi.org/10.1016/j.crma.2017.05.003

**Table 2**
Market values of European Call SX5E Mat $3_m$. Spot 3064.03.

| Strike $\alpha_i$ | 0.8 | 0.9 | 0.95 | 0.975 | 1 | 1.025 | 1.05 | 1.1 | 1.2 |
|---|---|---|---|---|---|---|---|---|---|
| Value $C^i$ | 559.2 | 292.6 | 180.7 | 133.6 | 93.76 | 61.59 | 37.99 | 11.34 | 0.31 |

**Table 3**
Price of European Best-of options MAT 10Y.

| MC $N = 1048576$ | $D = 1$ | $D = 4$ | $D = 16$ | $D = 64$ |
|---|---|---|---|---|
| | 0.12578 | 0.35941 | 0.68796 | 1.0166 |
| N = 32 | 0.128275 | 0.340435 | 0.678092 | 0.927398 |
| N = 128 | 0.126521 | 0.349573 | 0.693397 | 0.982611 |
| N = 512 | 0.125921 | 0.359632 | 0.688982 | 1.0144 |

**Table 4**
Call SX5E Mat $3_m$. Spot 3064.03.

| Strike % | Eur. Call values | $N = 16$ | $N = 64$ | $N = 256$ | $N = 1024$ |
|---|---|---|---|---|---|
| 0.8 | 559.224 | 650.54 | 650.54 | 650.54 | 650.54 |
| 1 | 93.76 | 103.16 | 104.19 | 104.33 | 104.37 |
| 1.2 | 0.31 | 0.31 | 0.31 | 0.31 | 0.32 |
| Computational time | | 0.02 s | 0.06 s | 1.52 s | 77 s |

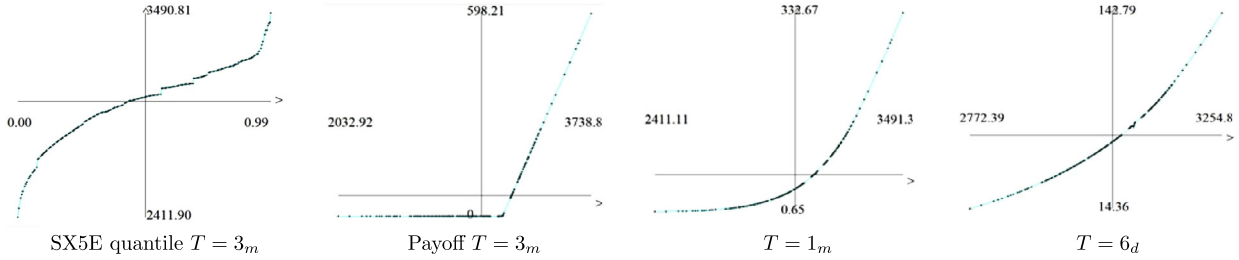


**Fig. 3.** Left: Calibrated SX5E quantile for $N = 256$, $D = 1$. Others: $P(T, S)$ at different retropropagation times, $N = 256$.

$K_i := \alpha_i S(0)$ (see our notation (4)). Then the leftmost Fig. 3 shows a quantile calibrated to these quotes at time $3_m$ using the calibration and the reordering algorithm (which amounts to reorder by increasing values in one dimension).

To illustrate the behavior of this calibration algorithm in large dimensions and for a large number of constraints, we now consider an independent log-normal process $X_d = e^{Z_d/10}$, $d = 1_d$ (one day), where $Z_d$ denotes a standard normal process. Table 3 presents, for $D = 1, 4, 16, 64$, $N = 32, 128, 512$, the computation of the expectation (6) for the function $P(T, x) = (|x|_\infty - K)^+$, with $K = 1$ (called a best-of option), where $|x|_\infty = \sup_{d=1,\dots,D}(|x_d|)$, and $T = 10Y$. Table 3 – line MC – presents reference computations using an $N = 1048576 \times D$ sequence of a pseudo-random Mersenne twister MT19937, which are confident with a relative error estimated at $0,1\% \simeq 1/\sqrt{1048576}$. The others columns present the same computations, but using calibrated sequences $S(T)$, matching all expectations, all variances, and the whole correlation identity matrix, that is 2144 constraints for $D = 64$. Indeed, we noticed that this calibration procedure accelerates the convergence of Monte-Carlo sampling (6), as could be expected.

### 3.3. Valuation algorithms

Our test here, which consists in solving (3), with the same data as presented in Table 2, but using the strategy $\mathcal{S}(t, x, P) = P - (x - K_i)^+$, corresponds to an option exercise. We considered the data set in Table 2 to calibrate the underlying process at time $T = 3_m$, see Fig. 3 at time $T = 3_m$. The qualitative properties of the solution are plotted in a series of three figures, illustrating the retro-propagation step for the special case $K_i = 3064$. The first one, the second one in Fig. 3, represents the initial conditions at time $T = 3_m$. It plots more precisely the surface $\{S_n^{3_m}, P(3_m, S_n^{3_m})\}_{n=1,\dots,256}$. The second one, the third one in Fig. 3, plots the solution at time $t = 1_m$, $\{S_n^{11_m}, P(1_m, S_n^{1_m})\}_{n=1,\dots,256}$. The third one plots the solution at time $t = 6_d$, $\{S_n^{6_d}, P(6_d, S_n^{6_d})\}_{n=1,\dots,256}$, which is the last computation time.

Expectations of the solution at time $T = 6_d$, which corresponds to fair options prices, are presented in Table 4, with different numbers of points $N$ – we recall that $N$ drives the accuracy of the computation – and different strikes $K_i$, but the same as that used for the calibration process. It has already been noted that these numerical schemes are very stable and accurate [8], as confirmed by this table.

Please cite this article in press as: P.G. LeFloch, J.-M. Mercier, A new method for solving Kolmogorov equations in mathematical finance, C. R. Acad. Sci. Paris, Ser. I (2017), http://dx.doi.org/10.1016/j.crma.2017.05.003

## References

[1] Y. Brenier, Polar factorization and monotone re-arrangements of vector-valued functions, Commun. Pure Appl. Math. 44 (1991) 375–417.
[2] M. Broadie, J.B. Detemple, Option pricing: valuation models and applications, Manag. Sci. 50 (2004) 1145–1177.
[3] G.E. Fasshauer, Meshfree methods, in: M. Rieth and, W. Schommers (Eds.), Handbook of Theoretical and Computational Nanotechnology, vol. 2, American Scientific Publishers, 2006.
[4] C. Homescu, Implied volatility surface: construction methodologies and characteristics, preprint, arXiv:1107.1834, unpublished work.
[5] P.G. LeFloch, J.-M. Mercier, Revisiting the method of characteristics via a convex hull algorithm, J. Comput. Phys. 298 (2015) 95–112.
[6] P.G. LeFloch, J.-M. Mercier, Tackling the curse of dimensionality, in preparation.
[7] M. Matsumoto, T. Nishimura, Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator, ACM Trans. Model. Comput. Simul. 8 (1998) 3–30.
[8] J.-M. Mercier, Optimally transported schemes and applications in mathematical finance, notes available at http://www.crimere.com/blog/jean-marc/?p=336, November 2008.
[9] I.M. Sobol, Distribution of points in a cube and approximate evaluation of integrals (in Russian), Ž. Vyčisl. Mat. Mat. Fiz. 7 (1967) 784–802.
[10] C. Villani, Optimal Transport: Old and New, Springer Verlag, 2006.

Please cite this article in press as: P.G. LeFloch, J.-M. Mercier, A new method for solving Kolmogorov equations in mathematical finance, C. R. Acad. Sci. Paris, Ser. I (2017), http://dx.doi.org/10.1016/j.crma.2017.05.003