Physica A 581 (2021) 126231

Contents lists available at ScienceDirect

Physica A

journal homepage: www.elsevier.com/locate/physa

Path integral Monte Carlo method for option pricing

Pietro Capuozzo, Emanuele Panella, Tancredi Schettini Gherardini, Dimitri D. Vvedensky *

The Blackett Laboratory, Imperial College London, London, SW7 2AZ, United Kingdom

ARTICLE INFO

Article history:
Received 8 March 2021
Received in revised form 7 June 2021
Available online 1 July 2021

Keywords:
Black–Scholes
Path integral
Markov chain Monte Carlo
Metropolis–Hastings
Asian options
Non-Gaussian

ABSTRACT

The Markov chain Monte Carlo (MCMC) method, in conjunction with the Metropolis–Hastings algorithm, is used to simulate the path integral for the Black–Scholes–Merton model of option pricing. After a brief derivation of the path integral solution of this model, we develop the MCMC method by discretizing the path integral on a time lattice and evaluating this discretized form for various scenarios. Particular attention is paid to the existence of autocorrelations, their decay with the number of sweeps, and the resulting estimate of the corresponding errors. After testing our approach against closed-form solutions, we demonstrate the utility and flexibility of our method with applications to non-Gaussian models.

1. Introduction

Options are among the most traded financial instruments in the world, with a gross market value of over-the-counter derivatives rising to $15.5 trillion during the first half of 2020 [1]. For this reason, developing models and computational methods for option prices is crucial for understanding price movements, creating new financial instruments, and assessing the risk associated with particular options. The best known approach to option pricing is based on the Black–Scholes–Merton (BSM) model [2–5]. Despite well-known deficiencies in some of its underlying assumptions, this model remains widely used, due in large part to the availability of closed-form solutions [5–10] for the pricing of several types of options and derivatives.

The Black–Scholes model is based on the principle of hedging and eliminating risks associated with the volatility of underlying assets and stock options. This strategy leads to partial differential equations, known as Black–Scholes equations, whose form depends on the particular financial instrument. In the simplest case of European put and call options, these partial differential equations can be transformed into the (backwards) heat equation, from which analytic pricing formulas are obtained [5]. For more exotic options, the partial differential equations do not have forms that are amenable to simplifying transformations, and the equations must be solved numerically using, for example, finite difference, finite element, or Monte Carlo methods [11–13].

An elegant alternative formulation of option pricing is based on path integrals, which were introduced into financial modeling by Dash [14–16] and Linetsky [17]. Path integrals were developed for quantum mechanics by Feynman [18,19], although Wiener [20] had earlier used an essentially equivalent development to study Brownian motion and the diffusion equation. The basic formulation of the path integral is based on the probability amplitude between given initial and final states expressed as the integral of complex exponentials of the *classical* action for all the paths between these states. The total amplitude is obtained from the usual rule for combining individual quantum mechanical amplitudes from each path.

* Corresponding author.
E-mail address: d.vvedensky@imperial.ac.uk (D.D. Vvedensky).

https://doi.org/10.1016/j.physa.2021.126231
0378-4371/

Path integrals have become a standard method in financial mathematics [16,21,22]. There are, broadly speaking, three types of solutions that can be obtained from a path integral formulation of an option or derivative. (i) Exact or closed-form solutions [23–26]. (ii) Perturbative [27] or approximate [28,29] solutions in particular parameter regimes. (iii) Numerical evaluation of the path integral based on Monte Carlo methods [26,30,31]. In this paper, we use the Markov chain Monte Carlo (MCMC) method, combined with the Metropolis–Hastings algorithm, to simulate the path integral for the BSM model of option pricing. After testing our methodology against standard closed-form solutions, we describe applications to non-Gaussian models. This approach can draw upon the extensive literature on applications of the MCMC method to the numerical evaluation of path integrals in both classical and quantum mechanical settings [32–35].

The outline of our paper is as follows. In Section 2 we derive the path integral for solving the Black–Scholes equation by constructing an analogy between the temporal evolution of option prices and that of quantum states. The numerical methods used to evaluate the Black–Scholes path integral are developed in Section 3, with particular attention paid to the autocorrelation of paths used for the Monte Carlo integration of the path integral, and the accuracy of our results with decreasing lattice spacing. Non-Gaussian models are investigated in Section 4, and a summary and prospects for future applications are provided in Section 5.

2. Theory

2.1. Financial options

A call option is a contract between two parties in which the buyer acquires the right, but not the obligation, to buy an underlying asset at a predetermined "strike price". In a European call option, for example, buyers may exercise their right only on the maturity date of the contract, which is agreed upon at signing. In other types of option, the right can be exercised at any time until the expiration of the contract.

The fluctuating price P of the underlying asset may change vastly between the signature date t_0 and the expiry date T of the contract. If, at the expiry date of a European call option, the price of the stock is greater than the strike price K, the value V of the option is given by $V(P(T), T) = P(T) - K$. Alternatively, if $P(T) < K$, the option is not exercised, in which case $V(P(T), T) = 0$. Thus, the payoff of the option at expiry is given by

$$V(P(T), T) = \max(P(T) - K, 0)\,. \tag{1}$$

The payoff of any path-independent option is expressed as

$$V(P(T), T) = F(P(T))\,, \tag{2}$$

where F is a specified function(al) of the final asset price. Beyond the break-even point ($P(T) = K$) of a European option, the holder can enjoy arbitrarily large profits. However, P may also decrease and, if the option holder declines to exercise their right, they will incur in the cost of having bought the option. The potential of unlimited profits countered by the pre-defined and bounded nature of the potential losses render financial options a particularly popular financial instrument for speculation and hedging.

The payoff of Asian options depends on the average price of the underlying asset over a certain period of time. These options are structured with a European expiration, meaning that they can be exercised only at expiration. However, their pay-off functional

$$F[P(t)] = \max(\exp(I(P)) - K, 0)\,, \tag{3}$$

is path dependent because the pay-off depends on the average of the asset price over the lifetime of the contract [17]:

$$I(P) = \frac{1}{T}\int_0^T P(t)\,dt\,. \tag{4}$$

2.2. Elements of stochastic differential equations

A Wiener process $W(t)$, or Brownian motion, over a time interval $t_0 \le t \le T$ is a stochastic process which is continuous in t and has independent Gaussian increments, i.e. $W(t'+\Delta t) - W(t')$ follows the normal distribution $N(0, \Delta t)$, with mean 0 and variance Δt and is independent of $W(t')$ for all times $t' \in [t_0, T]$. Given the basic process $W(t)$, we can define a Brownian motion $X(t)$ with drift $m(t)$ and variance $\sigma(t)$ as a Wiener process by the stochastic differential equation [5,36]

$$dX(t) = m(t)dt + \sigma(t)dW(t)\,. \tag{5}$$

We refer to a specific realization $\{X(t)\}$ of this process as a "path" in the state space of the process. The time is discretized by dividing the domain $[t_0, T]$ into N regular intervals $[t_k, t_k + \delta t]$, where $k = 0, 1, \ldots, N$, $\delta t = (T - t_0)/N$, with the identification $t_N = T$. Having discretized the time interval, we will use subscripts to indicate the time dependence on discrete variables, e.g. X_t instead of $X(t)$.

The conditional probability $p(X_T, T|X_0, t_0)$ for the process to be in the state X_T at time T given that it is in the state X_0 at time t_0 obeys the Chapman–Kolmogorov equation [37],

$$p\left(X_T, T\middle|X_0, t_0\right) = \prod_{i=1}^{N-1} \int_{\Omega(X_i)} dX_i \prod_{j=1}^{N} p\left(X_j, t_j\middle|X_{j-1}, t_{j-1}\right), \tag{6}$$

with $\Omega(X_i)$ being the domain of X_i. This equation recasts the problem of determining the full propagator $p(X_T, T|X_0, t_0)$ into that of calculating the short-time propagators, $p(X_{j+1}, t_{j+1}|X_j, t_j)$. The full propagator encodes all information about a stochastic process.

2.3. The Black–Scholes model and path integrals

In the celebrated Black–Scholes model, $\ln P$ is a Brownian process of the type in (5) [17]:

$$dP = mPdt + \sigma PdW\,. \tag{7}$$

The option premium V then follows the Black–Scholes equation

$$\frac{\partial V}{\partial t} + \frac{\sigma^2 P^2}{2}\frac{\partial^2 V}{\partial P^2} + rP\frac{\partial V}{\partial P} - rV_t = 0\,, \tag{8}$$

where r is the (fixed) growth rate of an instantaneously riskless portfolio.

The solution to a partial differential equation of the form of (8) is obtained from the Feynman–Kac formula [17],

$$V(P(t), t) = E(F(P)|P_0, t_0)\,, \tag{9}$$

where $E(F(P)|P_0, t_0)$ denotes the expectation value of the pay-off functional $F(P)$ given an initial state P_0 at time t_0. Like a standard expectation value, this can be obtained from the propagator $p(P_T, T|P_0, t_0)$ for the stochastic process of the underlying asset. We can therefore translate the problem of pricing options into that of constructing the short-time propagator $p(P_{j+1}, t_{j+1}|P_j, t_j)$ of the underlying stochastic process, which is then assembled into the full propagator to find the expectation value in (9).

We begin by introducing $X = \ln P$ into (8) to obtain

$$\frac{\partial V}{\partial t} = \hat{H}_{\rm BS} V\,, \tag{10}$$

where the Hamiltonian is [14,16,17,21,23]

$$\hat{H}_{\rm BS} = -\frac{1}{2}\sigma^2\frac{\partial^2}{\partial X^2} + \left(\frac{1}{2}\sigma^2 - r\right)\frac{\partial}{\partial X} + r\,. \tag{11}$$

This transformation reveals that the Black–Scholes equation can be mapped into the time-dependent Schrödinger equation for the temporal evolution of quantum states with a Wick rotation $t \to -it$.

We introduce the momentum operator $\hat{p}$ as the infinitesimal generator of translations $\delta X_j = X_{j+1} - X_j$, with the Hamiltonian (11) generating time translations $\delta t_j = t_{j+1} - t_j$ [38]. In this Wick-rotated framework, the unitary operator generating both translations is

$$\hat{U}(\delta t_j, \delta X_j) = \exp(i\hat{p}\delta X_j - \hat{H}_{\rm BS}\delta t_j) = \hat{\mathbb{1}} + i\hat{p}\delta X_j - \hat{H}_{\rm BS}\delta t_j + \mathcal{O}(\delta^2)\,, \tag{12}$$

where $\hat{\mathbb{1}}$ is the unit operator and we have used the fact that $\hat{U}$ is the Hilbert space representation of a Lie group of transformations, which therefore, can be expanded in a neighborhood of the identity [39]. In quantum mechanics, there would be a factor of $i/\hbar$ along with $\hat{H}_{\rm BS}$; however, the imaginary unit transforms into the real unit under Wick rotations, and we set $\hbar = 1$. In this formalism, the short time propagator can be written as

$$p(X_{j+1}, t_{j+1}|X_j, t_j) = \langle X_j|\hat{U}(\delta t_j, \delta X_j)|X_j\rangle = \int_{-\infty}^{\infty}\frac{dp}{2\pi}\langle X_j|\hat{U}(\delta t_j, \delta X_j)|p\rangle\langle p|X_j\rangle\,, \tag{13}$$

in which we have inserted a complete set of momentum states. By recasting the Hamiltonian (11) in terms of momentum operators,

$$\hat{H}_{\rm BS} = \frac{1}{2}\sigma^2\hat{p}^2 + \left(\frac{1}{2}\sigma^2 - r\right)i\hat{p} + r\hat{\mathbb{1}}\,, \tag{14}$$

substituting into the right-hand side of (13), using $\langle X_j|p\rangle\langle p|X_j\rangle = 1$, and performing the integration over the eigenvalues p of $\hat{p}$, we arrive at

$$p(X_{j+1}, t_{j+1}|X_j, t_j) = \exp\left\{-\left[\dot{X}_j - \left(\frac{\sigma^2}{2} - r\right)\right]^2 \frac{\delta t_j}{2\sigma^2}\right\} \frac{\exp(-r\delta t_j)}{\sqrt{2\pi\sigma^2\delta t_j}}, \tag{15}$$

where we have identified $\dot{X}_j = \delta X_j/\delta t_j$.

Equipped with the short-time propagator, we follow the prescription in (6) to construct the full propagator. Simultaneously, we take the continuum limit $N \to \infty$ (or, equivalently, $\delta t_j \to 0$) to obtain [14,16,17,21,23]

$$p\left(X_T, T\middle|X_0, t_0\right) = e^{-r\tau} \int_{X(t_0)=X_0}^{X(T)=X_T} \mathcal{D}[X(t)] \exp\left(-\int_{t_0}^{T} dt\ L\right), \tag{16}$$

where $\tau = T - t_0$ is the lifetime of the contract, and

$$L = \frac{\left[\dot{X}_t - \left(\frac{1}{2}\sigma^2 - r\right)\right]^2}{2\sigma^2} \tag{17}$$

is the Onsager–Machlup Lagrangian [40,41], and the integral over paths,

$$\int_{X(t_0)=X_0}^{X(T)=X_T} \mathcal{D}[X_t] \equiv \lim_{N\to\infty}\left[\prod_{k=1}^{N-1}\int_{-\infty}^{\infty} \frac{dX_k}{\sqrt{2\pi\sigma^2\delta t_k}}\right] \tag{18}$$

is the limit as the sequence of finite-dimensional multiple integrals becomes infinite.

Having obtained the full propagator, we are now able to evaluate the expectation value of any path-dependent function(al). In particular, by integrating over all possible final states, (9) becomes [14,16,17,21,23],

$$V(X_0, t_0) = e^{-r\tau} \int_{-\infty}^{\infty} dX_T \int_{X(t_0)=X_0}^{X(T)=X_T} \mathcal{D}[X(t)]F[X(t)]e^{-S_{\mathrm{BS}}}, \tag{19}$$

where

$$S_{\mathrm{BS}} = \int_{t_0}^{T} dt\ L \tag{20}$$

is the Black–Scholes action. Eq. (19) is our desired result: an expression for the price of an option given its defining pay-off function.

3. Numerical methods and results

By definition, the path integral is an infinite product of integrals. Equivalently, it can be seen as a single integral in infinite dimensions. Deterministic integration algorithms in d-dimensions have an error of order $\mathcal{O}(N^{-\alpha/d})$, where N is the number of samples and α is a constant that depends on the specific method ($\alpha = 2$ for trapezium method and $\alpha = 4$ for Simpson's method, for instance). However, the error in Monte Carlo methods scales as $N^{-1/2}$, independent of the dimensionality of the problem [42]. The path integral requires as many dimensions as possible to approximate the analytic result with increasing precision, making Monte Carlo methods the most suitable choice for a numerical evaluation of the Feynman–Kac formula.

The algorithm presented here is based on the Metropolis–Hasting (MH) scheme [43,44]. The MH method is a Markov chain Monte Carlo process [34,35] which generates samples (in our case option price paths) following a desired probability distribution. This method is ideally suited to the problem of option pricing due to the fact that, in (19), $F[X(t)]$ is a functional and $\exp\{-S_{\mathrm{BS}}[x(t')]\}$ can be interpreted as a probability density functional.

We first discretize the time between t_0 and T into a lattice of N points. We then generate a path – a specification of the asset price P at every lattice point. The path is created in a "hot state", that is, the price at every lattice site is chosen randomly, with the exception of the first point, which is set to the current asset price P_0. The Black–Scholes action is calculated for this initial path. A large number of sweeps is then performed across the lattice, large enough to minimize the inherent autocorrelations between sweeps (see below). In each sweep, N random perturbations are applied to a random selection of the N lattice points, meaning not all points are necessarily perturbed in every sweep. Clearly, the initial price P_0 is known, so the first lattice site is held fixed throughout. ' Each set of N perturbations gives rise to a proposed new path, which is approved or rejected according to the selection structure of the MH algorithm. The change in action δS_0 resulting from each perturbation is then evaluated. If $\delta S_0 \leq 0$, i.e. if the action of the new path is less than or equal to that of the previous path, the perturbation is always accepted, that is, the algorithm selects the proposed path as the trajectory from which to begin the following iteration. If instead $\delta S_0 > 0$, the new path is accepted with probability $\exp(-\delta S_0)$. For each accepted path, its payoff is evaluated and added to the running total. After the last sweep is performed, the running total is divided by the number of approved samples and multiplied by $\exp(-r\tau)$ to obtain the result of the path integral.

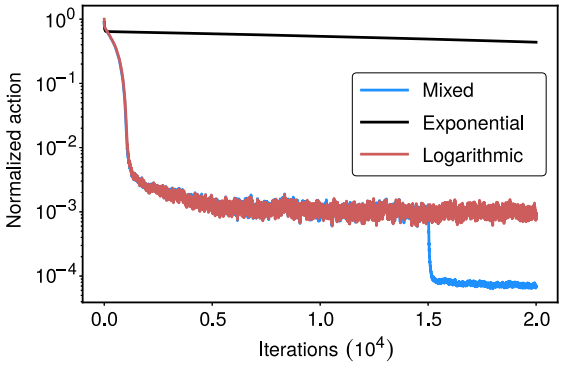

Fig. 1. Minimization of the (normalized) BS action for a hot state during the initialization of the simulation for different cooling schedules. The mixed schedule corresponds to a logarithmic cooling up to 15 000 iterations, when a more aggressive exponential cooling is turned on.

Following a large number of such sweeps, we are left with a path that is arbitrarily close to the classical path, the path minimizing the action. We are interested in the classical path, as well as nearby paths in action space, because these give the dominant contributions to the path integral in (19). For this reason, every Monte Carlo algorithm must account for "burn-in": the sample paths that are generated prior to convergence towards the classical path are generally not representative of the solution and, if used for evaluating the path integral, can distort the result. If the algorithm spawns a state far away from the classical path, many sweeps will be required to reach the region of action space in which paths have a large enough weight to provide a significant contribution to the integral. An example of this is the "hot state" initialization, in which the first path is just a random collection of points. While we could let the algorithm run and discard an appropriate number of initial samples (the ones before thermalization), this would be highly inefficient and would bias the result if the number of sweeps is not large enough. A better route would be to initialize the system in a "semi-cold state", i.e. a path similar to the classical path, if known, up to some random perturbations. For the BSM model, the classical path is just a straight line with slope μ, so is trivially achievable.

3.1. Simulated annealing

In the interest of possible future applications, we investigate the method of simulated annealing to optimally prepare the calculation. Simulated annealing [45] consists of a small modification to the Metropolis algorithm presented previously. In particular, we introduce a parameter T in the exponential factor:

$$\exp(-\delta S_0) \to \exp\left(-\frac{\delta S_0}{T}\right). \tag{21}$$

The choice of the notation T highlights the similar role that this parameter plays to the physical temperature of a system. At first, the temperature is set to $T = 1$, so that variations that increase the action have a reasonable probability of being accepted. The objective is for the chain to jump around, and avoid getting stuck in a local minimum of the action space. The parameter T is then lowered after each sweep using a "cooling scheme", which eventually forces the algorithm to almost only accept perturbations which lower the path's action until the trajectory becomes frozen around the classical path. After the minimization of the action is complete, the temperature is raised back to $T = 1$, and we begin evaluating the integral. The choice of the cooling scheme depends strongly on the problem at hand. To let the chain initially sample the function space of the paths extensively, we opt for a logarithmic decrease of N, which is followed by a more aggressive exponential decay. We find this combination to be very efficient, more than the soft or aggressive cooling schemes alone. In particular, while one might naively expect the exponential cooling to be faster, this scheme will actually freeze before reaching the classical path if the initial state is not close enough (Fig. 1).

Another device we implement to facilitate the convergence of the algorithm is an adaptive step size. If too many paths are approved, the size of the proposed perturbations is gradually increased, and vice versa, at least within a certain operative range. The algorithm tries to keep the average acceptance rate near 0.8, the conventional choice for such simulations [34,35].

3.2. Autocorrelation

Monte Carlo integration requires samples to be randomly chosen over the domain in order to evaluate an integral. Two factors can negatively affect the quality of the samples: the period of the random number generator and the correlation between the paths. To address the first issue, we employ a Mersenne Twister pseudorandom number generator [46]. The period of the Mersenne twister is a Mersenne prime number, which has the form $2p - 1$, where p is also a prime number.

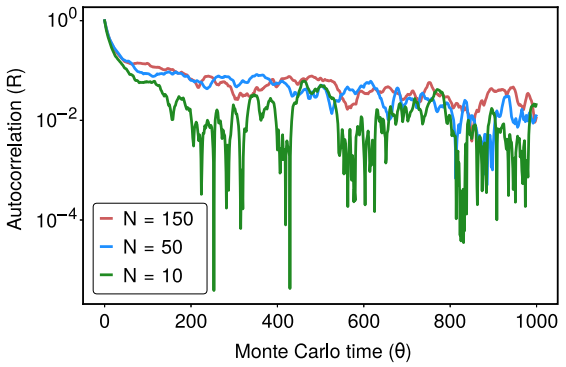

Fig. 2. The autocorrelation function as a function of Monte Carlo time (the number of sweeps) for paths generated for different lattice spacings with $P = £14$, $\tau = 6$ yrs, $r = 4.5$ yrs^{-1}, and $\sigma = 0.1$.

For the standard implementation of the Mersenne twister $p = 19937$. The second problem arises because the Metropolis algorithm generates a chain of paths in such a way that every path initiated relies on the previous one. Therefore, the paths are correlated, which creates a bias in the samples that needs to be accounted for. Note that autocorrelation is not dependent on the type of option; rather, it is an intrinsic property of the path generation scheme.

To mitigate the effects of autocorrelation, we first determine the Monte Carlo time θ (i.e. the number of sweeps) it takes for the chain to "forget" its initial state. We use the normalized autocorrelation function $R(\theta)$ to quantify these correlations:

$$R(\theta) = \frac{c(\theta)}{c(0)} \tag{22}$$

with

$$c(\theta) = \frac{1}{N-\theta}\sum_{n=1}^{N-\theta}(f_n - \langle f\rangle)(f_{n+\theta} - \langle f\rangle). \tag{23}$$

$R(\theta)$ takes values between 0 (corresponding to no correlation) and 1 (complete correlation). In this analysis, we choose the action of a path as the observable, since the paths are generated by minimizing S_{BS}.

Fig. 2 shows the calculated autocorrelation function for three lattices with the same lifetimes τ, but different spacing. In all three cases, there is an initial, rapid drop in R which then plateaus at $|R| < 10^{-1}$. Subsequently, we can consider the elements of the chain to be uncorrelated to the first chain. In general, we can approximate the autocorrelation function for small θ as a multi-modal exponential decay of the form

$$R(\theta) = a_1 e^{-\theta/\tau_1} + a_2 e^{-\theta/\tau_2} + \cdots, \tag{24}$$

which is usually truncated at the second term [35].

There are various definitions of the autocorrelation time τ_{corr}; for the sake of this discussion, we calculate τ_{corr} as the maximum of τ_1 and τ_2. We then estimate τ_{corr} by fitting a double exponential decay, as illustrated in Fig. 3(a). The analysis is quite sensitive to the set we choose for the fitting, but is useful for obtaining an order of magnitude estimate. As shown in Fig. 3(b), increasing the number of lattice sites extends the correlation time and therefore the computational burden, meaning sparser paths lose their correlation with the preceding paths more quickly. This explains the different decaying slopes in Fig. 2. In order to mitigate the effects of autocorrelation, we use only one of every τ_{corr} paths for the evaluation of the integral, discarding all the intermediate paths. In what follows, we take $\tau_{\mathrm{corr}} = 120$ yrs, keeping in mind that the typical size of our lattice is $N \approx 100$.

3.3. Accuracy and precision

The variables that most affect the precision and stability of the calculation are the number of lattice points and the number sweeps. Fig. 4 illustrates the performance of the method for a range of these variables when pricing an Asian option. The error bars in all our plots will show an uncertainty of σ on the data points. This is estimated by averaging over 5 statistically independent samples and extracting the standard deviation.

Fig. 4(a) clearly shows that the lattice spacing controls the accuracy of the method. Indeed, reducing δt, the temporal separation between neighboring points of a price path, has a crucial effect. A less coarse-grained lattice brings us closer to the continuum limit, increasing the number of nodes of the path and allowing us to sample a more complete path space. We have found that $\delta t \approx 10^{-3}$ yrs is more than adequate for satisfactory results, even if for European options this can

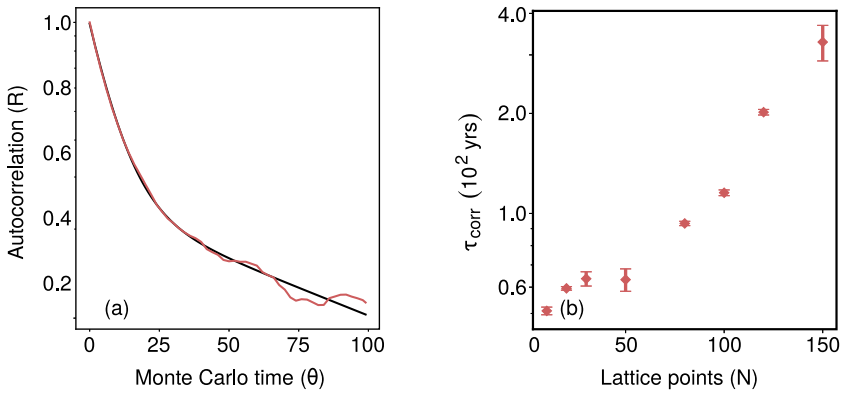

Fig. 3. (a) Autocorrelation function as a function of Monte Carlo time (the number of sweeps) for $N = 100$ lattice points with a fit to double exponential decay (24) with $\tau_1 = 8.6$ yrs and $\tau_2 = 115.4$ yrs, giving an estimate for $\tau_{corr} = 115.4$ yrs. (b) Autocorrelation times for different lattice spacings. Autocorrelation functions calculated with $P = £14$, $\tau = 6$ yrs, $r = 4.5$ yrs^{-1}, $\sigma = 0.1$. Error bars show uncertainty of σ on τ_{corr} obtained from the covariance matrix of the double exponential fit.

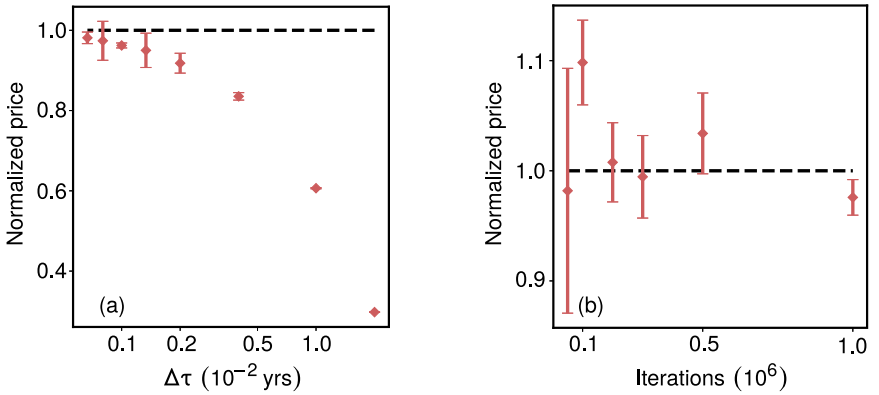

Fig. 4. Analysis of the precision and accuracy of Asian option pricing. (a) Increasing the number of lattice points improves accuracy, making the numerical result approach the theoretical prediction (indicated by the broken line), which corresponds to the continuum limit. (b) Increasing the number of sweeps enhances precision. The theoretical prediction is again indicated by the broken line.

be much longer. But δt is not the only key parameter. If the number of points is sufficiently large ($N \approx 120$) the method is accurate even for $\tau = 15$ yrs.

Fig. 4(b) shows how the number of iterations influences the precision of the method. In particular, to get an uncertainty as low as 5% we must perform at least 1 million sweeps (counting paths excluded to correct for autocorrelation). This can change significantly when δt is reduced. A more finely spaced lattice can accept only smaller perturbations, or the change in action of those variation would be too big and be rejected by the Metropolis algorithm. This implies that the chain moves slower and the computational burden is increased. In general, if $N = 120$, 2.5 million iterations are found to be more than adequate. These will be the parameters for all the calculations presented in the next sections, unless otherwise stated.

3.4. Performance in the BSM model

We first check that the paths generated with the Metropolis algorithm do indeed follow the required distribution. The easiest way to accomplish this is by examining the "kernel", i.e. the transition probability density between the initial and final state. Effectively, we replace the payoff in (19) with unity, and evaluate the path integral. We can then check this with the analytical result for this Gaussian path integral. In particular, the BSM model requires the logarithm of final prices to be distributed normally around the value that the asset would achieve at τ with constant drift [17]:

$$V_\mu(X_0, t_0) = \frac{1}{\sqrt{2\pi\sigma^2\tau}} \exp\left[-\frac{(X_T - X_0 - \mu\tau)^2}{2\sigma^2\tau}\right], \tag{25}$$

where, from (11), $\mu = r - \frac{1}{2}\sigma^2$. Fig. 5 shows that the distribution of final prices agrees with the analytic solution. Note that, for European options, only the kernel is important, since the distribution of final points determines the derivative.

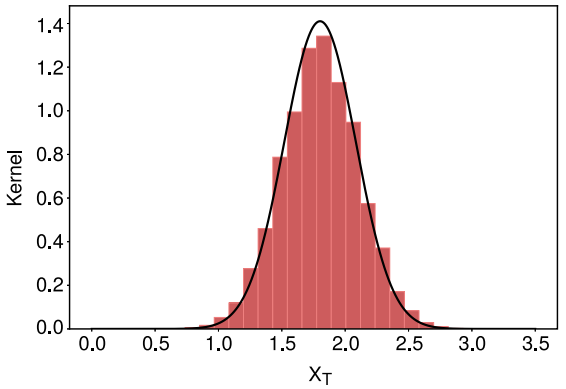

Fig. 5. Kernel for paths generated by the Metropolis algorithm with $P = £6$, $\sigma = 0.4$, $r = 0.1$ and $\tau = 0.5$ yrs for $N = 30$ final points, shown as a histogram, compared with the analytic result (25), shown as a solid curve.

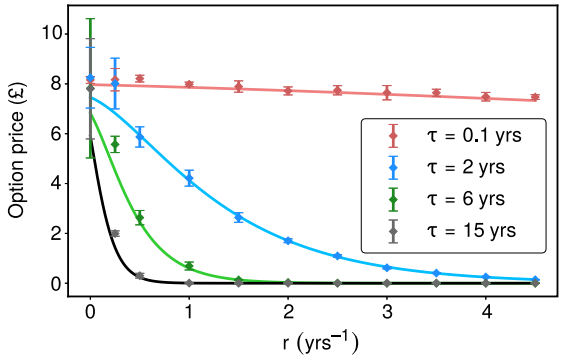

Fig. 6. Price as a function of interest rate for an Asian option with strike price $K = £6$ for different times to expiry. As $r \to \infty$ the value of the option tends to zero. The parameters of the simulation are $P = £14$ and $\sigma = 0.1$. Solid lines are the theoretical results.

Hence, we can be confident that the calculations for vanilla options will be correct. This is not sufficient for the more interesting path-dependent Asian options, but the fact that the final prices are correctly distributed strongly suggests that the paths are generated correctly.

Before moving on to non-Gaussian models, we can use options in the BSM model as proof of principle of our method, exploiting known analytical solutions. As a simple example, we consider the behavior of Asian option prices as a function of the interest rate r for different expiry times τ (Fig. 6). The results produced by our algorithm are in agreement with the available analytical results, with almost all data points lying well within one standard deviation of the analytical curves. All curves eventually vanish. This is likely due to the fact that the pay-off is related to the mean of the asset price, rather than the asset price itself; this grows at a slower rate – modulo stochastic fluctuations – leading to the dominance of the discounting factor $\exp(-r\tau)$ at large r. We see more extreme behavior for larger values of τ. This is explained by the differences in the asset price due to a larger or smaller r accumulating in time and thus becoming more noticeable at larger τ.

4. Non-Gaussian dynamics

Due to the Gaussian nature of its increments, the geometric Brownian motion that underlies the Black–Scholes model results in a normal distribution of the price paths. Despite its simplicity, this model captures several features observed in real markets. It fails, however, to properly take into account or predict features such as the observed leptokurtosis of real stock price curves, market crashes, and large price fluctuations [47,48].

Both leptokurtic and Gaussian distributions belong to the family of Lévy α-stable distributions. This category includes all functions that can be written as

$$f(x) = \frac{1}{2\pi} \int dt \, \exp\left(it\mu - |ct|^{\alpha}(1 - i\beta \, \mathrm{sgn}(t) \, \Phi)\right) , \tag{26}$$

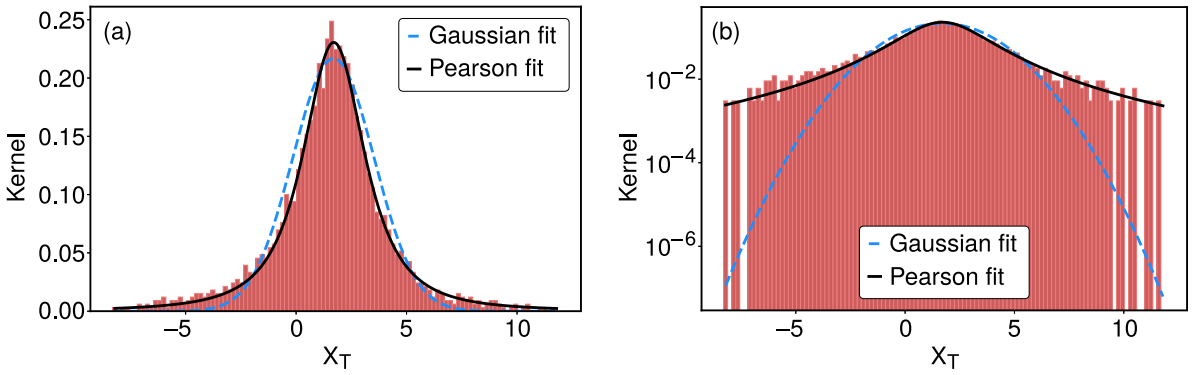

Fig. 7. (a) Kernel for paths generated with the modified Lagrangian for $p = 1.3$, $\gamma = 0.3$, $P = £14$, $\sigma = 0.1$, $r = 0.1$ and $\tau = 0.5$ yrs. The distribution of final points deviates significantly from the best Gaussian fit (black line), showing a narrower peak and fatter tails as required by a Lévy distribution. Note that X_T can be negative, since it is the logarithm of the price and not the price itself. (b) The same data, now shown against a semilogarithmic scale, highlights the deviation between the Gaussian fit and the actual distribution of final points.

where

$$\Phi = \begin{cases} \tan\left(\frac{\pi\alpha}{2}\right) & \text{if } \alpha \neq 1; \\ -\frac{2}{\pi}\log|t| & \text{if } \alpha = 1. \end{cases} \tag{27}$$

The parameters μ and c control the shift and scale of the distribution, respectively, while β and α define the shape.

For $\alpha = 2$ a Gaussian profile with variance $2c^2$ and mean μ is recovered (β has no effect in this case); any other value results in a leptokurtic distribution. Such distributions have sharper peaks and fatter, Paretian tails, thus allowing for a wider variety of stock price behaviors. For instance, the larger non-zero probability at the distribution's tails can be used to reproduce market crashes and surges. However, the fact that exact analytical definitions that do not involve Fourier integrals are in general not available for such stable distributions makes their use in stochastic models somewhat problematic.

Nevertheless, Paolinelli and Arioli [26] suggested that a simple redefinition of the Black–Scholes action S_{BS} in (20) results in a quasi-Lévy distribution of paths. Accordingly, we define the generalized action

$$S_L = \frac{1}{2\sigma^p}\left\{\int_{t_0}^{T} dt\left[\dot{X}_k - \left(\frac{1}{2}\sigma^2 - r\right)\right]^p\right\}^\gamma, \tag{28}$$

where $p \in [1, 2]$ and $\gamma \in [0, 1]$ are parameters of the distribution, so that (19) becomes,

$$V(X_0, t_0) = e^{-r\tau}\int_{-\infty}^{\infty} dX_T \int_{X(t_0)=X_0}^{X(T)=X_T} \mathcal{D}[X(t)]\, F[X(t)]\, e^{-S_L}\,. \tag{29}$$

This action recovers the Black–Scholes model for $p = 2$ and $\gamma = 1$. This simple redefinition of the action is fully compatible with our earlier theoretical considerations and can be easily implemented with the algorithm described above.

The distribution of final points for $p = 1.3$ and $\gamma = 0.3$ is shown in Fig. 7. The best Gaussian fit clearly does not follow the envelope of the histogram. However, the Pearson fit, which is designed for non-normal functions, perfectly matches the leptokurtic behavior of the data. As p and γ get closer to 2 and 1, respectively, the distribution more closely follows a Gaussian function. Conversely, very small values of p and γ produce distributions of final points that are too wide to be sampled in reasonable times for the available computational resources, due to the fat tails. Figs. 8 and 9 show a more quantitative study of how option pricing changes in this model with respect to the Black–Scholes case. The parameters for these simulations where chosen according to the investigation of [26], which is based on real market data. The results were obtained using lattices of 120 points and performing 3 million sweeps. We expect the precision to improve with greater numbers of iterations and lattice points, as evidenced in Fig. 4 and Section 3.2. As noted earlier, the large error bars are a result of the distributions being very wide and thus difficult to sample properly.

We notice that distributions with narrower tails correspond to more precise results. Furthermore, for both sets of European options, we notice a slight but systematic offset of the data points towards higher prices compared to the Black–Scholes results. This means that, while no data point by itself allows us to confidently reject the null hypothesis (that the leptokurtic data follows the Black–Scholes model), the data set as a whole does provide important evidence that a deviation occurs. As a quantitative approximation, we can assume that each result is independent of the others and follows a Gaussian distribution. Under these assumptions, we obtain a likelihood on the order of 10^{-7} that a data set randomly sampled from the Black–Scholes model would show the systematic offset seen in Fig. 8. While this is not a rigorous

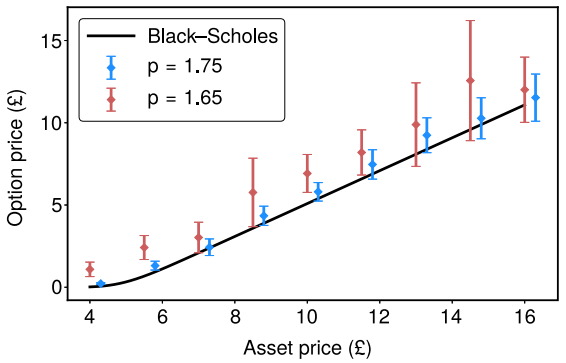

Fig. 8. Price as a function of initial asset price for a European option with strike price $K = £6$ in an approximated Lévy model. Blue data points are displaced on x-axis by 0.2 for reasons of clarity. The parameters of the simulations are $P = £14$, $\sigma = 0.1$, $r = 0.1$, $\tau = 2$ yrs.

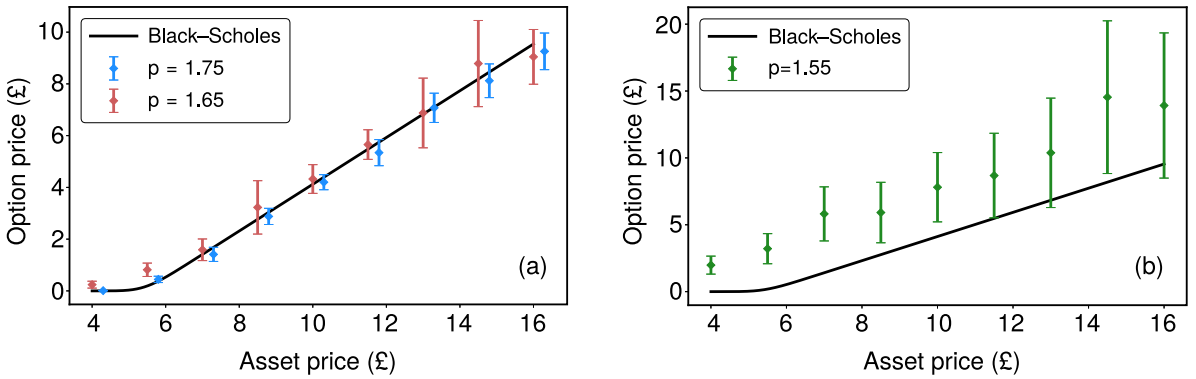

Fig. 9. Price as a function of initial asset price for an Asian option with strike price $K = £6$ in an approximated Lévy model. Blue data points are displaced on x-axis by 0.2 for reasons of clarity. The parameters of the simulations are $P = £14$, $\sigma = 0.1$, $r = 0.1$, $\tau = 2$ yrs.

estimate, it does suggest that the leptokurtic distribution results in larger option prices compared to the Black–Scholes model. This feature is less evident in the case of Asian options, which seem compatible with Black–Scholes dynamics for both $p = 1.65$ and $p = 1.75$, as shown in Fig. 9(a). It is only for $p = 1.55$ (Fig. 9(b)) that the data hint at a possible deviation; again, the option prices within the non-Gaussian model seem to be higher. This is expected: larger jumps in the paths render a very high pay-off at expiry more likely, which must be reflected in a larger value of the financial derivative. The behavior also agrees with the knowledge that the BSM model underprices options "deep in the money", i.e. with $S \gg K$ [49].

5. Conclusions

We have described a computational method that uses the path integral machinery developed in quantum mechanics to the pricing of options. Our approach is to generate a chain of paths with the Metropolis–Hastings algorithm (simulating the Brownian motion of the price of an asset in time) to statistically evaluate path integrals. After having outlined the theoretical fundamentals of the BSM model and of the path integral, we have analyzed the performance and limitations of the numerical method.

We first introduced the issue of autocorrelation and presented the steps taken to reduce its effects on the computational results. These consist of the estimation of the Monte Carlo autocorrelation time τ_{corr} by a double exponential fit and the selection of only one every τ_{corr} paths generated for the evaluation of the integral. We found that a denser lattice of the simulation implies a larger τ_{corr}, and suggested choosing $\tau_{\text{corr}} \approx 120$ for a lattice of size $N \approx 100$. We then analyzed the performance of the method, sampling the parameter space of the simulation. We observed that the lattice spacing Δt affects mainly the accuracy of the result, while its precision is affected mainly by the number of iterations performed.

Secondly, we evaluated option prices within the BSM model as a proof of principle of our approach. By looking at the distribution of final prices for the paths generated, we demonstrated that the computational method we adopt does indeed produce a log-normal distribution, which is the typical feature of the BSM framework. We then evaluated Asian (path dependent) option prices for a set of different parameters, showing that our results agree with the analytical predictions.

Having demonstrated the validity of our approach, we adapted the method to price options outside the BSM model. In particular, by suitably changing the action in the path integral formulation of the problem, we managed to reproduce a final distribution of prices which is log-leptokurtic instead of log-normal. These non-Gaussian models are known for modeling market movements more accurately, but have the draw-back of not having closed-form solutions. The possibility of extending the method to such frameworks shows its generality and flexibility. We then evaluated Asian and European option prices in these non-Gaussian regimes, finding that the more leptokurtic the final distribution of prices, the more our computational results differ from the BSM predictions. Nonetheless, as the tails in the kernel become fatter, our computational method converges more slowly, causing a large uncertainty in any calculation that is done in a reasonably short amount of time.

Further work should be done to reduce the autocorrelation time of the Markov chain, since it can have a very large effect on computational efficiency. In this direction, possible solutions to test could be over-relaxation [35] and generative adversarial networks [50]. If we do manage to improve the efficiency of the method, the model could be viable even for very log-leptokurtic models. It would also allow us to extend our method on stochastic volatility models. This would require a modification in order to generate a second collection of paths with the Metropolis algorithm, representing the Brownian motion of the volatility of the option in time. However, this would also heavily increase the computational burden, making the optimization of the method our first priority.

CRediT authorship contribution statement

Pietro Capuozzo: Methodology, Formal analysis, Software, Validation, Writing - original draft. **Emanuele Panella:** Methodology, Formal analysis, Software, Validation, Writing - original draft. **Tancredi Schettini Gherardini:** Methodology, Formal analysis, Software, Validation, Writing - original draft. **Dimitri D. Vvedensky:** Conceptualization, Methodology, Supervision, Writing - reviewing and editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] Bank for International Settlements, Statistical release: OTC derivatives statistics at end-June 2020, Available at https://www.bis.org/publ/otc_hy2011.htm.
[2] F. Black, M. Scholes, The pricing of options and corporate liabilities, J. Polit. Econ. 81 (1973) 637–654, http://dx.doi.org/10.1086/260062.
[3] R. Merton, Theory of rational option pricing, Bell J. Econ. 4 (1973) 141–183, http://dx.doi.org/10.2307/3003143.
[4] R.N. Mantegna, H.E. Stanley, An Introduction to Econophysics: Correlations and Complexity in Finance, Cambridge University Press, Cambridge, 2000.
[5] J.C. Hull, Options, Futures, and Other Derivatives, nineth ed., Pearson, Boston, 2015.
[6] S. Heston, A closed-form solution for options with stochastic volatility with applications to bond and currency options, Rev. Financ. Stud. 6 (1993) 327–343, http://dx.doi.org/10.1093/rfs/6.2.327.
[7] W. Schoutens, Exotic options under Lévy models: An overview, J. Comput. Appl. Math. 189 (2006) 526–538, http://dx.doi.org/10.1016/j.cam.2005.10.004.
[8] J.P.A. Devreese, D. Lemmens, J. Tempere, Path integral approach to Asian options in the black–scholes model, Physica A 389 (2010) 780–788, http://dx.doi.org/10.1016/j.physa.2009.10.020.
[9] L. Chan, S.-P. Zhu, An explicit analytic formula for pricing barrier options with regime switching, Math. Financ. Econ. 9 (2015) 29–37, http://dx.doi.org/10.1007/s11579-014-0119-z.
[10] J.A.G. Cervera, Solution of the black–scholes equation using artificial neural networks, J. Phys.: Conf. Ser. 1221 (2019) 012044, http://dx.doi.org/10.1088/1742-6596/1221/1/012044.
[11] P. Boyle, Options: A Monte Carlo approach, J. Financ. Econ. 4 (1977) 323–338, http://dx.doi.org/10.1016/0304-405X(77)90005-8.
[12] Y. Achdou, O. Pironneau, Computational Methods for Option Pricing, SIAM, Philadelphia, 2005.
[13] M. Gilli, D. Maringer, E. Schumann, Numerical Methods and Optimization in Finance, Elsevier, Amsterdam, 2011.
[14] J. Dash, Path Integrals and Options, Part I, CNRS Preprint CPT-88/PE.2206, 1988.
[15] J. Dash, Path Integrals and Options, Part II, CNRS Preprint CPT-89/PE.2333, 1989.
[16] J.W. Dash, Quantitative Finance and Risk Management: A Physicist's Approach, World Scientific, Singapore, 2004.
[17] V. Linetsky, The path integral approach to financial modeling and options pricing, Comput. Econ. 11 (1998) 129–163, http://dx.doi.org/10.1023/A:1008658226761.
[18] R.P. Feynman, Space-time approach to non-relativistic quantum mechanics, Rev. Modern Phys. 20 (1948) 367–387, http://dx.doi.org/10.1103/RevModPhys.20.367.
[19] R.P. Feynman, A.R. Hibbs, Quantum Mechanics and Path Integrals, McGraw–Hill, New York, 1965.
[20] N. Wiener, The average value of a functional, Proc. Lond. Math. Soc. 22 (1921) 454–467, http://dx.doi.org/10.1112/plms/s2-22.1.454.
[21] B.E. Baaquie, Quantum Finance: Path Integrals and Hamiltonians for Options and Interest Rates, Cambridge University Press, Cambridge, 2004.
[22] H. Kleinert, Path Integrals in Quantum Mechanics, Statistics, Polymer Physics, and Financial Markets, World Scientific, Singapore, 2004.
[23] B.E. Baaquie, A path integral approach to option pricing with stochastic volatility: Some exact results, J. Phys. I (France) 7 (1997) 1733–1753, http://dx.doi.org/10.1051/jp1:1997167.
[24] H. Kleinert, Option pricing for non-Gaussian price fluctuations, Physica A 338 (2004) 151–159, http://dx.doi.org/10.1016/j.physa.2004.02.037.
[25] D. Lemmens, M. Wouters, J. Tempere, Path integral approach to closed-form option pricing formulas with applications to stochastic volatility and interest rate models, Phys. Rev. E 78 (2008) 016101, http://dx.doi.org/10.1103/PhysRevE.78.016101.

[26] G. Paolinelli, G. Arioli, A model for stocks dynamics based on a non-Gaussian path integral, Physica A 517 (2019) 499–514, http://dx.doi.org/10.1016/j.physa.2018.11.044.
[27] Z. Kakushadze, Path integral and asset pricing, Quan. Financ. 15 (2015) 1759–1771, http://dx.doi.org/10.1080/14697688.2015.1052092.
[28] A. Matacz, Path-dependent option pricing: the path integral partial averaging method, J. Comput. Financ. 6 (2002) 79–103, doi: https://10.21314/JCF.2002.108.
[29] A. Issaka, I. SenGupta, Feynman path integrals and asymptotic expansions for transition probability densities of some Lévy driven financial markets, J. Appl. Math. Comput. 54 (2017) 159–182, http://dx.doi.org/10.1007/s12190-016-1002-2.
[30] M. Rosa-Clot, S. Taddei, A path integral approach to derivative pricing II: Numerical methods, Int. J. Theor. Appl. Financ. 5 (2002) 123–146, http://dx.doi.org/10.1142/S0219024902001377.
[31] G. Montagna, O. Nicrosini, N. Moreni, A path integral way to option pricing, Physica A 310 (2002) 450–466, http://dx.doi.org/10.1016/S0378-4371(02)00796-3.
[32] M. Creutz, B. Freedman, A statistical approach to quantum mechanics, Ann. Physics 132 (1981) 427–462, http://dx.doi.org/10.1016/0003-4916(81)90074-9.
[33] M. Creutz, L. Jacobs, C. Rebbi, Monte Carlo Computations in lattice gauge theories, Phys. Rep. 95 (1983) 201–282, http://dx.doi.org/10.1016/0370-1573(83)90016-9.
[34] C. Morningstar, The Monte Carlo method in quantum field theory, 2007, https://arxiv.org/abs/hep-lat/0702020.
[35] M.J.E. Westbroek, P.R. King, D.D. Vvedensky, S. Dürr, User's guide to Monte Carlo methods for evaluating path integrals, Amer. J. Phys. 86 (2018) 293–304, http://dx.doi.org/10.1119/1.5024926.
[36] M. Baxter, A. Rennie, Financial Calculus: An Introduction to Derivative Pricing, Cambridge University Press, Cambridge, 1996.
[37] N.G. van Kampen, Stochastic Processes in Physics and Chemistry, Elsevier, Amsterdam, 2004.
[38] J.J. Sakurai, Modern Quantum Mechanics, Cambridge University Press, Cambridge, 2010.
[39] S. Weinberg, The Quantum Theory of Fields, Vol. 1, Cambridge University Press, Cambridge, 2005.
[40] L. Onsager, S. Machlup, Fluctuations and irreversible processes, Phys. Rev. 91 (1953) 1505–1512, http://dx.doi.org/10.1103/PhysRev.91.1505.
[41] W. Horsthemke, A. Bach, Onsager–machlup function for one dimensional nonlinear diffusion processes, Z. Phys. B 22 (1975) 189–192, http://dx.doi.org/10.1007/BF01322364.
[42] W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.T. Flannery, Numerical Recipes in C++ The Art of Scientific Computing, Cambridge University Press, Cambridge, 1992.
[43] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (1953) 1087–1092, http://dx.doi.org/10.1063/1.1699114.
[44] W.A. Hastings, Monte Carlo Sampling methods using Markov chains and their applications, Biometrica 57 (1970) 97–190, http://dx.doi.org/10.1093/biomet/57.1.97.
[45] S. Kirkpatrick, C.D. Gelatt Jr., M.P. Vecchi, Optimization by simulated annealing, Science 220 (1983) 671–680, http://dx.doi.org/10.1126/science.220.4598.671.
[46] M. Matsumoto, T. Nishimura, Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator, ACM Trans. Model. Comput. Simul. 8 (1998) 3–30, http://dx.doi.org/10.1145/272991.272995.
[47] P. Bak, M. Paczuski, M. Shubik, Price variations in a stock market with many agents, Physica A 246 (1997) 430–453, http://dx.doi.org/10.1016/S0378-4371(97)00401-9.
[48] A.A. Farhadi, D.D. Vvedensky, Risk, randomness, crashes and quants, Contemp. Phys. 44 (2003) 237–257, http://dx.doi.org/10.1080/0010751031000077396.
[49] S. Hodges, Options: Recent Advances in Theory and Practice, Vol. 2, Manchester University Press, Manchester, 1992.
[50] J.M. Pawlowski, J.M. Urban, Reducing autocorrelation times in lattice simulations with generative adversarial networks, Mach. Learn.: Sci. Technol. 1 (2020) 045011, http://dx.doi.org/10.1088/2632-2153/abae73.