

A variational quantum algorithm for the Feynman-Kac formula

Hedayat Alghassi¹, Amol Deshmukh¹, Noelle Ibrahim¹, Nicolas Robles¹, Stefan Woerner², and Christa Zoufal^{2,3}

¹IBM Quantum, Yorktown Heights, NY, US

²IBM Quantum, IBM Research Europe – Zurich, Switzerland

³Institute for Theoretical Physics, ETH Zurich, Switzerland

We propose an algorithm based on variational quantum imaginary time evolution for solving the Feynman-Kac partial differential equation resulting from a multidimensional system of stochastic differential equations. We utilize the correspondence between the Feynman-Kac partial differential equation (PDE) and the Wick-rotated Schrödinger equation for this purpose. The results for a $(2 + 1)$ dimensional Feynman-Kac system obtained through the variational quantum algorithm are then compared against classical ODE solvers and Monte Carlo simulation. We see a remarkable agreement between the classical methods and the quantum variational method for an illustrative example on six and eight qubits. In the non-trivial case of PDEs which are preserving probability distributions – rather than preserving the ℓ_2 -norm – we introduce a proxy norm which is efficient in keeping the solution approximately normalized throughout the evolution. The algorithmic complexity and costs associated to this methodology, in particular for the extraction of properties of the solution, are investigated. Future research topics in the areas of quantitative finance and other types of PDEs are also discussed.

1 Introduction

Differential equations govern the evolution of complex systems arising in diverse fields such as physical and natural sciences as well as in economics and finance. The Feynman-Kac formula connects stochastic differential equations (SDEs), which describe the evolution of random variables, to parabolic (second-order, linear) partial differential equations (PDEs) [1, 2]. This kind of equation is similar and can formally be identified to the Schrödinger equation [2, §1]. This remarkable connection provides a methodology for linking the heat equation to Brownian motion and yields a technique for solving PDEs by Monte Carlo simulations. Conversely, many stochastic processes are amenable to be studied using deterministic methods. A perennial application of these notions is the Black-Scholes equation

Hedayat Alghassi: hedayat.alghassi@ibm.com

Amol Deshmukh: amol.deshmukh@ibm.com

Noelle Ibrahim: noel.ibrahim@ibm.com

Nicolas Robles: nicolas.robles@ibm.com

Stefan Woerner: wor@zurich.ibm.com

Christa Zoufal: ouf@zurich.ibm.com

and geometric Brownian motion which enables pricing of financial derivatives [3]. This connection also plays a crucial role in the rigorous and constructive formulation of quantum field theories, which are essential for the description of nature at the most fundamental level [4, 5]. On a more practical level, this formula is routinely utilized for the numerical evaluation of the quantum properties of atomic and subatomic systems in quantum chemistry as well as in atomic and particle physics [6, 7, 8].

Recent research on the suitability of quantum computers for solving differential equations has grown substantially due to the promise of efficiency and speed-up brought by solving these problems quantum mechanically. In [9, §I], the link between the Feynman-Kac formula and stochastic processes is pointed out but not further developed. Similarly, on [10, §II], a remark about a ‘deep’ connection between the heat equation and stochastic processes is made. In [11, §2.1], Feynman-Kac formula is invoked to transform the price of a European call option from a conditional expectation value to the solution of the Wick-rotated Schrödinger equation through a particular Hamiltonian. Furthermore, in [12], the connection between trinomial trees for random walks, SDEs, and the Fokker-Planck equation, is established but the full picture of generalizing to the Feynman-Kac formula is not developed. In this paper, we will explore the advantages of entirely *transitioning* from an SDE to a PDE by not relying on a random-walk tree or other kinds of discrete probabilistic structures.

An alternative approach to estimate properties of stochastic processes is by leveraging Quantum Amplitude Estimation (QAE) [13] and its variants [14, 15, 16], which can achieve a quadratic speed-up over classical Monte Carlo simulation. In contrast to classical Monte Carlo simulation, all possible paths of a (discretized) stochastic process are modelled in a single quantum state in superposition. In other words, a set of qubits is prepared such that measuring their state would result in sampling from the distribution of the stochastic process of interest. Although preparing such states is in principle always possible for reasonable stochastic processes [17], efficient realization of this method demands a careful analysis [18, 19, 20, 21] and may not always result in a practical quantum advantage. Once such a state is prepared, QAE can be used to evaluate different (path-dependent) objective functions. In [22], a similar connection between SDEs and PDEs is developed focused on Monte Carlo methods rather than variational methods associated to PDEs.

In most of the existing literature on option pricing for equities using quantum computers, e.g. [18, 23], however, an SDE is tacitly solved, namely the so-called Geometric Brownian motion $dX_t = \mu X_t dt + \sigma X_t dW_t$, where μ and σ are constants known as drift and volatility respectively, and W_t is the Brownian motion. Once this SDE is solved and shown to be $X_t = X_0 e^{(\mu - \sigma^2/2)t + \sigma(W_t - W_0)}$ with $X_0 > 0$, which produces a log-normal distribution, then the pricing of a particular security begins by applying QAE. However, this SDE is one of the simplest and one of the few that can be solved analytically. For the most part SDEs rarely admit analytical solutions [24, 25] and are usually dealt with numerically with large amounts of classical resources. Here we bypass this problem by not solving the SDE and instead simulating an associated – and deterministic – PDE for the calculations of the conditional expectations of the stochastic process X_t .

We now propose to synthesize these ideas into a unified framework and bring economy of thought by describing how the heat, Schrödinger, Black-Scholes, Hamilton-Jacobi, Fokker-Planck equations can be thought of as different manifestations of the Feynman-Kac formulation. Once this framework is established, we then employ recent techniques from variational quantum time evolution [26, 27, 28] to introduce an algorithm for solving expectations of SDEs on quantum computers by *first* linking them to PDEs and *then* by establishing a differential operator, an infinitesimal generator, that can be evolved on a

quantum computer.

The paper is structured as follows. In Section 2, we introduce the Feynman-Kac formula. The algorithm along with its complexity and advantages is developed in Section 3 and numerically illustrated in Section 4. Future exploratory items are brought forward in Appendix H.

2 The Feynman-Kac formula

In this section we first introduce the definition and existence of Brownian motion and immediately use its properties to derive a partial differential equation whose solution will be a conditional expectation of the Brownian motion. We formalize these results in one and several dimensions. A table of formal identifications of the Feynman-Kac formula and other types of second order PDEs is also provided. Moreover, we establish the discretization process that will be needed in Section 4 for our quantum and classical algorithms with periodic boundary conditions.

2.1 Brownian motion and differential equations

We start by reminding the reader about Brownian motion and link it to the heat equation. We shall proceed axiomatically in order to provide all the tools needed throughout the paper. Suppose Ω is a sample space, $\omega \in \Omega$ is an event of this space and $\mathcal{N}(\mu, \sigma)$ is the normal distribution with mean μ and variance σ^2 , respectively.

Definition 2.1. Brownian motion $W_t = W(t)$ with $t \geq 0$ is a continuous stochastic process that satisfies the following characteristics [29, §5.3]:

- 1) $W(0) = 0$ almost surely;
- 2) $W(t)$ has independent increments: $W(t+u) - W(t)$ is independent of $\sigma(W(s) : s \leq t)$ for $u \geq 0$, where σ denotes the sigma algebra generated by $W(s)$;
- 3) $W(t)$ has Gaussian increments: $W(t+u) - W(t)$ is normally distributed with mean 0 and variance u , i.e. $W(t+u) - W(t) \sim \mathcal{N}(0, u)$;
- 4) $W(t)$ has continuous paths: $W(t)$ is a continuous function of t , i.e. $t \rightarrow W(t, \omega)$ is continuous in t for all $\omega \in \Omega$.

The probability measure under which W_t is a Brownian motion is denoted by \mathbb{P} and the filtered probability space will be denoted by $(\Omega, \mathcal{F}, \mathbb{P})$. The existence of Brownian motion – often overlooked in the literature of quantum computing, e.g. [12, §II.A], [30, §II.A] or [23, §II] – was established by Wiener by proving that indeed the first three items were compatible with the continuity of the paths of W_t .

Theorem 2.1 (Wiener). Brownian motion exists.

In Figure 1 we display some possible paths of Brownian motion and their distribution. A key result, which is best motivated informally, is due to Itô and it states the following [31, §3.1] and [32, Appendix B], see also [33, 29, 34].

Theorem 2.2 (Itô's lemma). For infinitesimal Brownian motion increments, one has that

$$dW_t^2 = dt. \quad (1)$$

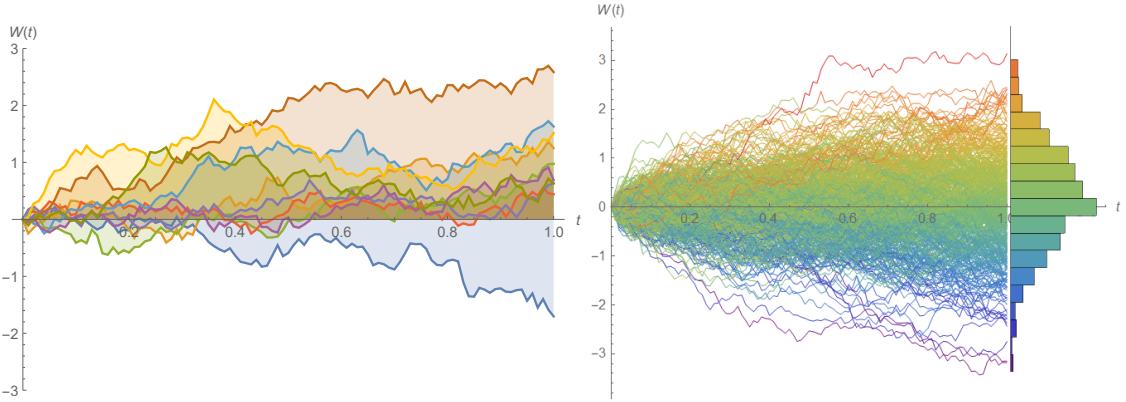


Figure 1: Left: 10 realizations of Brownian motion. Right: 500 realizations of Brownian motion and their resulting distribution.

This result states that the variance of the Brownian increment is proportional to the time interval of the increment. In other words, Brownian motion accumulates quadratic variation at rate one per unit time.

Furthermore, Brownian motion has some unusual properties. For instance while it is continuous everywhere it is (with probability one) nowhere differentiable and it scales like a fractal. Moreover, Brownian motion will eventually attain any and every value no matter how large or negative and once Brownian motion attains a value, it immediately attains it again infinitely often, and then after that again from time to time in the future, see [33, §3.1] for other items in the ‘bestiary’ of Brownian motion. Globally, the law of large numbers states that $W_t/t \rightarrow 0$ almost surely as $t \rightarrow \infty$, [5, Proposition 2.8].

Next, we move on to the backward heat equation [32, §3.1.9] as an illustrative example. Suppose that $(W_t)_{t \geq 0}$ is a Brownian motion defined on the filtered probability space $(\Omega, \mathcal{F}, \mathbb{P})$, and consider the simplest type of SDE

$$dX_t = dW_t. \quad (2)$$

Let $\psi : \mathbb{R} \rightarrow \mathbb{R}$ be a Borel-measurable function such that

$$\int_{-\infty}^{\infty} e^{-ax^2} |\psi(x)| dx < \infty \quad (3)$$

for some $a > 0$. Set the conditional expectation

$$u(x, t) = \mathbb{E}[\psi(X_T) | X_t = x], \quad \forall (x, t) \in \mathbb{R} \times [0, T]. \quad (4)$$

From the properties of Brownian motion we can write u as

$$u(x, t) = \frac{1}{\sqrt{2\pi(T-t)}} \int_{-\infty}^{\infty} \psi(y) \exp\left(-\frac{(y-x)^2}{2(T-t)}\right) dy \quad (5)$$

for every $(x, y, t) \in \mathbb{R}^2 \times [0, T]$. Taking the partial derivative with respect to t and the second order partial derivative with respect to x , one can easily verify that u satisfies the PDE

$$-\frac{\partial u}{\partial t} = \frac{1}{2} \frac{\partial^2 u}{\partial x^2}, \quad \forall (x, t) \in \mathbb{R} \times [0, T] \quad (6)$$

with the terminal condition $u(x, T) = \psi(x)$ for $x \in \mathbb{R}$. This result contains the essence of Feynman-Kac formulation as it expresses the solution to the parabolic PDE (6) as

the expected value of a functional of Brownian motion (4) coming from an SDE (2) with a terminal condition that is subjected to (3). While (6) is indeed the heat equation, some guesswork would be involved in generalizing this to a full SDE of the type $dX_t = \mu(X_t, t)dt + \sigma(X_t, t)dW_t$, let alone to multi-dimensional generalizations. In order to overcome this, it is worth investing in seeing how Itô's lemma plays a critical role in the derivation of the Feynman-Kac formula. This investment will also explain how (5) was arrived at. The details are in Appendices A and B.

2.2 One-dimensional Feynman-Kac formula

As we have seen, the Feynman-Kac technique is a strategy for solving a certain type of partial differential equations by using Brownian motion and Monte Carlo methods. The converse can also be quite useful as it implies that expectations of Wiener processes such as (2) or more general, which are stochastic, can be translated to and computed by purely deterministic methods. Let us now formalize these notions.

Let $(W_t)_{t \geq 0}$ be a one-dimensional Brownian motion defined on a filtered probability space $(\Omega, \mathcal{F}, \mathbb{P})$. We shall consider the Wiener process W_t and the stochastic differential equation

$$dX_t = \mathfrak{b}(X_t, t)dt + \sqrt{2\mathfrak{a}(X_t, t)}dW_t, \quad (7)$$

where \mathfrak{a} and \mathfrak{b} are suitable analytic functions with $\mathfrak{a} > 0$ and with condition $X_t = x$, see [31, §4]. Note how (7) now incorporates a drift term $\mathfrak{b}(X_t, t)$, unlike (2). Let $\mathfrak{f} : \mathbb{R} \rightarrow \mathbb{R}$ be an analytic function. For a Borel-measurable function $\psi : \mathbb{R} \rightarrow \mathbb{R}$, we define the function $u : \mathbb{R} \times [0, T] \rightarrow \mathbb{R}$ by setting

$$\begin{aligned} u(x, t) = \mathbb{E} \Big[& \int_t^T \exp \left(- \int_t^s \mathfrak{c}(X_\tau, \tau) d\tau \right) \mathfrak{f}(X_s, s) ds \\ & + \exp \left(- \int_t^T \mathfrak{c}(X_\tau, \tau) d\tau \right) \psi(X_T) \mid X_t = x \Big], \end{aligned} \quad (8)$$

where \mathfrak{c} is a suitable real function called the discount function. Moreover, suppose that

$$\int_{-\infty}^{\infty} e^{-\delta x^2} |\psi(x)| dx < \infty,$$

for some $\delta > 0$. The function u is defined for $0 < T - t < \frac{1}{2}\delta$ and $x \in \mathbb{R}$, and has derivatives of all orders. In particular, it belongs to the class $\mathcal{C}^{2,1}(\mathbb{R} \times (0, T))$. The Feynman-Kac result is the fact that u satisfies the following partial differential equation

$$\begin{cases} \frac{\partial u}{\partial t} + \mathfrak{a}(x, t) \frac{\partial^2 u}{\partial x^2} + \mathfrak{b}(x, t) \frac{\partial u}{\partial x} - \mathfrak{c}(x, t)u + \mathfrak{f}(x, t) = 0, & \text{for } t < T \\ u(x, T) = \psi(x). \end{cases} \quad (9)$$

From this point, we shall drop the term involving \mathfrak{f} by setting it to zero as it will not be very useful in our formulation. In this case, equation (9) becomes

$$\begin{cases} \frac{\partial u}{\partial t} + \mathfrak{a}(x, t) \frac{\partial^2 u}{\partial x^2} + \mathfrak{b}(x, t) \frac{\partial u}{\partial x} = \mathfrak{c}(x, t)u, & \text{for } t < T \\ u(x, T) = \psi(x). \end{cases} \quad (10)$$

and its associated conditional expectation (8) reduces to

$$u(x, t) = \mathbb{E} \Big[\exp \left(- \int_t^T \mathfrak{c}(X_\tau, \tau) d\tau \right) \psi(X_T) \mid X_t = x \Big]. \quad (11)$$

Shortly it will become clear that the term $\frac{\partial u}{\partial x}$ is not always helpful for our objectives and it is advantageous to remove it now by applying changes of variables and transformations. First we set $u(x, t) = e^{g(x, t)}v(x, t)$ where g is a function of \mathfrak{a} and \mathfrak{b} given (49) in Appendix C. Moreover, we switch to an initial value problem by setting $\tau = T - t$ and then perform a Wick rotation $\xi = -i\tau$, where i is the imaginary unit. The last step requires analytical continuation. This transforms the problem to

$$-i\frac{\partial v}{\partial \xi} = \mathfrak{a}(x, \xi)\frac{\partial^2 v}{\partial x^2} + \mathfrak{w}(x, \xi)v,$$

where the new term \mathfrak{w} can be written in terms of the other coefficients $\mathfrak{a}, \mathfrak{b}$ and \mathfrak{c} of the PDE by the formula

$$\mathfrak{w}(x, \xi) = -\frac{\partial}{\partial \xi} \int \frac{\mathfrak{b}(x, \xi)}{2\mathfrak{a}(x, \xi)} dx - \frac{\mathfrak{b}^2(x, \xi)}{4\mathfrak{a}(x, \xi)} - \frac{1}{2}\mathfrak{a}(x, \xi) \left(\frac{\partial}{\partial x} \frac{\mathfrak{b}(x, \xi)}{\mathfrak{a}(x, \xi)} \right) - \mathfrak{c}(x, \xi).$$

We shall now write this in Dirac notation as

$$-i\frac{\partial}{\partial \xi} |\psi\rangle = \hat{\mathcal{H}} |\psi\rangle, \quad (12)$$

where the wave function is written in Dirac notation as $\psi(x, t) = \langle x, t | \psi \rangle$ and it takes the role of the modified solution in the setting of an infinite dimensional Hilbert space. The resulting function $|\psi\rangle$ does not necessarily correspond to an ℓ_2 normalized wavefunction and $\hat{\mathcal{H}}$ is not necessarily Hermitian. The wave function $|\psi\rangle$ is not to be confused with the boundary condition ψ from (11). In this case, the Hamiltonian is given by the expression

$$\hat{\mathcal{H}} := \mathfrak{a}(x, \xi)\frac{\partial^2}{\partial x^2} + \mathfrak{w}(x, \xi)\mathbb{I},$$

where \mathbb{I} is the identity operator. This Hamiltonian is time-dependent as ξ appears in both \mathfrak{a} and \mathfrak{w} . The reason the term $\frac{\partial u}{\partial x}$ in (10) was undesirable now becomes more apparent. While performing a canonical quantization the momentum operator is taken to be $\hat{p}_x \rightarrow \frac{\hbar}{i}\frac{\partial}{\partial x}$. The differential operator $\frac{\partial}{\partial x}$ is skew-Hermitian, i.e. $\frac{\partial}{\partial x}^\dagger = -\frac{\partial}{\partial x}$, whereas the second order derivative is Hermitian, $\frac{\partial^2}{\partial x^2}^\dagger = \frac{\partial^2}{\partial x^2}$. Therefore, when we performed the Wick rotation, the absence of the derivative of first order leaves the condition of the Hermicity of $\hat{\mathcal{H}}$ solely on the term $\mathfrak{a}(x, \xi)\frac{\partial^2}{\partial x^2}$. In general this term will not be Hermitian unless \mathfrak{a} and $\frac{\partial^2}{\partial x^2}$ commute. However, for our case of interest in Section 4, we shall set \mathfrak{a} to be constant and this will result in our Hamiltonian being Hermitian. This means that the solution to the Schrödinger-type equation (12) is

$$|\psi(x, \xi)\rangle = \exp \left(i \int_0^\xi \hat{\mathcal{H}}(\chi) d\chi \right) |\psi(x, 0)\rangle. \quad (13)$$

If there were no ξ dependency in $\hat{\mathcal{H}}$, then (13) would reduce to $|\psi(x, \xi)\rangle = \exp(i\hat{\mathcal{H}}\xi) |\psi(x, 0)\rangle$.

The first order derivative in (10) is associated to the drift term $\mathfrak{b}(X_t, t)$ in (7). An SDE that has no dt term is called a strict local martingale [35, §2.3] or [34, 31]. Alternatively, transforming an initial SDE into a martingale also helps in obtaining a Wick rotated equation without the first order derivative.

In the real time evolution, the evolution operator $e^{-i\hat{\mathcal{H}}t}$ must be unitary. This operator is unitary if the Hamiltonian $\hat{\mathcal{H}}$ is Hermitian. In the imaginary time evolution, however, the evolution operator $e^{-\tau\hat{\mathcal{H}}}$ is not in general unitary and thus we do not need to be

concerned about the Hermiticity of $\hat{\mathcal{H}}$. The situation for non-Hermitian Hamiltonians is worth discussing. In [11, §2.1], a sequence of very judicious transformations and changes of variables transformed the Black-Scholes equation into the heat equation, see also [36, §6.5.1] as well as the Lamperti transform. The advantage is that the Wick-rotated heat equation becomes a Schrödinger equation with a Hermitian Hamiltonian. In [9, §II] (which does not employ variational methods), these changes were not applied and as a result the authors have to deal with a non-Hermitian Hamiltonian which gets embedded, by unitary dilation [37], into a larger space and then deal with probabilities of success when measuring the outcomes of quantum circuits. Moreover, in [10, §II] (which does make use variational methods), these transformation were not employed either and the author has to utilize similarity transformations motivated from supersymmetry [38] in order to obtain a Hermitian Hamiltonian. These techniques are indeed useful for situations where no change of variables or transformations lead to Hermitian Hamiltonians, see also [39] for useful insights.

2.3 Special cases

According to the values of the coefficients in (10), we can make the following formal identifications.

2.3.1 Feynman-Kac

- Coefficients: Suitable a, b, c with $f = 0$.
- Transformations: None.
- Interpretation: The solution is an expectation (11) of a Wiener process (7) involving Brownian motion.
- Notes: Most general case before specializing the coefficients.

2.3.2 Heat equation

- Coefficients: $a = 1, b = c = 0$.
- Transformations: None.
- Interpretation: The solution u corresponds to the temperature field and the Laplacian indicates whether the material surrounding each point x is hotter or colder, on average, than the material at the point x .
- Notes: Density and properties of materials can be taken into account by applying certain coefficients to the terms that appear in the equation. It is ℓ_1 -norm preserving.

2.3.3 Black-Scholes

- Coefficients: $a = \frac{1}{2}\sigma^2x^2, b = rx, c = r$.
- Transformations: $t \rightarrow T - t$.
- Interpretation: The solution is the price of an option whose underlying follows a geometric Brownian motion.
- Notes: Type of financial derivative depends on payoff boundary condition.

2.3.4 Schrödinger

- Coefficients: $\mathfrak{a} = \frac{\hbar^2}{2m}$, $\mathfrak{b} = 0$, $\mathfrak{c} = V(x)$, with $V(x)$ the potential.
- Transformations: $t \rightarrow -it$.
- Interpretation: The solution is interpreted as the amplitude of wave function in quantum mechanics.
- Notes: The resulting Hamiltonian $\hat{\mathcal{H}}$ is Hermitian and the equation is ℓ_2 -norm preserving.

2.3.5 Fokker–Planck

- Coefficients: $\mathfrak{a} = -D(x, t)$, $\mathfrak{b} = -2\frac{\partial D}{\partial x} + \mu$, $\mathfrak{c} = -\frac{\partial^2 D}{\partial x^2} + \frac{\partial \mu}{\partial x}$.
- Transformations: Here D is the diffusion coefficient given by $D(X, t) = \frac{1}{2}\sigma^2$.
- Interpretation: The solution $p(x, t)$ is the probability density function of the random variable X_t whose Itô process is $dX_t = \mu(X_t, t)dt + \sigma(X_t, t)dW_t$.
- Notes: Time evolution of the probability density function of a particle under the influence of drag forces and random forces. The equation is ℓ_1 -norm preserving.

2.3.6 Kolmogorov

- Coefficients: $\mathfrak{a} = D(x, t)$, $\mathfrak{b} = \mu(x, t)$.
- Transformations: See Fokker-Planck equation above.
- Interpretation: These equations are used to characterize stochastic processes by describing how they change over time.
- Notes: ‘Forward’ equation is Fokker-Planck, and ‘backward’ equation is used to determine the probability distribution of a state at a later time $s > t$. The equation is ℓ_1 -norm preserving.

2.3.7 Hamilton-Jacobi

- Coefficients: Use ansatz $\psi(x, t) = \psi_0 e^{iS/\hbar}$ where S is action on Schrödinger.
- Transformations: See Schrödinger equation above.
- Interpretation: Take $\hbar \rightarrow 0$ for limiting semi-classical case and use $S = \int L dt$, where L is the Lagrangian.
- Notes: This is used to derive equations of motion and identifying conserved quantities of mechanical systems. It is non-linear.

2.3.8 Ornstein - Uhlenbeck

- Coefficients: See Fokker-Planck above. In one dimension, Ornstein - Uhlenbeck satisfies $\frac{\partial D}{\partial t} = \theta \frac{\partial}{\partial x}(xP) + D \frac{\partial^2 P}{\partial x^2}$.
- Transformations: Here $P(x, t)$ is the probability of finding the process in the state x and time t and diffusion $D = \frac{\sigma^2}{2}$.

- Interpretation: The associated SDE is $dX_t = -\theta X_t dt + \sigma dW_t$ and it is frequently written as a Langevin equation of the type $\frac{dX_t}{dt} = -\theta X_t + \sigma \eta(t)$, where $\eta(t)$ is white noise. Note that dW_t/dt does not exist, as Brownian motion is not differentiable, so this is only heuristic.
- Notes: The Ornstein - Uhlenbeck process is a stationary Gauss-Markov process and it is temporarily homogeneous. Wide applications in finance (interest rate modeling), mechanical systems (overdamped processes) and biology (modeling of phenotypes).

Other kinds of formal identification in several dimensions also exist.

2.4 Multi-dimensional Feynman-Kac formula

Let $D \in \mathbb{N}$ and $N \in \mathbb{N}$ and consider a D -dimensional stochastic process

$$\mathbf{X}_t = (X_t^1, X_t^2, \dots, X_t^D)^T$$

governed by an N -dimensional Brownian motion

$$\mathbf{W}_t = (W_t^1, W_t^2, \dots, W_t^N)^T, \quad (14)$$

whose time evolution is given by the following system of SDEs:

$$dX_t^i = \mu_i(\mathbf{X}_t, t)dt + \sum_{\ell=1}^N \sigma_{i\ell}(\mathbf{X}_t, t)dW_t^\ell, \quad \text{for } i = 1, 2, \dots, D. \quad (15)$$

We may abbreviate this to

$$\begin{cases} d\mathbf{X}_t &= \mu(\mathbf{X}_t, t)dt + \Sigma \cdot d\mathbf{W}_t, \\ \mathbf{X}_0 &= \mathbf{x}_0, \end{cases}$$

where \mathbf{X}_t , \mathbf{x}_0 and μ are in \mathbb{R}^D , \mathbf{W}_t is in \mathbb{R}^N , and Σ is the $\mathbb{R}^{D \times N}$ matrix with elements $\sigma_{ik}(\mathbf{X}_t, t)$. The infinitesimal generator \mathcal{G} of the diffusion process \mathbf{X}_t is given by the differential operator (see e.g. [40, §4.1] as well as [41, §4.2])

$$\mathcal{G} = \frac{1}{2} \sum_{i=1}^D \sum_{j=1}^D \sum_{\ell=1}^N \sigma_{i\ell}(\mathbf{X}_t, t) \sigma_{j\ell}(\mathbf{X}_t, t) \frac{\partial^2}{\partial x_i \partial x_j} + \sum_{i=1}^D \mu_i(\mathbf{X}_t, t) \frac{\partial}{\partial x_i}. \quad (16)$$

We may further set $(\Sigma \Sigma^T)_{ij} = \sum_{\ell=1}^N \sigma_{i\ell}(\mathbf{X}_t, t) \sigma_{j\ell}(\mathbf{X}_t, t)$. The terms in \mathcal{G} correspond to the terms that appear in the multidimensional Itô formula

$$du(\mathbf{X}_t, t) = \frac{\partial u}{\partial t}(\mathbf{X}_t, t)dt + \mathcal{G}u(\mathbf{X}_t, t)dt + R(dW_t^j), \quad (17)$$

where R stands for the various stochastic terms involving the different dW_t^j . Thus, \mathcal{G} represents the deterministic, non-time dependent, part of (17). Now set

$$u(\mathbf{x}, t) = \mathbb{E} \left[\exp \left(- \int_0^t r(\mathbf{X}_s, s)ds \right) \psi(\mathbf{X}_t) \mid \mathbf{X}_t = \mathbf{x} \right], \quad (18)$$

for some function ψ satisfying appropriate Lipschitz and growth hypotheses [41, Theorem 6]. The function u defined by (18) satisfies the diffusion equation

$$\frac{\partial u}{\partial t} = \mathcal{G}u - ru, \quad t > 0 \quad (19)$$

as well as the initial condition $u(\mathbf{x}, 0) = \psi(\mathbf{x})$. This problem could be mapped to the corresponding Schrödinger-type equation

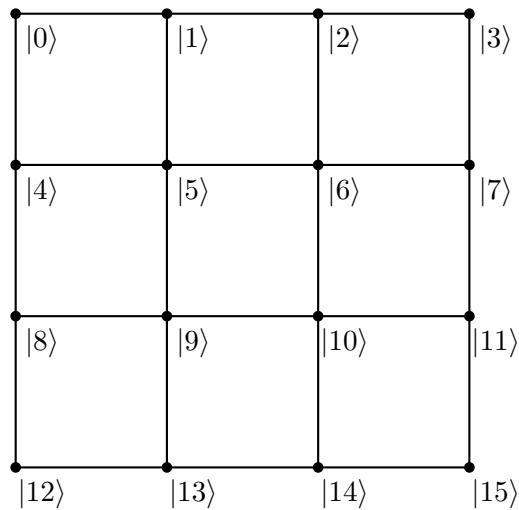
$$\frac{\partial}{\partial t} |\psi(t)\rangle = -i\tilde{\mathcal{G}} |\psi(t)\rangle \quad \text{where} \quad \tilde{\mathcal{G}} = \mathcal{G} - r\mathbb{I},$$

by introducing an appropriate Wick rotation, however $\tilde{\mathcal{G}}$ might not necessarily be Hermitian.

There is no need to impose correlations between the elements of the N -dimensional Brownian motion as we shall see in Section 4.1. Not only does this approach bypass the need to set individual correlations between Brownian motions, it also allows for substantial flexibility when modeling different particles whose behavior is governed by several Brownian motions $n = 1, 2, \dots, N$ per dimension $d = 1, 2, \dots, D$ as in (14). We further note that generalization we present here, in addition to multi Brownian motions, also contains the discount function r . None of these features appear in the previous literature of VarQITE and stochastic differential equations [10, 11, 12].

2.5 Finite difference approximation

We briefly explain the discretization and the embedding schemes needed to solve the Feynman-Kac system using a Schrödinger equation solver, such as VarQITE which will be described in Section 3. Let n_q be the total number of qubits. We demonstrate the discretization procedure for a simple problem of a two dimensional Feynman-Kac generator, as specified in (38) below which is a special case of (16), involving the second order derivatives $\frac{\partial^2}{\partial x^2}$, $\frac{\partial^2}{\partial y^2}$ and $\frac{\partial^2}{\partial x \partial y}$. To solve the system using $n = 4$ qubits (i.e. $2^4 = 16$ computational basis states: $|0\rangle, |1\rangle, \dots, |15\rangle$), we embed the solution u_{ij} in the quantum state as shown in the diagram below. This type of diagram in fact can be used to elucidate the embedding of a multidimensional differential operator on a spatial grid. The mesh is in general a hypercube in d dimensions, but if n_q is not divisible by D , then the mesh is a D dimensional hypercube with the sum of its dimensions equal to n_q .



Note that the size of the square mesh is $2^{\frac{n_q}{D}} = 2^2 = 4$, where n_q was the number of qubits and D is the number of dimensions. Considering the above mesh, the second order derivatives from (38) can be approximated by using the stencil method [42] as

$$\frac{\partial^2 u}{\partial x^2} \approx \frac{u_{i+1,j} - 2u_{i,j} + u_{i-1,j}}{\Delta x^2},$$

$$\begin{aligned}\frac{\partial^2 u}{\partial y^2} &\approx \frac{u_{i,j+1} - 2u_{i,j} + u_{i,j-1}}{\Delta y^2}, \\ \frac{\partial^2 u}{\partial x \partial y} &\approx \frac{u_{i+1,j+1} - u_{i+1,j-1} - u_{i-1,j+1} + u_{i-1,j-1}}{4\Delta x \Delta y}.\end{aligned}$$

Here Δx and Δy refer to the lattice spacing in x and y directions respectively. We additionally employ periodic boundary conditions on the grid in both directions.

The second order derivatives appearing in the general case of D dimensional Feynman-Kac operator (16) can be approximated as

$$\begin{aligned}\frac{\partial u(x_{i_1}, x_{i_2}, \dots, x_{i_D}, t)}{\partial x_{i_d}} &\approx \frac{u_{\mathbf{i}+\mathbf{e}_d} - u_{\mathbf{i}-\mathbf{e}_d}}{2\Delta x_{i_d}}, \\ \frac{\partial^2 u(x_{i_1}, x_{i_2}, \dots, x_{i_D}, t)}{\partial x_{i_d}^2} &\approx \frac{u_{\mathbf{i}+\mathbf{e}_d} - 2u_{\mathbf{i}} + u_{\mathbf{i}-\mathbf{e}_d}}{\Delta x_{i_d}^2}, \\ \frac{\partial^2 u(x_{i_1}, x_{i_2}, \dots, x_{i_D}, t)}{\partial x_{i_d} \partial x_{i_{d'}}} &\approx \frac{u_{\mathbf{i}+\mathbf{e}_d+\mathbf{e}_{d'}} - u_{\mathbf{i}+\mathbf{e}_d-\mathbf{e}_{d'}} - u_{\mathbf{i}-\mathbf{e}_d+\mathbf{e}_{d'}} + u_{\mathbf{i}-\mathbf{e}_d-\mathbf{e}_{d'}}}{4\Delta x_{i_d} \Delta x_{i_{d'}}},\end{aligned}\quad (20)$$

where $\mathbf{i} = \{i_1, i_2, \dots, i_D\}$ and $\mathbf{i} \pm \mathbf{e}_d = \{i_1, \dots, i_{d-1}, i_d \pm 1, i_{d+1}, \dots, i_D\}$ and $\mathbf{i} \pm \mathbf{e}_d \pm \mathbf{e}_{d'} = \{i_1, \dots, i_{d-1}, i_d \pm 1, i_{d+1}, \dots, i_{d'-1}, i_{d'} \pm 1, i_{d'+1}, \dots, i_D\}$. Similarly if one intends to consider periodic boundary conditions then appropriate modular operations should be taken into account. Using the approximations above we obtain the discretized form for the operator $\tilde{\mathcal{G}}$ as

$$\tilde{\mathcal{G}} = \sum_{i_1=0}^{n_m-1} \sum_{i_2=0}^{n_m-1} \cdots \sum_{i_D=0}^{n_m-1} \sum_{j_1=0}^{n_m-1} \sum_{j_2=0}^{n_m-1} \cdots \sum_{j_D=0}^{n_m-1} [\hat{\mathcal{G}}]_{i_1, i_2, \dots, i_D, j_1, j_2, \dots, j_D} |\mathbf{i}\rangle \langle \mathbf{j}|,\quad (21)$$

where $n_m = 2^{n_q/D}$. The elements of the matrix are given by

$$[\tilde{\mathcal{G}}]_{\mathbf{i}, \mathbf{j}} = \begin{cases} \left(\sum_{d=1}^D \frac{(\Sigma \Sigma^T)_{i_d i_d}}{\Delta x_{i_d}^2} \right) - r & (\mathbf{i} = \mathbf{j}), \\ \frac{1}{2} \frac{(\Sigma \Sigma^T)_{i_d i_d}}{(\Delta x_{i_d})^2} \mp \frac{1}{2} \frac{\mu_{i_d}}{\Delta x_{i_d}} & (\mathbf{i} = \mathbf{j} \pm \mathbf{e}_d), \\ \frac{1}{4} \frac{(\Sigma \Sigma^T)_{i_d i_{d'}}}{(\Delta x_{i_d} \Delta x_{i_{d'}})} & (\mathbf{i} = \mathbf{j} \pm \mathbf{e}_d \pm \mathbf{e}_{d'}), \\ -\frac{1}{4} \frac{(\Sigma \Sigma^T)_{i_d i_{d'}}}{(\Delta x_{i_d} \Delta x_{i_{d'}})} & (\mathbf{i} = \mathbf{j} \pm \mathbf{e}_d \mp \mathbf{e}_{d'}), \\ 0 & \text{otherwise.} \end{cases}\quad (22)$$

3 Quantum algorithm for solving the Feynman-Kac formula

In this section, we describe a quantum algorithm for solving the PDE associated to the SDE of interest as prescribed within the Feynman-Kac formulation. We first discuss variational quantum imaginary time evolution (VarQITE), i.e., the fundamental algorithm used, then how to embed the problems into quantum states and the efficient decomposition of the differential operators of the PDE into unitaries. After that, we discuss how to extract properties of the solution, and lastly how to (approximately) enforce the ℓ_1 norm preservation during the evolution.

3.1 VarQITE and state embedding

Suppose we have a time-independent differential operator \mathcal{E} acting on n qubits with associated propagator $\exp(-i\mathcal{E}t)$, evolving for real values of t . Provided the operator \mathcal{E} is

Hermitian, the quantum real time evolution (QRTE) is unitary and can be implemented, e.g., by a suitable Trotter decomposition, solving the Schrödinger equation.

If we apply the Wick rotation $\tau = it$, i.e., we translate the problem to the realm of quantum imaginary time evolution (QITE), the evolution operator $\exp(-\mathcal{E}\tau)$ will not be a unitary operation anymore and the underlying equation is more complicated to (approximately) solve on a quantum computer. If we are given an initial state $|\psi(0)\rangle$, then the normalized imaginary time evolution is defined by $|\psi(\tau)\rangle = \alpha(\tau)e^{-\mathcal{E}\tau}|\psi(0)\rangle$, where $\alpha(\tau) = 1/\sqrt{\langle\psi(0)|e^{-2\mathcal{E}\tau}|\psi(0)\rangle}$ is the normalization factor with respect to the ℓ_2 norm. The corresponding Wick-rotated Schrödinger equation is

$$\frac{\partial |\psi(\tau)\rangle}{\partial \tau} = -(\mathcal{E} - \lambda_\tau \mathbb{I}) |\psi(\tau)\rangle, \quad (23)$$

where $\lambda_\tau = \langle\psi(\tau)|\mathcal{E}|\psi(\tau)\rangle$ ensures the normalization.

VarQITE is an algorithm to approximate QITE, see [43, 12, 26, 11, 44, 27]. Instead of evolving quantum states in the complete exponential state space, a parameterized ansatz is used and the evolution is approximated by mapping it to the ansatz parameters via McLachlan's variational principle [45]. The main advantage is the possibility of implementing it using shallow quantum circuits, suitable for near-term quantum devices [44, p. 1]. More precisely, instead of considering a general state $|\psi(\tau)\rangle$, we encode the trial state $|\phi(\boldsymbol{\theta}(\tau))\rangle$, with $\boldsymbol{\theta}(\tau) = (\theta_1(\tau), \theta_2(\tau), \dots, \theta_N(\tau))$, where N is an integer depending on the ansatz circuit of our choice. The choice of the ansatz is crucial for the performance of the algorithm and we discuss this for our concrete example in Sec. 4.2.

Instead of introducing standard VarQITE for quantum states, we now discuss the evolution for states that are possibly not ℓ_2 normalized [26]. In the situation of an unnormalized state $|\tilde{\psi}(\mathbf{x}, t)\rangle$ governed by the equation

$$\frac{\partial}{\partial t} |\tilde{\psi}(\mathbf{x}, t)\rangle = \mathcal{E}(t) |\tilde{\psi}(\mathbf{x}, t)\rangle, \quad (24)$$

where $\mathcal{E}(t)$ is now a linear time-dependent –not necessarily Hermitian– operator. The dynamical evolution of $|\tilde{\psi}(\mathbf{x}, t)\rangle$ can be simulated by introducing an ansatz $|\tilde{v}(\boldsymbol{\theta}(t))\rangle = \alpha(t) |v(\boldsymbol{\theta}(t))\rangle$ of the form $|v(\boldsymbol{\theta}(t))\rangle = \mathbf{G}(\boldsymbol{\theta}(t)) |0\rangle^{\otimes n}$ where $\alpha(t)$ is a parameter of the ansatz that scales the ℓ_2 normalized quantum state to the desired scale and $\mathbf{G}(\boldsymbol{\theta}(t)) = \prod_{i=1}^N \mathbf{G}_i(\theta_i(t))$ is the product of N parametric unitaries \mathbf{G}_i , each composed of one parametric rotation gates $e^{i\theta_k \mathfrak{G}_k}$ with $\mathfrak{G}_k^\dagger = \mathfrak{G}_k$.

To produce the VarQITE evolution we need to replace $|\tilde{\psi}(\mathbf{x}, t)\rangle$ by $|\tilde{v}(\boldsymbol{\theta}(t))\rangle$ in (24) by first mapping the dynamics of the quantum state to the dynamics of the ansatz. McLachlan's variational principle

$$\delta \left\| \frac{\partial}{\partial t} |\tilde{v}(\boldsymbol{\theta}(t))\rangle - \mathcal{E}(t) |\tilde{v}(\boldsymbol{\theta}(t))\rangle \right\| = 0 \quad (25)$$

yields the Euler-Lagrange-type of equations [27]

$$\sum_{j=0}^N M_{k,j} \dot{\theta}_k = V_k, \quad \text{for each } k = 0, 1, \dots, N. \quad (26)$$

Here the elements of the matrix $M_{k,j}$ are given by

$$M_{k,j} = \text{Re} \left(\alpha^2(t) \frac{\partial \langle v(\boldsymbol{\theta}(t)) |}{\partial \theta_k} \frac{\partial |v(\boldsymbol{\theta}(t))\rangle}{\partial \theta_j} \right), \quad \text{for } k, j \neq 0,$$

as well as

$$M_{0,j} = M_{j,0} = \alpha(t) \operatorname{Re} \left(\langle v(\boldsymbol{\theta}(t)) | \frac{\partial |v(\boldsymbol{\theta}(t))\rangle}{\partial \theta_j} \right), \quad \text{for } j > 0,$$

and $M_{0,0} = 1$. This matrix is fully dependent on the ansatz circuit. The vector V_k , however, depends on the operator \mathcal{E} as follows

$$V_k = \alpha(t) \operatorname{Re} \left(\frac{\partial \langle v(\boldsymbol{\theta}(t)) |}{\partial \theta_k} \mathcal{E} |v(\boldsymbol{\theta}(t))\rangle \right), \quad \text{for } k > 0,$$

and the very first term gets simplified to $V_0 = \operatorname{Re}(\langle v(\boldsymbol{\theta}(t)) | \mathcal{E} |v(\boldsymbol{\theta}(t))\rangle)$. Notably, one may have to include so-called phase-fix terms if the global phase of the state evolution underlying \mathcal{E} can change. These terms are explained in [27, 28]. Since $|v(\boldsymbol{\theta}(t))\rangle$ is implemented with parameterized unitaries, the terms $M_{k,j}$ and V_k can be computed parametrically by using the quantum circuit shown in Figure 2.

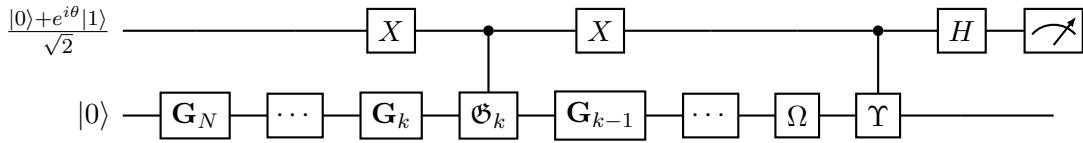


Figure 2: This quantum circuit evaluates the matrix elements $M_{k,j}$ and the vector elements V_k depending on Ω and Υ .

The extraction of the matrix elements $M_{k,j}$ and the vector elements V_k is as follows.

- One takes the sequence of gates \mathbf{G}_{k-1} to Ω to be \mathbf{G}_{k-1} to \mathbf{G}_j and $\Upsilon = \mathfrak{G}_j$, if we are interested in $M_{k,j}$,
- or one sets the sequence of \mathbf{G}_{k-1} to Ω to be \mathbf{G}_{k-1} to \mathbf{G}_1 and then $\Upsilon = U_j$, where U_n are easily-implementable unitary operators such as multi-qubit Pauli operators as represented in (27) below, if we are interested in V_k .

Further details on the respective implementation are given in [27, 44, 28].

From (26) and using the above formulas for $M_{k,j}$ and V_k , we can now obtain the evolution of the parameters $\dot{\theta}_k$ using the forward Euler method

$$\boldsymbol{\theta}(\tau + \delta\tau) \sim \boldsymbol{\theta}(\tau) + \dot{\boldsymbol{\theta}}\delta\tau = \boldsymbol{\theta}(\tau) + \mathbf{M}^{-1}(\tau) \cdot \mathbf{V}\delta\tau.$$

That is, McLachlan's principle defines an ODE which can be numerically solved with an arbitrary ODE solver with $N_T = \tau/\delta\tau$ time steps. We would like to point out that the choice of the time step size $\delta\tau$ is crucial to achieve reasonable results. Although, it is generally preferable to choose rather smaller time steps one has to consider the trade off between accuracy and computational cost. The use of an ODE solver that implements an adaptive step size scheme such as Runge-Kutta can help to find reasonable time steps. In certain cases the matrix \mathbf{M} will not be well-conditioned and thus we will need to invoke the Moore-Penrose inverse or use least squares, see e.g. [11, 44]. Notably, the construction cost of the terms \mathbf{M} and \mathbf{V} scales quadratically in the number of circuit parameters and linearly in the number of Hamiltonian terms. Furthermore, the computational complexity of solving the underlying system of linear equations scales generally as $\mathcal{O}(N^3)$. This complexity comes from standard Moore-Penrose inverse calculations and solving systems of linear equations where the matrix has dimensions $N \times N$ and the vector on the right hand side has dimension N .

One has to decompose the operator $\mathcal{E}(t)$ as a sum of tensor products of Pauli matrices

$$\mathcal{E}(t) = \sum_{h=1}^H \lambda_h U_h, \quad (27)$$

where $\lambda_h \in \mathbb{C}$ and U_h is an easily implementable unitary operator made of Pauli operators. This requires an evaluation of $\mathcal{O}(N^2) + \mathcal{O}(NH)$ different quantum circuits. The circuits themselves will each require an ancilla qubit in addition to the qubits needed for the dynamics. This implies that in order to avoid undesirable growth of resources, the VarQITE evolution must be feasible with depths of the unitaries no more than polynomial in n , i.e. the depth of the tensor products of Pauli matrices must be $\mathcal{O}(\text{poly}(n))$. This will be explained in Section 3.2.

Different manifestations of the Feynman-Kac equation will result in situations where the solution to the PDE might correspond to a probability distribution. To enable this we need to map the probabilities to amplitudes. Since quantum states have to be normalized with respect to the ℓ_2 norm, while probabilities are normalized with respect to the ℓ_1 norm, we then need to re-scale the state to handle this mismatch. In other words, the considered states have the form

$$|\tilde{\psi}(t)\rangle = \alpha(t) |\psi(t)\rangle, \quad |\tilde{\psi}(t)\rangle = \sum_{i=0}^{2^n-1} p_i(t) |i\rangle \quad \text{with} \quad p_i(t) = u(x_i, t) = \mathbb{P}[X(t) = x_i], \quad (28)$$

where X is the random variable whose probability distribution function is given by ℓ_1 -normalizable function u , and $\alpha(t)$ is given by the ℓ_2 norm

$$\alpha(t) = \left(\sum_{i=0}^{2^n-1} p_i^2(t) \right)^{1/2}.$$

This means that we embed the probability density into a state-vector which is re-scaled with an additional scalar parameter. This will result in a bijection. Moreover, the scalar parameter $\alpha(t)$ will be provided by the parameterized quantum circuit. It must be noted that indeed this differs from the traditional ℓ_2 embedding which is of the form

$$|\psi(t)\rangle = \sum_{i=0}^{2^n-1} \sqrt{p_i(t)} |i\rangle.$$

Because of this difference of embedding, extraction of quantities involving p_i , such as the moments, need to be re-derived in this new context, see Section 3.3 and Appendix G for further details.

Suppose we are confronted with solving (19). Obtaining the full distribution $u(x_i, t) = p_i(t)$ as its solution is the most complete information, but in general quantum algorithms cannot provide this type of result unless full-state tomography is employed, which is an expensive process [46]. However, there could be circumstances when knowing the first and second moments (say) of the solution, with quadratic speedup, might be advantageous enough. This also raises the question of approximating a probability distribution from (a few of) its moments. Some of the instances where the moments of the solution are useful are listed below:

- The type of heat equation in (19) describes the diffusion of point particles. In this case $u = u(x, t)$, for example in one spatial dimension with $u : \Omega \rightarrow \mathbb{R}$, describes

the density of particles at a certain point x at time t . The first moment $\int_{\Omega} u(x, t) dx$ is proportional to the total number of particles. If we consider a closed system, which is realized by imposing Neumann boundary conditions, then the conservation of energy means conservation of mass. If Neumann boundary conditions are not imposed, then the total mass is not relevant and it is sensible to instead define the moment $F[u(\cdot, t)] := \int_{\Omega} u^2(x, t) dx$. Here F is known as the *energy functional* for the heat equation and it measures the distance, in ℓ_2 -norm, from the equilibrium solution $u = 0$. Moreover, F is related to the *Dirichlet energy* $E[u(\cdot, t)] := \int_{\Omega} (\partial_x u)^2 dx$, see [47, §2.2.5 and §2.3.4], as follows. For solutions of the heat equation $u_t = ku_{xx}$ integration by parts yields

$$\begin{aligned} \frac{d}{dt} F[u(\cdot, t)] &= 2 \int_{\Omega} u(x, t) u_t(x, t) dx = 2k \int_{\Omega} u(x, t) u_{xx}(x, t) dx \\ &= -2k \int_{\Omega} (u_x(x, t))^2 dx = -2k E[u(\cdot, t)]. \end{aligned} \quad (29)$$

When we integrate over t , the functional $F[u(\cdot, t)]$ describes the accumulated loss of Dirichlet energy for $t > 0$ and equals our scaling factor $\alpha^2(t)$. The time derivative can be approximated by employing finite differences as we have the time evolution of the proxy $\alpha(t)$. Dirichlet energy plays an important role in soft mapping [48], machine learning [49] and discrete differential geometry [50, 51].

- Suppose that q and p are the coordinate and momenta respectively of a wave function and that $\sigma(x) = \sqrt{v(x)}$ is the standard deviation of the observable \hat{x} , where $v(x) = \mu(x - \mu(x))^2$ is the variance and $\mu(x)$ is the mean of \hat{x} . Reasoning from the Heisenberg uncertainty principle $\Delta p \cdot \Delta x \approx \hbar$ or $\sigma(p) \cdot \sigma(q) \geq \frac{1}{2}\hbar$ in [52], the *uncertainty length* is defined as $\delta[x] := (\int_{\mathbb{R}} p(x)^2 dx)^{-1}$, where $p(x)$ is the probability distribution of the stochastic variable \hat{x} . This is also known as the *Süssmann measure* and it can be interpreted as the total length of all the intervals that produce rectangles of total areas $h_x \delta[x] = 1$. Here the height h_x represents the mean value of the normalized $p(x)$ with $p(x)$ itself as its own weight factor. Its usefulness rests on the fact that other quantities related to width, like variance, might not necessarily exist for certain distributions (e.g. the Cauchy distribution) whereas the Süssmann measure will. The inverse of the Süssmann measure can be thought of as the average height of the distribution.
- The quantum mechanical observables corresponding to Hermitian operators \hat{O} acting on the wave function $|\psi\rangle = |\psi(x, t)\rangle$ obtained via solving Schrödinger equation (12) or the Wick-rotated Schrödinger equation (23), are expectation values of the form $\langle \psi | \hat{O} | \psi \rangle$. Thus, all such properties are addressable via the algorithm described in this work and the references mentioned herein.
- Let us suppose we have an educated belief that the resulting distribution is normal, or a function thereof, e.g. lognormal. One plausible reason for this belief is that the Central Limit Theorem gives the normal distribution a special status and it is characterised by μ and σ . Furthermore, as will be shown in Section 3.3 the evolution of the Feynman-Kac system needs to be evaluated only once to extract the final circuit parameters. Once these circuit parameters are obtained, additional higher order moments can be evaluated in parallel using several quantum devices, in those cases where skewness and kurtosis may play significant role in the description of the solution. As the dimension of the problem increases, so will the overhead. Lastly, for

time-critical applications one might want to know the first moments using a quantum device while waiting for the full distribution to be computed classically.

3.2 Decomposition of the generator

In order to implement VarQITE, we need to employ the matrix M and vector V from (26). We shall start with only one dimension. First, we need to decompose the operator \mathcal{G} into unitaries. The following procedure is borrowed from [12][§III.C] and then adapted to our operator of interest \mathcal{G} given by (38) below. Start by defining the operators

$$V_+(n) := \sum_{i=0}^{2^n-2} |i+1\rangle\langle i| \quad \text{and} \quad V_-(n) := \sum_{i=0}^{2^n-1} |i-1\rangle\langle i|. \quad (30)$$

If we make the boundary identifications $| -1 \rangle \leftrightarrow | 2^n - 1 \rangle$ and $| 2^n \rangle \leftrightarrow | 0 \rangle$, then the operators V_{\pm} can be manufactured from the n -qubit operators $\text{Cyc}_{\pm}(n) := \sum_{i=0}^{2^n-1} |i \pm 1\rangle\langle i|$. As per [53], these operators can be implemented as a product of $\mathcal{O}(n)$ Toffoli, CNOT, and X gates with $\mathcal{O}(n)$ ancilla qubits. The explicit construction of V_{\pm} from Cyc_{\pm} is achieved by defining an n -qubit-control gate $C^{n-1}Z := \sum_{i=0}^{2^n-2} |i\rangle\langle i| - |2^n-1\rangle\langle 2^n-1|$. According to [54], then implementation of $C^{n-1}Z$ can be achieved as a product of $\mathcal{O}(n^2)$ Toffoli, CNOT and other single-qubit gates. Using the identity $\frac{1}{2}(C^{n-1}Z + I^{\otimes n}) = \sum_{i=0}^{2^n-2} |i\rangle\langle i|$, one can prove that the V 's from (30) and the Cyc's are related by

$$V_+(n) = \text{Cyc}_+(n) \cdot \frac{1}{2}(C^{n-1}Z + I^{\otimes n}) \quad \text{and} \quad V_-(n) = \frac{1}{2}(C^{n-1}Z + I^{\otimes n}) \cdot \text{Cyc}_-(n).$$

Thus we can decompose V_{\pm} into a sum of two unitaries made of $\mathcal{O}(n^2)$ few-qubit gates. Lastly, we bring in an operator composed of weighted projectors

$$D(n) := \sum_{i=0}^{2^n-1} i|i\rangle\langle i| = \frac{2^n-1}{2} I^{\otimes n} - \sum_{i=1}^n 2^{n-i-1} Z_i, \quad (31)$$

where Z_i is a Z gate acting on the i th qubit. This implies that $D(n)$ is now a sum of $\mathcal{O}(n)$ unitaries composed of a single-qubit gate. Utilizing these operators we end up with

$$V_+(n)(D(n))^m = \sum_{i=0}^{2^n-2} i^m |i+1\rangle\langle i| \quad \text{and} \quad V_-(n)(D(n))^m = \sum_{i=1}^{2^n-1} i^m |i-1\rangle\langle i|.$$

For example, let us suppose we have an operator \mathcal{F} whose decomposition is given by

$$\begin{aligned} \mathcal{F}(t) &:= \sum_{i,k=0}^{2^n-1} (\mathcal{F}(t))_{i,k} |i\rangle\langle k| \\ &= \sum_{i=0}^{2^n-2} f_1(x_i, t) |i+1\rangle\langle i| + \sum_{i=1}^{2^n-1} f_2(x_i, t) |i-1\rangle\langle i| + \sum_{i=0}^{2^n-1} f_3(x_i, t) |i\rangle\langle i|, \end{aligned}$$

with all other matrix elements $(\mathcal{F}(t))_{i,k}$ being zero, which is a typical situation from the stencil method we are implementing in our formulation. Then we decompose the functions f_j into Taylor expansions

$$f_j(x_i, t) = \sum_m \frac{f_j^{(m)}(0, t)}{m!} x_i^m, \quad j \in \{1, 2, 3\},$$

where $f^{(m)}$ denotes the m th derivative. Using $x_i = \Delta x \cdot i$ allows us to write

$$\begin{aligned}\mathcal{F}(t) &= \sum_m \left(\sum_{i=0}^{2^n-2} \frac{f_1^{(m)}(0, t)}{m!} (\Delta xi)^m |i+1\rangle\langle i| + \sum_{i=1}^{2^n-1} \frac{f_2^{(m)}(0, t)}{m!} (\Delta xi)^m |i-1\rangle\langle i| \right. \\ &\quad \left. + \sum_{i=0}^{2^n-1} \frac{f_3^{(m)}(0, t)}{m!} (\Delta xi)^m |i\rangle\langle i| \right) \\ &= \sum_m \left(\frac{f_1^{(m)}(0, t)}{m!} (\Delta x)^m V_+(n)(D(n))^m + \frac{f_2^{(m)}(0, t)}{m!} (\Delta x)^m V_-(n)(D(n))^m \right. \\ &\quad \left. + \frac{f_3^{(m)}(0, t)}{m!} (\Delta x)^m (D(n))^m \right).\end{aligned}$$

The operators $V_+(n)(D(n))^m$, $V_-(n)(D(n))^m$ and $(D(n))^m$ are all made of the sum of $\mathcal{O}(n^m)$ unitaries. Each of these unitaries is made of $\mathcal{O}(n^2)$ few-qubit gates.

Our next step is to now generalize this to multiple dimensions $(x_1, x_2, \dots, x_D)^T$. This generalization builds up naturally from the one dimensional situation we just described above. In this case, we use multivariate Taylor expansions and the operators $V_\pm^{(k)}(n) := I^{\otimes k-1} \otimes V_\pm(n) \otimes I^{\otimes D-k}$ and $D^{(k)}(n) := I^{\otimes k-1} \otimes D(n) \otimes I^{\otimes D-k}$. These operators will then satisfy

$$V_\pm(n)(D^{(k)}(n))^m = \sum_{i_k=0}^{2^n-2} \sum_{\substack{i_m \in \{0, 1, \dots, 2^n-1\} \\ m \neq k}} i_k^m |\mathbf{i} \pm \mathbf{e}_k\rangle\langle \mathbf{i}|, \quad (32)$$

as well as

$$V_\pm(n)(D^{(k)}(n))^{m_k} V_\pm(n)(D^{(\ell)}(n))^{m_\ell} = \sum_{i_k=0}^{2^n-2} \sum_{i_\ell=0}^{2^n-2} \sum_{\substack{i_m \in \{0, 1, \dots, 2^n-1\} \\ m \neq k \\ m \neq \ell}} i_k^{m_k} i_\ell^{m_\ell} |\mathbf{i} \pm \mathbf{e}_k \pm \mathbf{e}_\ell\rangle\langle \mathbf{i}|. \quad (33)$$

Expansions for different combinations of V 's and different projectors can be obtained by similar methods. With (32) and (33) we may now decompose the generator $\hat{\mathcal{G}}$ given by (21) and (22) into easily implementable unitaries composed of few-qubit gates and thereby generate a feasible VarQITE. Specifically, the operators $V_\pm(n)(D^k(n))^m$ are the sums of $\mathcal{O}(n^m)$ unitaries, like in the one dimensional case, each composed of $\mathcal{O}(n^2)$ few-qubits gates. This procedures scales polynomially in the number of qubits n due to the truncation of the Taylor approximation at a fixed order m .

3.3 Readout and efficiency

In this section we discuss how to readout results of the solution of a PDE associated to an SDE as well as the resulting total algorithmic complexity. Further, we put this into context and identify the situations where this can lead to a potential quantum advantage.

Before we go into the details of the presented algorithm, we discuss the complexity of classical methods. Suppose we want to solve an SDE in D dimensions and time $t \in [0, T]$ or the corresponding PDE. Starting with the SDE, we can run a Monte Carlo simulation to generate different scenarios and estimate the solution to the SDE. To achieve a target accuracy $\varepsilon > 0$ this will scale as $\mathcal{O}(T/\varepsilon^2)$. For small ε and complicated SDEs this can

become prohibitive. An alternative to Monte Carlo simulation is considering the corresponding PDE, discretizing the state space and using a classical PDE solver. We assume here that a discretization of the state space using 2^n grid points, i.e., with $2^{n/D}$ grid points per dimension, is sufficient to reach the target accuracy. Then we assume that the total complexity would scale as $\mathcal{O}(T2^n)$ to access the final state and evaluate properties of interest of that state, where we assumed the generator \mathcal{G} corresponds to a sparse matrix such that matrix-vector products scale linearly in the dimension and that the number of iterations needed by the ODE solver scales linearly in T .

The quantum algorithm introduced in this paper is closely related to the latter approach and in the following we compare a potential advantage between those two. The comparison between the classical and quantum PDE solvers is not completely fair: the classical solver yields the complete state, while the quantum solver only gives access to some properties of the solution. The classical algorithm allows us to control and evaluate the accuracy, while the quantum algorithm is a heuristic unless the ansatz is universal, which would defeat the purpose as this would scale exponentially in the number of qubits. We ignore these differences in the remainder of this section, and thus, the following can be seen as necessary – but not necessarily sufficient – conditions for a quantum advantage.

While obtaining the full solution to the PDE generated by the presented algorithm is not feasible, we may try to extract certain properties of interest. Here we need to distinguish between two cases: when we solve our problem forward in time, as, e.g., for the heat equation, or backward in time, as for pricing financial options. The two situations imply differences in how the initial states are prepared and how the results are read out. In the following, we focus on the readout and its efficiency when solving the problem forward in time. The case of solving a PDE backward in time using the proposed method is discussed in Appendix E.

Suppose the solution to the PDE is given by a scaled n -qubit quantum state $|\tilde{\psi}\rangle = \alpha |\psi\rangle = \sum_{i=0}^{2^n-1} p_i |i\rangle$, i.e., the scaled amplitudes correspond to the elements of the discretized solution to the considered PDE. We ignore in the following that we only approximate this state. In [12] the authors discuss how to estimate properties of the form

$$\left| \sum_{i=0}^{2^n-1} p_i f(x_i) \right|, \quad (34)$$

where f is some function defined on a grid x_i . If the p_i correspond to a probability distribution of a random variable X (and for simplicity assuming $f(x) \geq 0$), then this results in an expected value $\mathbb{E}[f(X)]$.

This can be achieved if we can efficiently construct an operator S_f such that

$$S_f |0\rangle = \sum_{i=0}^{2^n-1} f(x_i) |i\rangle,$$

or by a weighted sum of such operators [12]. Note that this imposes an ℓ_2 normalization constraint on the function f , i.e. $\sum_{i=0}^{2^n-1} |f(x_i)|^2 = 1$. If f is not ℓ_2 normalized, the normalization factor needs to be multiplied on the result later on, amplifying the estimation error, as we discuss later in this section. The operator S_f can then be used to construct

$$\langle \tilde{\psi} | S_f |0\rangle \langle 0| S_f^\dagger | \tilde{\psi} \rangle = \alpha^2 \langle \psi | S_f |0\rangle \langle 0| S_f^\dagger | \psi \rangle = \left| \sum_{i=0}^{2^n-1} p_i f(x_i) \right|^2. \quad (35)$$

Since

$$|0\rangle \langle 0| = (\mathbb{I} - X^{\otimes n} \cdot C^{n-1} Z \cdot X^{\otimes n})/2,$$

we can estimate (35) as

$$\alpha^2(1 - \langle\psi|S_f X^{\otimes n} \cdot C^{n-1} Z \cdot X^{\otimes n} S_f^\dagger |\psi\rangle)/2,$$

i.e., by estimating the expected value of $C^{n-1}Z$ for the state $X^{\otimes n}S_f^\dagger|\psi\rangle$. As the authors of [12] point out, this can be achieved, for instance, up to accuracy $\varepsilon > 0$ with a Hadamard test, using $\mathcal{O}(1/\varepsilon^2)$ samples or with Quantum Phase Estimation (QPE), using $\mathcal{O}(\log(1/\varepsilon))$ samples but with a $\mathcal{O}(1/\varepsilon)$ -times longer circuit. Note that we evaluate the square of the expected value using the quantum computer and then multiply the result by α^2 . Thus, to achieve an overall accuracy ε for $\mathbb{E}[f(X)]$, we need to estimate the expected value with an accuracy $\mathcal{O}(\varepsilon/\alpha)$ [12]. In the case of the p_i being probabilities, it follows from the relation between ℓ_1 and ℓ_2 norms that $\alpha \in [1/\sqrt{2^n}, 1]$, i.e., α may have a positive impact on the estimation error.

Although this looks promising, there are a couple of potential pitfalls that can diminish or at least significantly reduce the potential quantum advantage:

1. The operator S_f is often difficult to construct and in general requires a number of terms exponential in the represented dimension that usually scales linearly in the number of qubits.
2. The function f needs to be ℓ_2 -normalized, that is we estimate $f/\|f\|_2$ for a non-normalized function. When going back to the original scale, the resulting estimation error will be $\|f\|_2\varepsilon$. Thus, the expected value needs to be estimated more accurately to compensate that scaling. This factor is maximal for constant functions, where it scales as $\|f\|_\infty\sqrt{2^n}$. With the Hadamard test, this implies $\mathcal{O}(2^n)$ samples which would diminish the quantum advantage, while QPE would at least achieve a quadratic advantage, depending on $\|f\|_\infty$.
3. Since $\alpha \in [1/\sqrt{2^n}, 1]$ (for probability distributions) it may help to partially compensate the error amplification introduced by the normalization of f .

These points are critical to be taken into account and determine the potential advantage of the presented technique, and are often ignored in the related literature. For solving PDEs backwards in time, similar phenomena appear and we discuss them in more depth in Appendix E.

For illustration, we assume the solution represents the probability distribution p of a D -dimensional random variable X and we will focus on how to estimate its expected value $\mathbb{E}[X]$, for more details, see Appendix G. To this extent we assume that the scaled quantum state $\alpha|\psi\rangle$ on n qubits represents the final solution, i.e., we represent each dimension with n/D qubits.

Further, we assume an n/D -qubit operator

$$S_x : |0\rangle \mapsto \frac{1}{C} \sum_{x=0}^{2^{n/D}-1} x|x\rangle,$$

i.e., we set $f(x) = x$, and

$$C = \sqrt{\sum_{x=0}^{2^{n/D}-1} x^2} = \theta(\sqrt{(2^{3n/D})}) = \theta(\|f\|_\infty \sqrt{(2^{n/D})})$$

is the required normalization constant. There are different ways to construct S_x , e.g., via a piece-wise constant approximation as suggested in [12] or following the approach suggested

in [55], which may be simplified due to the simple structure of the function considered. In the following, we focus on how to estimate a single component

$$\mathbb{E}[X_i] = \sum_{X_1=0}^{2^{n/D}-1} \cdots \sum_{X_D=0}^{2^{n/D}-1} p(X_1, \dots, X_D) X_i,$$

with $i \in \{1, \dots, D\}$, of the expected value. This needs to be repeated D times to estimate the full D -dimensional result. Without loss of generality we now assume $i = 1$ and define

$$S_f = S_x \otimes \bigotimes_{i=2}^D H^{\otimes n/D}.$$

It is easy to see that this corresponds to the objective function

$$f(X_1, \dots, X_D) = \frac{X_1}{C\sqrt{2^{n-n/D}}},$$

and we can apply the above approach to estimate $\mathbb{E}[X_1]/(C\sqrt{2^{n-n/D}})$. To achieve a target accuracy $\varepsilon > 0$ for $\mathbb{E}[X_1]$ using QPE requires an estimation accuracy of $\varepsilon/(C\sqrt{2^{n-n/D}})$ which results in a $\theta(2^{n/D}\sqrt{2^n})$ -times longer circuits (ignoring α).

To summarize, the proposed quantum algorithm can achieve an exponential speed-up for the (approximate) state evolution, but only up to a quadratic speed-up for the extraction of solution properties, which results in an overall complexity of $\mathcal{O}(T \text{poly}(n) + \sqrt{2^n})$ since the readout can be done using the final parameters without rerunning the evolution. Thus, the proposed algorithm might be particularly suitable for problems with a large time horizon. However, it could also accumulate a higher approximation error for those, and understanding where these two effects balance is subject to further research.

3.4 Enforcing ℓ_1 normalization

Many PDEs describe the evolution of probability distributions, i.e., the components of a discretized solution should be in $[0, 1]$ and sum up to one. Although we allow the scaling factor α_t to overcome the ℓ_2 -normalization of the quantum state $|\psi_t\rangle$, the variational principle in general does neither preserve the ℓ_1 -norm for $\alpha|\psi_t\rangle$ nor that the elements should be within $[0, 1]$. Since it minimizes a geometric distance, the closest representable state might not be ℓ_1 -normalized. In the following we describe how the procedure could be adjusted to (approximately) enforce ℓ_1 -normalization while sacrificing a bit of accuracy following the approach introduced in [56].

As shown in Section 3.3, we can estimate properties of the form $|\sum_{i=0}^{2^n-1} p_i f(x_i)|$ for an ℓ_2 -normalized function f and amplitudes p_i of the current state $|\psi_t\rangle$ up to an accuracy $\varepsilon > 0$. If we now set $f(x_i) = \sqrt{1/2^n}$ for all i , then we can easily construct $S_f = H^{\otimes n}$, i.e., just a single layer of Hadamard gates, which will result in an estimation of $|\sum_{i=0}^{2^n-1} p_i|/\sqrt{2^n}$.

If we estimate this to an accuracy $\varepsilon/\sqrt{2^n}$ using QPE, this can be used as an approximation to the ℓ_1 -norm of the state and allows to adjust the α_t such that the scaled quantum state would be (approximately) ℓ_1 -normalized. Note that the accuracy requirements on the estimation of the ℓ_1 -norm likely limit the potential advantage to a quadratic speed-up and increase the overall runtime to $\mathcal{O}(T\sqrt{2^n})$. We will illustrate the result of this approximation numerically in Section 4.3. Appendix D contains additional comparisons with and without the ℓ_1 -enforcement procedure described in this section.

3.5 Summarizing the algorithm

The algorithm we have described is now summarized in Algorithm 1. For clarity, we use one spatial dimension as well as the forward Euler ODE solver. However, in practice, more sophisticated solvers such as for instance Runge-Kutta methods are usually more suitable.

Algorithm 1: Variational quantum algorithm for the Feynman-Kac formula

Input: $n_q \in \mathbb{N}$, $D = 1$, $N_T > 0$, $\delta\tau > 0$, $N_T = \tau_{\text{total}}/\delta\tau$, $\delta x > 0$ and initial condition ψ . SDE of the form $dX_t = \mathbf{b}(X_t, t)dt + \sqrt{2\mathbf{a}(X_t, t)}dW_t$.

Output: Normalized state kets for solution of

$$\frac{\partial u}{\partial t} + \mathbf{a}(x, t)\frac{\partial^2 u}{\partial x^2} + \mathbf{b}(x, t)\frac{\partial u}{\partial x} - \mathbf{c}(x, t)u = 0. \quad (36)$$

- 1 Apply appropriate Feynman-Kac formulae to switch SDE to PDE where

$$u(x, t) = \mathbb{E}[e^{-\int_0^t \mathbf{c}(X_\tau, \tau)d\tau} \psi(X_T) \mid X_t = x] \text{ for } t > 0 \text{ and } u(x, 0) = \psi(x).$$

- 2 Determine the appropriate normalization (ℓ_1, ℓ_2, \dots) of (36).
 - 3 Extract generator/Hamiltonian \mathcal{G} for (36).
 - 4 Discretize \mathcal{G} with periodic boundary conditions.
 - 5 Decompose $\mathcal{G} = \sum_{h=1}^H \lambda_h U_h$, where $\lambda_h \in \mathbb{C}$ and U_h are implementable unitary operators composed of tensor products of Pauli operators.
 - 6 Choose ansatz circuit with $N + 1$ parameterized gates.
 - 7 Define initial condition $|\tilde{v}(\boldsymbol{\theta}(0))\rangle$.
 - 8 Determine initial parameters $\boldsymbol{\theta}_0 = (\theta_1(0), \dots, \theta_N(0), \theta_{N+1}(0))$ where $\theta_{N+1}(\tau) = \alpha(\tau)$, see (42).
 - 9 Initialize $\tau = 0$.
 - 10 **while** $\tau \leq \tau_{\text{total}}$ **do**
 - 11 Compute ansatz-dependent matrix elements
 $M_{k,j} = \text{Re}(\alpha^2(\tau) \frac{\partial \langle v(\boldsymbol{\theta}(\tau)) |}{\partial \theta_k} \frac{\partial |v(\boldsymbol{\theta}(\tau))\rangle}{\partial \theta_j})$ for $k, j \neq 0$ and
 $M_{0,j} = M_{j,0} = \alpha(\tau) \text{Re}(\langle v(\boldsymbol{\theta}(\tau)) | \frac{\partial |v(\boldsymbol{\theta}(\tau))\rangle}{\partial \theta_j})$ for $j > 0$ and $M_{0,0} = 1$.
 - 12 **if** \mathbf{M}^{-1} is singular **then**
 \quad Use least squares or Moore-Penrose inversion.
 - 13 **if** \mathcal{G} is time-dependent **then**
 \quad Update generator $\mathcal{G}(\tau)$ and its associated Pauli decomposition
 $\quad \sum_{h=1}^H \lambda_h(\tau) U_h(\tau)$.
 - 14 Compute generator-dependent vector elements
 $V_k = \alpha(\tau) \text{Re}(\frac{\partial \langle v(\boldsymbol{\theta}(\tau)) |}{\partial \theta_k} \mathcal{G} |v(\boldsymbol{\theta}(\tau))\rangle)$ for $k > 0$ and
 $V_0 = \text{Re}(\langle v(\boldsymbol{\theta}(\tau)) | \mathcal{G} |v(\boldsymbol{\theta}(\tau))\rangle)$.
 - 15 Update values of $\boldsymbol{\theta}$ with $\boldsymbol{\theta}(\tau + \delta\tau) \approx \boldsymbol{\theta}(\tau) + \mathbf{M}^{-1}(\tau) \cdot \mathbf{V} \delta\tau$.
 - 16 Update $\tau \rightarrow \tau + \delta\tau$.
-

4 Numerical experiments

We now provide numerical evidence of the algorithm we have proposed. To do so, we first need to establish the exact set of SDEs that we wish to study, in our case it will be a simple

set of two correlated Brownian motions. From there we will write down the PDE for the expectation of these Brownian motions. This PDE is an anisotropic heat equation that mixes the space variables x and y due to the correlation between the Brownian motions. The full solution to this PDE is provided in Appendix F by means of techniques from Mellin and Laplace transforms. The ansatz circuit will be explained and we will also illustrate the behavior of the ℓ_1 and ℓ_2 norms during the evolution. The VarQITE algorithm outputs the basis kets in one dimension, and since we shall use six and eight qubits, we will obtain 2^6 and 2^8 basis kets. Although the problem is two dimensional, it is instructive to take amplitude vector in the computational basis and the comparison against the classical methods of forward Euler as well as Monte Carlo with periodic boundary conditions.

4.1 Two-dimensional anisotropic heat equation

We shall now specialize the number of dimensions, the number of Brownian motions and the coefficients in the SDE to illustrate the Feynman-Kac formula on a quantum computer. Taking $D = N = 2$ in (15) yields

$$\begin{cases} dX_t^1 = \mu_1(t, X_t^1, X_t^2)dt + \sigma_{11}(t, X_t^1, X_t^2)dW_t^1 + \sigma_{12}(t, X_t^1, X_t^2)dW_t^2, \\ dX_t^2 = \mu_2(t, X_t^1, X_t^2)dt + \sigma_{21}(t, X_t^1, X_t^2)dW_t^1 + \sigma_{22}(t, X_t^1, X_t^2)dW_t^2. \end{cases} \quad (37)$$

Next, we specialize to $\mu_1 = \mu_2 = 0$ as well as

$$\sigma_{11} = \sqrt{\tilde{\sigma}_1}, \quad \sigma_{12} = 0, \quad \sigma_{21} = \sqrt{\tilde{\sigma}_2}\rho, \quad \sigma_{22} = \sqrt{\tilde{\sigma}_2}\sqrt{1 - \rho^2},$$

where ρ is a real in the interval $[-1, 1]$ which now is interpreted as the correlation between the two Brownian motions W_t^1 and W_t^2 . The resulting matrix becomes

$$\Sigma = \begin{pmatrix} \sqrt{\tilde{\sigma}_1} & 0 \\ \sqrt{\tilde{\sigma}_2}\rho & \sqrt{\tilde{\sigma}_2}\sqrt{1 - \rho^2} \end{pmatrix}.$$

The infinitesimal generator $\hat{\mathcal{G}}$ reduces to

$$\hat{\mathcal{G}} = \frac{\tilde{\sigma}_1}{2} \frac{\partial^2}{\partial x_1^2} + \rho\sqrt{\tilde{\sigma}_1\tilde{\sigma}_2} \frac{\partial^2}{\partial x_1 \partial x_2} + \frac{\tilde{\sigma}_2}{2} \frac{\partial^2}{\partial x_2^2} - r(x_1, x_2, t).$$

in this particular case. We shall also relabel $x_1 = x$ and $x_2 = y$. Moreover for simplicity our discount function will be $r(x, y, t) = 0$ for all t , and we shall choose σ_1 and σ_2 to be 1.

In summary, we are working with the differential operator and conditional expectation

$$\mathcal{G} = \rho \frac{\partial^2}{\partial x \partial y} + \frac{1}{2} \frac{\partial^2}{\partial x^2} + \frac{1}{2} \frac{\partial^2}{\partial y^2} = \mathcal{G}^\dagger \quad \text{and} \quad u(x, y, t) = \mathbb{E}[\psi(X_t, Y_t) \mid X_t = x, Y_t = y]. \quad (38)$$

The associated Feynman-Kac PDE for the expectation of the system of SDEs $dX_t = dW_t^1$ and $dY_t = dW_t^2$ is

$$\frac{\partial u}{\partial t} = \mathcal{G}u \quad t > 0, \quad (39)$$

with initial value condition ψ given by

$$u(x, y, 0) = \delta(x - x_0)\delta(y - y_0), \quad (40)$$

where δ is the Dirac delta function. The operator \mathcal{G} is an anisotropic Hamiltonian that includes the Laplacian operator $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$, as well as the term mixing both first order derivatives $\frac{\partial^2}{\partial x \partial y}$. Equation (39) admits an exact heat kernel solution, which we derive – in a more general setting involving arbitrary constant coefficients – in Appendix F,

$$u(x, y, t) = \frac{1}{2\pi t\sqrt{1-\rho^2}} \exp \left[-\frac{1}{t(1-\rho^2)} \left(\rho(x_0-x)(y-y_0) + \frac{1}{2}(x-x_0)^2 + \frac{1}{2}(y-y_0)^2 \right) \right]. \quad (41)$$

Incidentally, while the solution to (39) without the mixed term $\frac{\partial^2}{\partial x \partial y}$ and with initial condition (40) is well-known, the general solution (41) was not immediately available in the literature [57, 5, 58], see Mehler's formula. The results of a large number of Monte Carlo simulations and the plot of exact solution are shown in Figure 3.

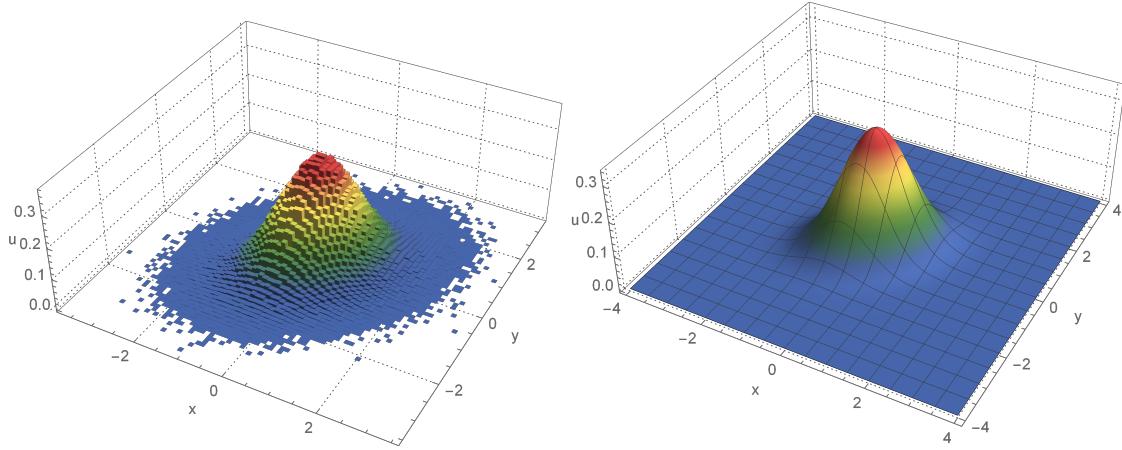


Figure 3: Left: probability distribution of 1 million paths governed by the system of SDEs from (37). Right: Corresponding analytical solution to (39) given by (41) with $\rho = \frac{1}{3}$ at $t = \frac{1}{10}$.

4.2 Choice of ansatz, expressivity, barren plateaus and initialization of parameters

The choice of ansatz will be what is called `RealAmplitudes` from `Qiskit` [59] and it can be found in Figure 4.

It is a circuit that consists of six qubits with several layers of single qubit rotations as well as CNOT gates with circular entanglement. The circuit always returns real-valued amplitudes. Other ansatze are indeed possible for example using R_x and R_z gates. However, this comes at the cost of taking the real part of the resulting ket at the end of the evolution, see e.g. [27]. Moreover, the non-parametric controlled gates could be replaced by parameter gates (such as controlled rotations) for more generality, see for instance the ansatz proposed in [11].

In order to get reliable simulation results, we need to ensure that the model ansatz is able to represent the problem in state space. However, we also need to consider that an ansatz which is sufficiently expressive, such that it corresponds to an (approximate) t-design [60, 61], is prone to result in barren plateaus [62]. More explicitly, it is known that for an ansatz that corresponds to an (approximate) t-design the gradient operator \mathbf{V} becomes exponentially small in the system size which in turn may cause problems in the model simulation. It is thus vital to trade off the expressivity of a chosen ansatz with potential propagation issues. One particularly promising approach relies on the use of additional information to design problem-specific ansätze.

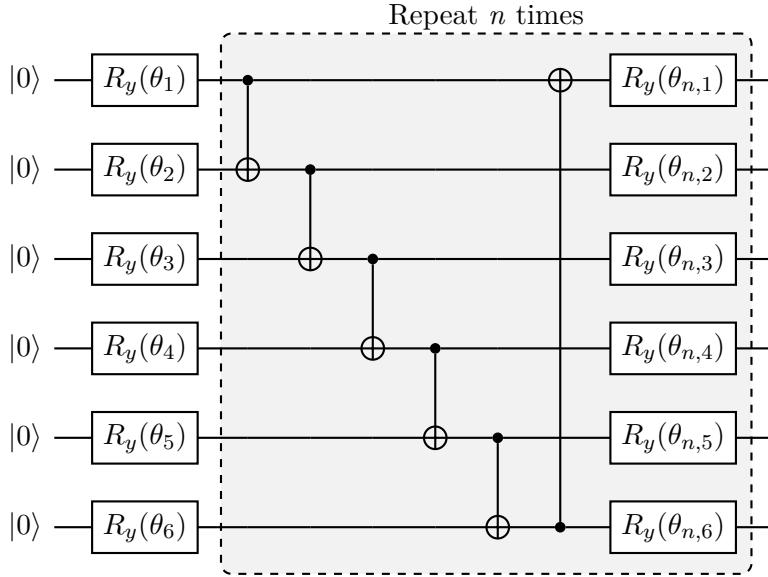


Figure 4: Parametric quantum circuit used in the numerical experiments in this work. It is composed of a layer of R_y rotation gates on every qubit, a block of nonparametric CNOT gates connecting successive qubits, and another layer of R_y rotation gates on every qubit. This parametric quantum circuit is also referred to as **RealAmplitudes** ansatz due to the real-valued nature of the amplitudes in the resulting state vector.

The choice of an appropriate ansatz for a given problem is still an active area of research. In our case, we are solving a PDE, (39), that admits only real solutions, which are in fact probability distributions. Therefore, it is convenient to keep the ansatz from producing complex amplitudes effectively avoiding dealing with the imaginary part of these amplitudes. Consequently, for the problem at hand it is reasonable to use the **RealAmplitudes** ansatz put forward in this section.

The parameters $\boldsymbol{\theta}_0 = (\theta_1, \dots, \theta_{12}, \theta_{13})$, where $\theta_{13} = \alpha(t)$ are initialized classically using

$$\boldsymbol{\theta}_0 = \operatorname{argmin}_{\boldsymbol{\theta} \in \mathbb{R}^n} \{ || |\psi(\boldsymbol{\theta})\rangle - |\psi(0)\rangle || \}. \quad (42)$$

In [11, §4], the authors make the point that care is needed when solving (42) due to the fact that the functions involved might not be convex and certain optimization techniques could be trapped in local minima. Moreover, the design of the ansatz should also go hand in hand with the optimization (42). Note that in general this scales poorly with the number of qubits. As with many quantum algorithms, one needs to be careful to ensure that the general state preparation techniques do not add costly overhead. Note that this initialisation step might also be performed with a quantum computer using SWAP test when the dimension of the problem starts to be too large for classical computers.

4.3 VarQITE vs forward Euler vs Monte Carlo in computational basis

Although we are dealing with the three-dimensional problem in (39), it is instructive to plot the amplitudes in the computational basis. The parameters are set to $dx = dy = 1$ as well as $dt = 0.001$. The number of evolutions is $N_T = 1000$. We first use six qubits in the ansatz with twelve parameterized gates, i.e. $n = 1$, and $\alpha(t) = \theta_{13}(t)$ as the norm encoder variable. With a circuit of six qubits and $n = 3$ layers, we have $\alpha(t) = \theta_{25}(t)$ as

norm encoder. Then we use eight qubits with $n = 5$ layers and $\alpha(t) = \theta_{41}(t)$ as the norm encoder variable.

The ℓ_1 norm enforcement of the evolution discussed in 3.4 can be seen in these plots of 1000 evolutions on the bottom left of Figures 6, 8 and 10. Earlier in the evolution period, the first kets came out with a flipped sign due to a phase. Moreover, periodic boundary conditions and the nature of the forward Euler method also yielded a norm slightly larger than one for the classical methods. Small deviations in the ℓ_1 -norm from VarQITE are coming from the approximate nature of the ℓ_1 -norm enforcement. Another useful metric to keep in mind is the difference between the ℓ_2 norms of forward Euler and VarQITE as well as Monte Carlo and VarQITE as shown on the bottom right of Figures 6, 8 and 10.

We note the following analysis of the experiments:

- (1) One can see in Figure 5 that VarQITE was significantly off around the computational basis states 32 to 38 with six qubits and only $n = 1$ layer. This situation is due to the choice of ansatz and its associated expressivity.
- (2) To see the effect of increasing the expressivity of the ansatz, we added more layers by increasing n from 1 to 3 while keeping the number of qubits constant at six. The results are plotted on Figure 7. Lastly, we then chose an eight qubit ansatz and added two additional layers of CNOTs and R_y rotations, i.e. $n = 5$. The results are in Figure 9.
- (3) In both instances (2) and (3), the match of the plots of VarQITE and forward Euler no longer show the discrepancy in the computational bits 32 to 38 for six qubits (respectively, 170 to 180 for eight qubits) while still maintaining a tight ℓ_1 normalization.
- (4) To perform the simulations with additional layers and additional qubits, numerical differentiation needed to be performed which added some instability. We expect these results to match more tightly with symbolic or circuit-based differentiation and this will be studied in future research.
- (5) We do not include finite-sampling noise or hardware noise. All numerical simulations have been tested and produced using `Python` as well as the `NumPy` library. The effect of noise on the results is an interesting topic and will be left for future research.
- (6) Although this methodology works with a high degree of generality, for high dimensional cases, the expressivity of the ansatz employed in VarQITE will limit the accuracy of the results.

5 Conclusion

We have proposed a quantum algorithm to transition from stochastic differential equations involving Wiener processes X_1, X_2, \dots, X_D to partial differential equations $\frac{\partial u}{\partial t} = \mathcal{G}u$ where u is the expectation of the Wiener processes. The resulting PDEs have been simulated through variational quantum imaginary time evolution (VarQITE). We have seen that well-known PDEs such as Schrödinger, heat and Fokker-Planck equations can be recast through this type of methodology and thereby can be unified in the single framework of the Feynman-Kac formula. Moreover, we have maintained a high degree of generality by allowing arbitrarily many Brownian motions per stochastic process and high dimensionality. However when dealing with dimensional cases, the expressivity of the ansatz will limit the

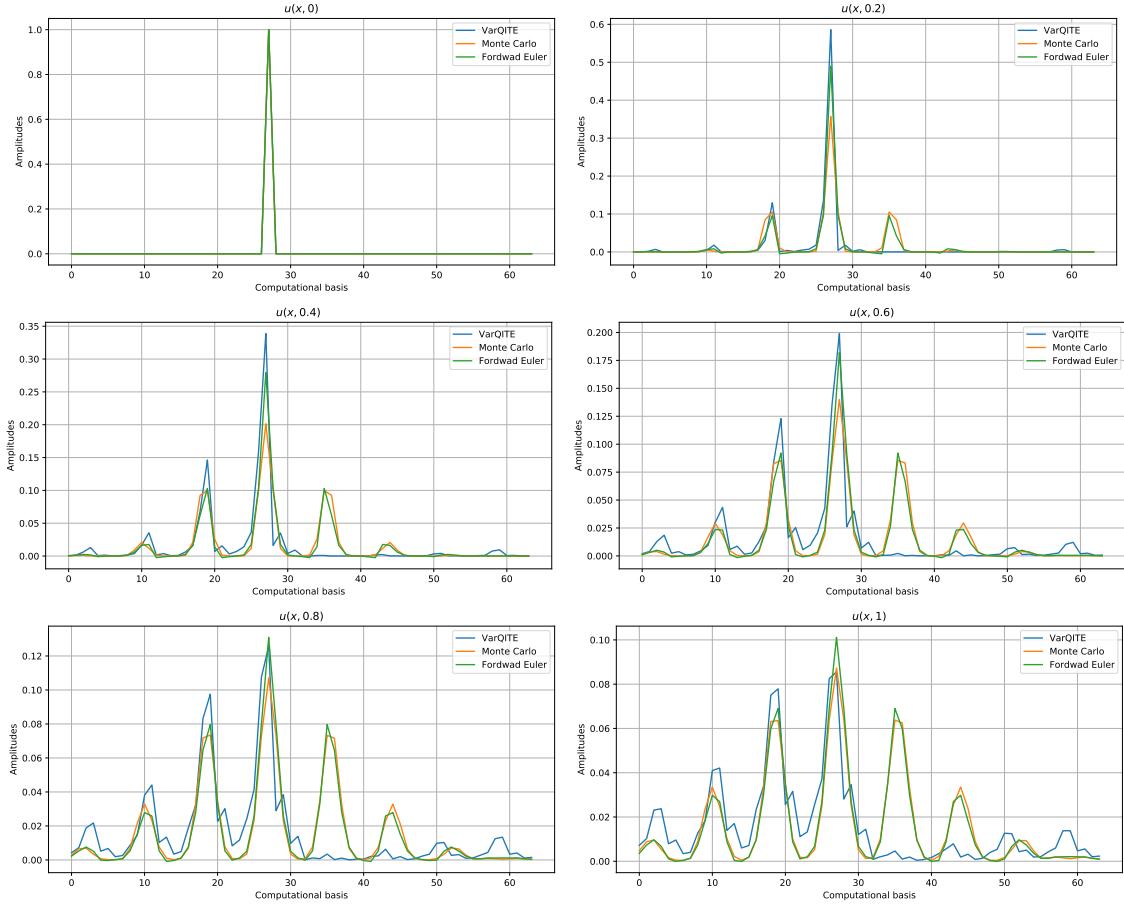


Figure 5: Plots of $u_{\text{VarQITE}}(x, t)$, $u_{\text{MC}}(x, t)$ and $u_{\text{FE}}(x, t)$ for $t \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$ in computational basis states with an ansatz consisting of six qubits and $n = 1$ layer. The horizontal axis represents $2^6 = 64$ computational basis and vertical axis represents $u(x, t)$ with $dt = 0.001$ and 1000 evolutions.

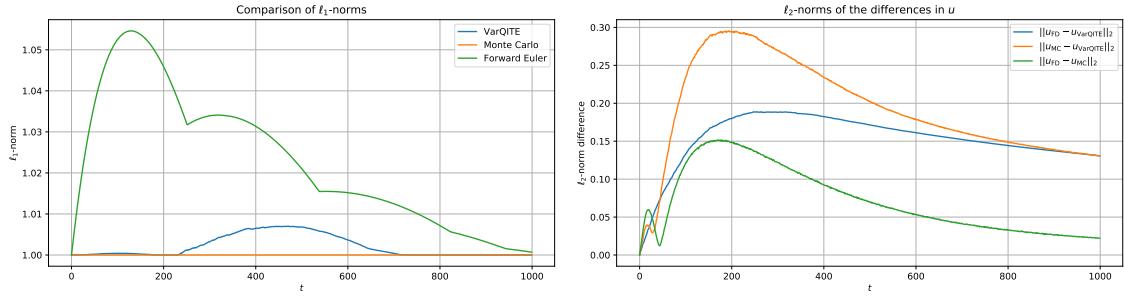


Figure 6: Left: ℓ_1 norms for the first 1000 evolutions are plotted using three different methods: VarQITE, Monte Carlo, and forward Euler method. Note that the scale on the y -axis has been narrowed down to demonstrate the approximate preservation of the ℓ_1 norms for all three methods. The x -axis represents time-steps and y axis represents ℓ_1 -norm. Right: This error plot shows the ℓ_2 norms of the differences between the solutions u obtained via VarQITE and forward Euler (blue) and between VarQITE and Monte Carlo method (yellow). The x -axis represents time and y axis represents ℓ_2 -norm of the differences in u .

accuracy of the results. Therefore, as the dimension of the problem increases one has to improve or enhance the ansatz accordingly. Some of the resulting PDEs do not preserve ℓ_2 norm, such as the heat equation which preserves probability distributions. This represents a non-trivial obstacle and we have utilized novel ideas to maintain the solution approximately

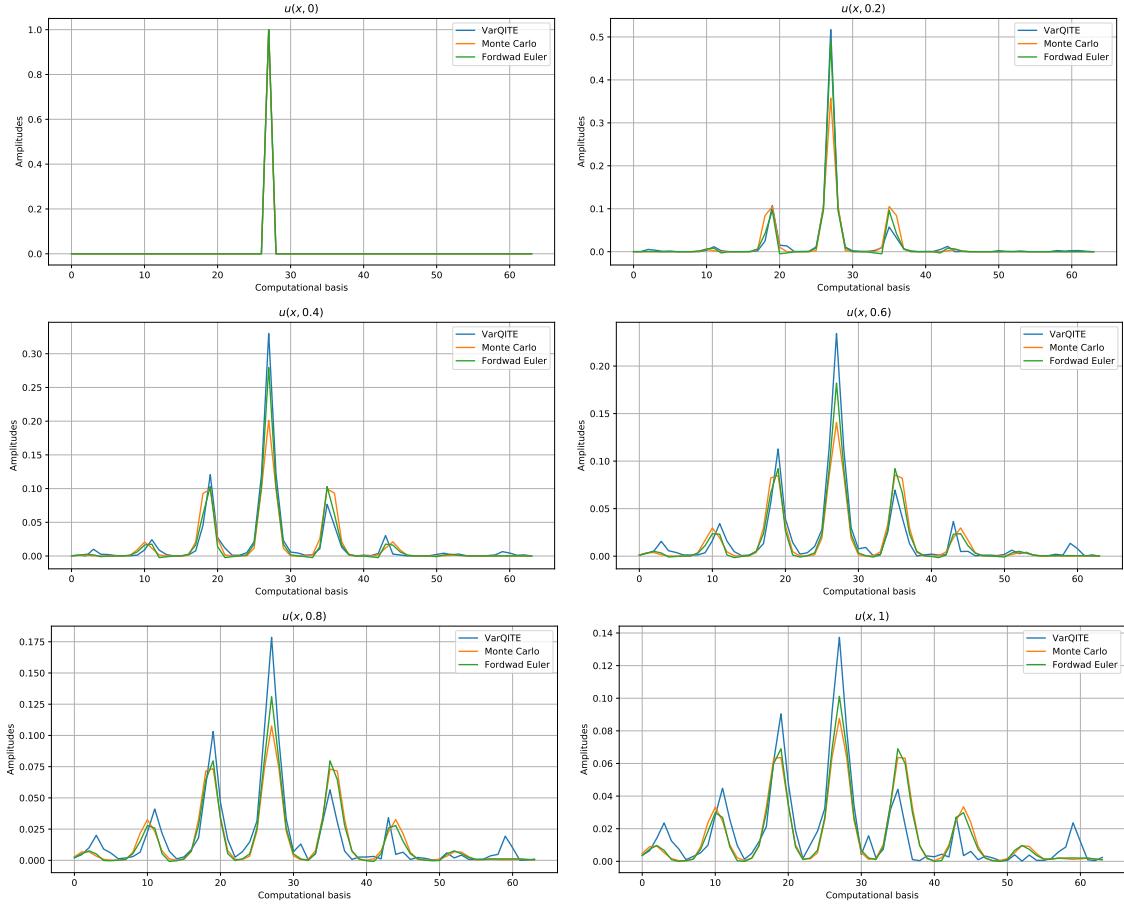


Figure 7: Plots of $u_{\text{VarQITE}}(x, t)$, $u_{\text{MC}}(x, t)$ and $u_{\text{FE}}(x, t)$ for $t \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$ in computational basis states with an ansatz consisting of six qubits and $n = 3$ layers. The horizontal axis represents $2^6 = 64$ computational basis and vertical axis represents $u(x, t)$ with $dt = 0.001$ and 1000 evolutions.

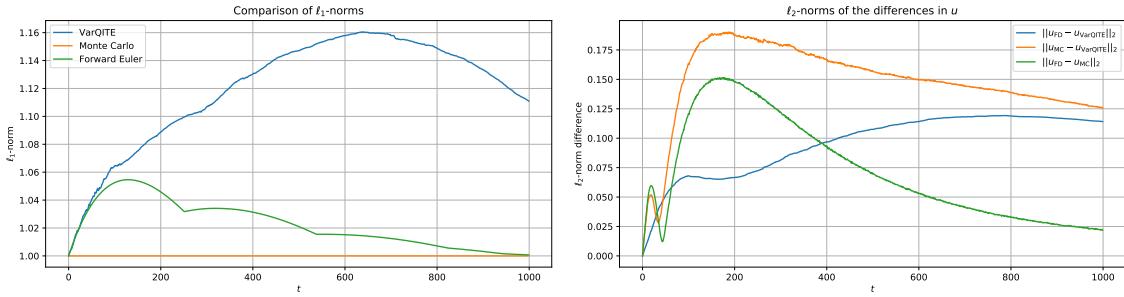


Figure 8: Left: ℓ_1 norms for the first 1000 evolutions are plotted using three different methods: VarQITE, Monte Carlo, and forward Euler method. Note that the scale on the y -axis has been narrowed down to demonstrate the approximate preservation of the ℓ_1 norms for all three methods. The x -axis represents time-steps and y axis represents ℓ_1 -norm. Right: This error plot shows the ℓ_2 norms of the differences between the solutions u obtained via VarQITE and forward Euler (blue) and between VarQITE and Monte Carlo method (yellow). The x -axis represents time and y axis represents ℓ_2 -norm of the differences in u .

normalized throughout the quantum evolution by introducing a proxy to the ℓ_1 norm.

The algorithm has shown substantial agreement in the solution of the PDE through VarQITE, classical Monte Carlo and forward Euler in a sufficiently complex and non-trivial

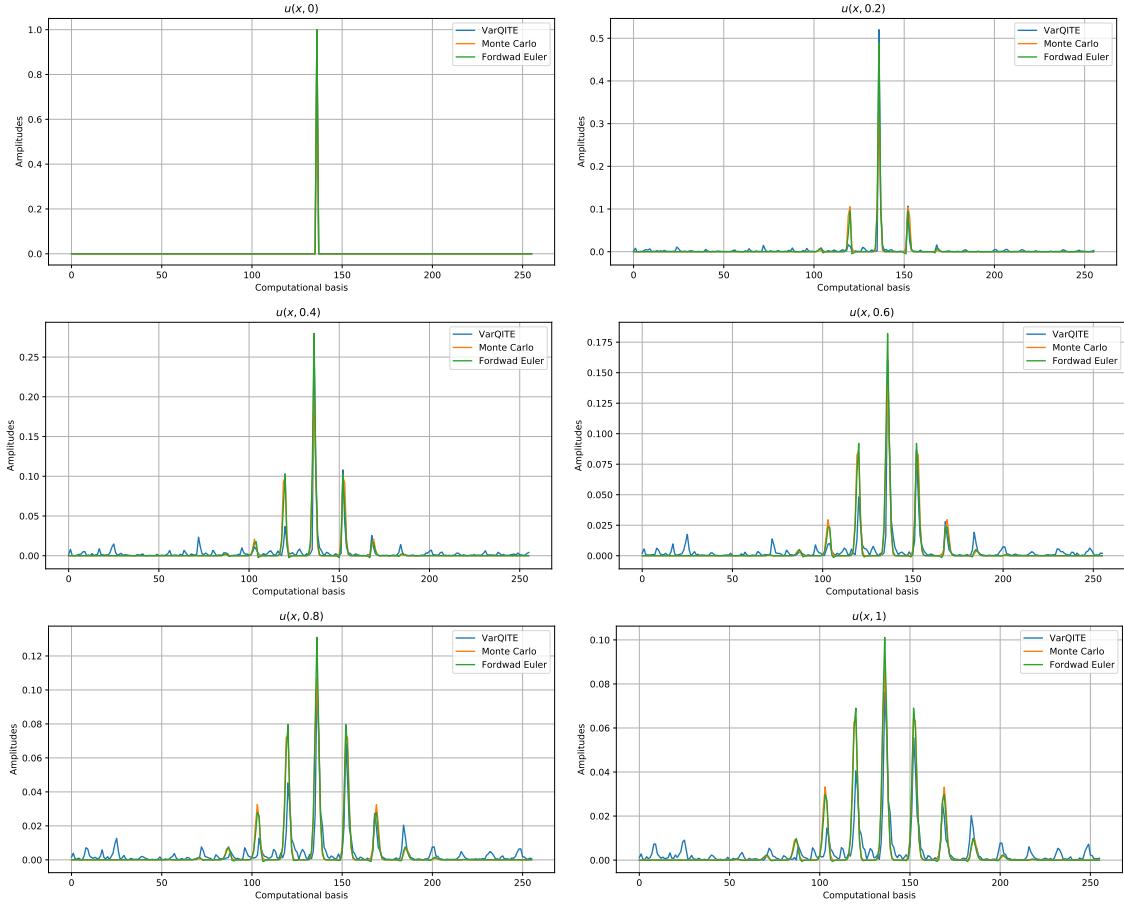


Figure 9: Plots of $u_{\text{VarQITE}}(x, t)$, $u_{\text{MC}}(x, t)$ and $u_{\text{FE}}(x, t)$ for $t \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$ in computational basis states with an ansatz consisting of eight qubits and $n = 5$ layers. The horizontal axis represents $2^8 = 256$ computational basis and vertical axis represents $u(x, t)$ with $dt = 0.001$ and 1000 evolutions.

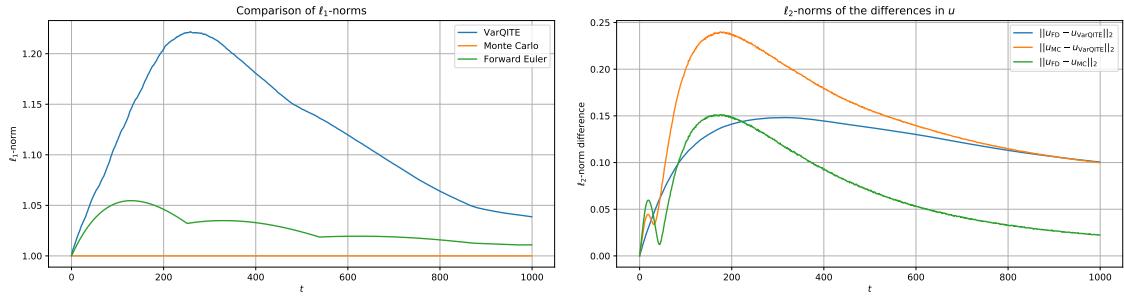


Figure 10: Left: ℓ_1 norms for the first 1000 evolutions are plotted using three different methods: VarQITE, Monte Carlo, and forward Euler method. Note that the scale on the y -axis has been narrowed down to demonstrate the approximate preservation of the ℓ_1 norms for all three methods. The x -axis represents time-steps and y axis represents ℓ_1 -norm. Right: This error plot shows the ℓ_2 norms of the differences between the solutions u obtained via VarQITE and forward Euler (blue) and between VarQITE and Monte Carlo method (yellow). The x -axis represents time and y axis represents ℓ_2 -norm of the differences in u .

problem involving an anisotropic Fokker-Planck equation in $(2 + 1)$ dimensions. Because obtaining the full solution to the PDE is an expensive problem, we need to instead address obtaining the multidimensional moments of the solution through quantum techniques.

In order to do so, we have decomposed the second order differential operator \mathcal{G} from the PDE into a feasible set of unitaries composed of few-qubit gates. For a suitable function f , the moments $\mathbb{E}[f(X_1, X_2, \dots, X_D)]$ can be computed by approximating f with its multidimensional Taylor series to a fixed order and this approximation can be improved by increasing the number of qubits to allow for finer resolution. These moments can be computed through QPE or Hadamard tests with potential advantages depending on the concrete situation.

The methodology provided here supplements the existing quantum-enhanced Monte Carlo algorithms and both techniques can coexist and complement each other. The scope of the topics we have addressed and their mapping to quantum computers allows us to open the possibilities for other interesting items for future work such as option pricing with stochastic volatility, see [36], as well as stochastic control problems such as American and decision-embedded options which require the Hamilton-Jacobi-Bellman equation, see [63]. These topics might have to be explored in the context of real, as opposed to imaginary, time evolution. Lastly, the heat equation is not the only candidate for this algorithm. Itô's lemma can be used [40, §4.1.1] to transform a conditional expectation arising from a vectorized SDE into the solution of the PDE. Another venue would be to consider the wave or Klein-Gordon equations (or in general, other type of hyperbolic differential equations). In this case, the time derivative is now of second order, unlike in the heat equation, which was of first order. For these situations, other types of probability distributions and Brownian motions need to be added, specially by considering jumps and Poisson processes, see [64].

6 Acknowledgments

The authors would like to thank Peter Tysowski and Vaibhaw Kumar for constructive discussions regarding these topics. Antoine Jacquier was kind enough to read the manuscript and provide some corrections and suggestions. Oleksandr Kyriienko also provided insightful remarks. CZ acknowledges support from the National Centre of Competence in Research *Quantum Science and Technology* (QSIT). The authors would like to acknowledge the referees for the very helpful comments that have greatly increased the accuracy, precision, and readability of the manuscript.

IBM, the IBM logo, and ibm.com are trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. The current list of IBM trademarks is available at <https://www.ibm.com/legal/copytrade>.

7 Code availability

The code that supports the findings of this study is available from the corresponding author NR upon reasonable request.

References

- [1] R. P. Feynman. The principle of least action in quantum mechanics. In *Feynman's Thesis—A New Approach To Quantum Theory*, pages 1–69. World Scientific, 2005. Available online: https://doi.org/10.1142/9789812567635_0001.

- [2] M. Kac. On Distributions of Certain Wiener Functionals. *Transactions of the American Mathematical Society*, 65(1):1–13, 1949. DOI: [10.1090/S0002-9947-1949-0027960-X](https://doi.org/10.1090/S0002-9947-1949-0027960-X).
- [3] F. Black and M. Scholes. The pricing of options and corporate liabilities. *Journal of Political Economy*, 81(3):637–654, 1973. Available online: <https://doi.org/10.1086/260062>.
- [4] J. Glimm and A. Jaffe. *Quantum Physics: A Functional Integral Point of View*. Springer-Verlag New York, 2012. DOI: [10.1007/978-1-4612-4728-9](https://doi.org/10.1007/978-1-4612-4728-9).
- [5] J. Lörinczi, F. Hiroshima, and V. Betz. *Feynman-Kac-Type Theorems and Gibbs Measures on Path Space: With Applications to Rigorous Quantum Field Theory*. De Gruyter, 2011. DOI: [10.1515/9783110203738](https://doi.org/10.1515/9783110203738).
- [6] A. Korzeniowski, J. Fry, D. Orr, and N. Fazleev. Feynman-Kac path-integral calculation of the ground-state energies of atoms. *Phys Rev Lett*, 10(69):893–896, 1992. DOI: [10.1103/PhysRevLett.69.893](https://doi.org/10.1103/PhysRevLett.69.893).
- [7] M. Caffarel and P. Claverie. Development of a pure diffusion quantum Monte Carlo method using a full generalized Feynman–Kac formula. I. Formalism. *The Journal of Chemical Physics*, 88(2):1088–1099, 1988. DOI: [10.1063/1.454227](https://doi.org/10.1063/1.454227).
- [8] P. A. Faria da Veiga, M. O’Carroll, and R. Schor. Existence of baryons, baryon spectrum and mass splitting in strong coupling lattice QCD. *Commun. Math. Phys.*, 245:383–405, 2004. DOI: [10.1007/s00220-003-1022-2](https://doi.org/10.1007/s00220-003-1022-2).
- [9] J. Gonzalez-Conde, Á. Rodríguez-Rozas, E. Solano, and M. Sanz. Pricing Financial Derivatives with Exponential Quantum Speedup. arxiv:2101.04023, 2021. DOI: [10.48550/arXiv.2101.04023](https://doi.org/10.48550/arXiv.2101.04023).
- [10] S. K. Radha. Quantum option pricing using Wick rotated imaginary time evolution. arXiv:2101.04280, 2021. DOI: [10.48550/arXiv.2101.04280](https://doi.org/10.48550/arXiv.2101.04280).
- [11] F. Fontanella, A. Jacquier, and M. Oumgari. A Quantum algorithm for linear PDEs arising in Finance. *SSRN Electronic Journal*, 2019. DOI: [10.2139/ssrn.3499134](https://doi.org/10.2139/ssrn.3499134).
- [12] K. Kubo, Y. O. Nakagawa, S. Endo, and S. Nagayama. Variational quantum simulations of stochastic differential equations. *Phys. Rev. A*, 103:052425, 2021. DOI: [10.1103/PhysRevA.103.052425](https://doi.org/10.1103/PhysRevA.103.052425).
- [13] G. Brassard, P. Hoyer, M. Mosca, and A. Tapp. Quantum Amplitude Amplification and Estimation. *Quantum Computation and Quantum Information, Samuel J. Lomonaco, Jr. (editor), AMS Contemporary Mathematics*, 305:53–74, 2002. DOI: [10.1090/conm/305](https://doi.org/10.1090/conm/305).
- [14] A. Montanaro. Quantum speedup of Monte Carlo methods. *Proceedings of the Royal Society A*, 471(2181), 2015. DOI: [10.1098/rspa.2015.0301](https://doi.org/10.1098/rspa.2015.0301).
- [15] Y. Suzuki, S. Uno, R. Raymond, T. Tanaka, T. Onodera, and N. Yamamoto. Amplitude Estimation without Phase Estimation. *Quantum Inf Process*, 19(75), 2020. DOI: [10.1007/s11128-019-2565-2](https://doi.org/10.1007/s11128-019-2565-2).
- [16] D. Grinko, J. Gacon, C. Zoufal, and S. Woerner. Iterative quantum amplitude estimation. *npj Quantum Information*, 7:52, 2021. DOI: [10.1038/s41534-021-00379-1](https://doi.org/10.1038/s41534-021-00379-1).
- [17] A. Carrera Vazquez and S. Woerner. Efficient state preparation for quantum amplitude estimation. *Phys. Rev. Applied*, 15:034027, 2021. DOI: [10.1103/PhysRevApplied.15.034027](https://doi.org/10.1103/PhysRevApplied.15.034027).

- [18] N. Stamatopoulos, D. J. Egger, Y. Sun, C. Zoufal, R. Iten, N. Shen, and S. Woerner. Option Pricing using Quantum Computers. *Quantum*, 4(291), 2020. DOI: [10.22331/q-2020-07-06-291](https://doi.org/10.22331/q-2020-07-06-291).
- [19] S. Woerner and D. J. Egger. Quantum Risk Analysis. *npj Quantum Information*, 5(15), 2019. DOI: [10.1038/s41534-019-0130-6](https://doi.org/10.1038/s41534-019-0130-6).
- [20] D. J. Egger, R. Garcia Gutierrez, J. Cahue Mestre, and S. Woerner. Credit risk analysis using quantum computers. *IEEE Transactions on Computers*, pages 1–1, 2020. DOI: [10.1109/TC.2020.3038063](https://doi.org/10.1109/TC.2020.3038063).
- [21] S. Chakrabarti, R. Krishnakumar, G. Mazzola, N. Stamatopoulos, S. Woerner, and W. J. Zeng. A Threshold for Quantum Advantage in Derivative Pricing. *Quantum*, 5:463, 2021. DOI: [10.22331/q-2021-06-01-463](https://doi.org/10.22331/q-2021-06-01-463).
- [22] D. An, N. Linden, J.-P. Liu, A. Montanaro, C. Shao, and J. Wang. Quantum-accelerated multilevel Monte Carlo methods for stochastic differential equations in mathematical finance. *Quantum*, 5:481, 2021. DOI: [10.22331/q-2021-06-24-481](https://doi.org/10.22331/q-2021-06-24-481).
- [23] P. Rebentrost, B. Gupt, and T. R. Bromley. Quantum computational finance: Monte Carlo pricing of financial derivatives. *Phys. Rev. A*, 98:022321, 2018. DOI: [10.1103/PhysRevA.98.022321](https://doi.org/10.1103/PhysRevA.98.022321).
- [24] B. Oksendal. *Stochastic Differential Equations*. Springer, 2013. DOI: [10.1007/978-3-642-14394-6](https://doi.org/10.1007/978-3-642-14394-6).
- [25] ocw.mit.edu. Lecture 21: Stochastic Differential Equations. Available online: https://ocw.mit.edu/courses/mathematics/18-s096-topics-in-mathematics-with-applications-in-finance-fall-2013/lecture-notes/MIT18_S096F13_lecnote21.pdf.
- [26] S. Endo, J. Sun, Y. Li, S. C. Benjamin, and X. Yuan. Variational quantum simulation of general processes. *Phys. Rev. Lett.*, 125:010501, 2020. DOI: [10.1103/PhysRevLett.125.010501](https://doi.org/10.1103/PhysRevLett.125.010501).
- [27] X. Yuan, S. Endo, Q. Zhao, Y. Li, and S. C. Benjamin. Theory of variational quantum simulation. *Quantum*, 3(191), 2019. DOI: [10.22331/q-2019-10-07-191](https://doi.org/10.22331/q-2019-10-07-191).
- [28] C. Zoufal, D. Sutter, and S. Woerner. Error Bounds for Variational Quantum Time Evolution. arXiv:2108.00022, 2021. DOI: [10.48550/arXiv.2108.00022](https://doi.org/10.48550/arXiv.2108.00022).
- [29] N. H. Bingham and R. Kiesel. *Risk-Neutral Valuation: Pricing and Hedging Financial Derivatives*. Springer-Verlag Berlin Heidelberg, 2004. DOI: [10.1007/978-1-4471-3856-3](https://doi.org/10.1007/978-1-4471-3856-3).
- [30] S. Ramos-Calderer, A. Pérez-Salinas, D. García-Martín, C. Bravo-Prieto, J. Cortada, J. Planagumà, and J. I. Latorre. Quantum unary approach to option pricing. *Phys. Rev. A*, 103:032414, 2021. DOI: [10.1103/PhysRevA.103.032414](https://doi.org/10.1103/PhysRevA.103.032414).
- [31] I. Karatzas and S. Shreve. *Brownian Motion and Stochastic Calculus*. Graduate Texts in Mathematics, Springer, 1984. DOI: [10.1007/978-1-4612-0949-2](https://doi.org/10.1007/978-1-4612-0949-2).
- [32] M. Musiela and M. Rutkowski. *Martingale Methods in Financial Modelling*. Springer-Verlag Berlin Heidelberg, 2005. DOI: [10.1007/b137866](https://doi.org/10.1007/b137866).
- [33] M. Baxter and A. Rennie. *Financial Calculus: An introduction to derivative pricing*. Cambridge University Press, 1996. DOI: [10.1017/CBO9780511806636](https://doi.org/10.1017/CBO9780511806636).
- [34] E. Chin, D. Nel, and S. Olafsson. *Problems and Solutions in Mathematical Finance: Stochastic Calculus*. Wiley Finance Series, 2014. DOI: [10.1002/9781118845141](https://doi.org/10.1002/9781118845141).

- [35] A. Papanicolaou. Introduction to Stochastic Differential Equations (SDEs) for Finance. arXiv:1504.05309, 2015. DOI: 10.48550/arXiv.1504.05309.
- [36] P. Wilmott. *Paul Wilmott On Quantitative Finance*. Wiley, 2006.
- [37] T. Constantinescu. *Schur parameters, Dilation and Factorization Problems*. Birkhauser Verlag 82, 1996. DOI: 10.1007/978-3-0348-9108-0.
- [38] F. Bagarello. Susy for non-Hermitian Hamiltonians, with a view to coherent states. *Math. Phys. Anal. Geom.*, 23(3):28, 2020. DOI: 10.1007/s11040-020-09353-3.
- [39] S. Dogra, A. A. Melnikov, and G. S. Paraoanu. Quantum simulation of parity-time symmetry breaking with a superconducting quantum processor. *Communications Physics*, 4(26), 2021. DOI: 10.1038/s42005-021-00534-2.
- [40] J. N. D. of Mathematics at Stanford University) math.stanford.edu/~ryzhik/STANFORD/STANF227-10/notes227-09.pdf. Partial differential equations and diffusion processes. Available online: <http://math.stanford.edu/~ryzhik/STANFORD/STANF227-10/notes227-09.pdf>.
- [41] S. L. D. of Statistics at University of Chicago) stat.uchicago.edu/~lalley/Courses/391/Lecture12.pdf. Stochastic Differential Equations, Diffusion Processes and the Feynman-Kac formula. Available online: <http://www.stat.uchicago.edu/~lalley/Courses/391/Lecture12.pdf>.
- [42] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge University Press, 2007. DOI: 10.1145/1874391.187410.
- [43] Y. Li and S. C. Benjamin. Efficient variational quantum simulator incorporating active error minimization. *Phys. Rev. X*, 7:021050, 2017. DOI: 10.1103/PhysRevX.7.021050.
- [44] S. McArdle, T. Jones, S. Endo, Y. Li, S. Benjamin, and X. Yuan. Variational ansatz-based quantum simulation of imaginary time evolution. *npj Quantum Information*, 5(75), 2019. DOI: 10.1038/s41534-019-0187-2.
- [45] A. D. McLachlan. A variational solution of the time-dependent Schrödinger equation. *Molecular Physics*, 8(1), 1964. DOI: 10.1080/00268976400100041.
- [46] G. M. D'Ariano, M. G. Paris, and M. F. Sacchi. Quantum tomography. *Advances in Imaging and Electron Physics*, 128:206–309, 2003. DOI: 10.1016/S1076-5670(03)80065-4.
- [47] L. C. Evans. *Partial Differential Equations*. American Mathematical Society, 1998. DOI: 10.1112/blms/20.4.375.
- [48] J. Solomon, L. Guibas, and A. Butscher. Dirichlet energy for analysis and synthesis of soft maps. In *Computer Graphics Forum*, volume 32, pages 197–206. Wiley Online Library, 2013. DOI: 10.1111/cgf.12186.
- [49] K. Zhou, X. Huang, D. Zha, R. Chen, L. Li, S.-H. Choi, and X. Hu. Dirichlet energy constrained learning for deep graph neural networks. *Advances in Neural Information Processing Systems*, 34, 2021. DOI: 10.48550/arXiv.2107.02392.
- [50] O. Stein, M. Wardetzky, A. Jacobson, and E. Grinspun. A simple discretization of the vector dirichlet energy. In *Computer Graphics Forum*, volume 39, pages 81–92. Wiley Online Library, 2020. DOI: 10.1111/cgf.14070.

- [51] O. Stein, A. Jacobson, M. Wardetzky, and E. Grinspun. A smoothness energy without boundary distortion for curved surfaces. *ACM Transactions on Graphics (TOG)*, 39(3):1–17, 2020. DOI: 10.1145/3377406.
- [52] G. Sussmann. Uncertainty Relation: From Inequality to Equality. *Zeitschrift für Naturforschung A*, 1997. DOI: 10.1515/zna-1997-1-214.
- [53] X. Li, G. Yang, C. M. Torres, D. Zheng, and K. L. Wang. A Class of Efficient Quantum Incrementer Gates for Quantum Circuit Synthesis. *International Journal of Modern Physics B*, 28(1), 2014. DOI: 10.1142/S0217979213501919.
- [54] M. A. Nielsen and I. L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 2012. DOI: 10.1017/CBO9780511976667.
- [55] L. Grover and T. Rudolph. Creating superpositions that correspond to efficiently integrable probability distributions. arXiv:0208112, 2002. DOI: 10.48550/arXiv.quant-ph/0208112.
- [56] A. Carrera Vazquez, R. Hiptmair, and S. Woerner. Enhancing the Quantum Linear Systems Algorithm using Richardson Extrapolation. arXiv:2009.04484, 2020. DOI: 10.1063/1.454227.
- [57] N. Berline, E. Getzler, and M. Vergne. *Heat Kernels and Dirac Operators*. Springer-Verlag Berlin Heidelberg, 2004.
- [58] P. DuChateau and D. W. Zachmann. *Schaum's outlines: partial differential equations*. McGraw-Hill, 1986. DOI: 10.1007/978-1-4612-0949-2.
- [59] qiskit.org/documentation/stubs/qiskit.circuit.library.RealAmplitudes.html. qiskit.circuit.library.realamplitudes, 2021. Available online: [qiskit.org](https://qiskit.org/documentation/stubs/qiskit.circuit.library.RealAmplitudes.html).
- [60] A. Hayashi, T. Hashimoto, and M. Horibe. Reexamination of optimal quantum state estimation of pure states. *Physical Review A*, 72(3), 2005. DOI: 10.1103/PhysRevA.72.032325.
- [61] A. Ambainis and J. Emerson. Quantum t-designs: t-wise Independence in the Quantum World. In *Twenty-Second Annual IEEE Conference on Computational Complexity (CCC'07)*, 2007. DOI: 10.1109/CCC.2007.26.
- [62] J. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush, and H. Neven. Barren plateaus in quantum neural network training landscapes. *Nature Communications*, 9, 2018. DOI: 10.1038/s41467-018-07090-4.
- [63] J. Vecer. A New PDE Approach for Pricing Arithmetic Average Asian Options. *Journal of Computational Finance*, 4(4), 2001. DOI: 10.21314/JCF.2001.064.
- [64] R. C. Dalang, C. Mueller, and R. Tribe. A Feynman-Kac Type Formula for the deterministic and stochastic wave equations and other PDEs. *Transactions of the American Mathematical Society*, 360(9):4681–4703, 2008. DOI: 10.1090/S0002-9947-08-04351-1.
- [65] R. Brummelhuis. Mathematical Methods - MSc Financial Engineering - Birbeck College, 2004. Available online: http://www.cato.tzo.com/brad_bbk/teaching/Methods/old_methods_notes_RB.pdf.
- [66] V. V. Shende, S. S. Bullock, and I. L. Markov. Synthesis of quantum-logic circuits. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 25(6), 2006. DOI: 10.1109/TCAD.2005.855930.

- [67] C. Zoufal, A. Lucchi, and S. Woerner. Quantum Generative Adversarial Networks for learning and loading random distributions. *npj Quantum Information*, 5(103), 2019. DOI: [10.1038/s41534-019-0223-2](https://doi.org/10.1038/s41534-019-0223-2).
- [68] F. Oberhettinger. *Tables of Mellin Transforms*. Springer-Verlag Berlin Heidelberg, 1974. DOI: [10.1007/978-3-642-65975-1](https://doi.org/10.1007/978-3-642-65975-1).
- [69] E. Knill, G. Ortiz, and R. D. Somma. Optimal quantum measurements of expectation values of observables. *Phys. Rev. A*, 75:012328, 2007. DOI: [10.1103/PhysRevA.75.012328](https://doi.org/10.1103/PhysRevA.75.012328).
- [70] D. Wang, O. Higgott, and S. Brierley. Accelerated variational quantum eigensolver. *Phys. Rev. Lett.*, 122:140504, 2019. DOI: [10.1103/PhysRevLett.122.140504](https://doi.org/10.1103/PhysRevLett.122.140504).

A Details on Feynman-Kac, Itô's lemma and the heat equation

Suppose that $u = u(x, t)$ with $x \in \mathbb{R}$ and $0 \leq t \leq T$ is a solution to the ‘backward’ heat equation:

$$\begin{cases} \frac{\partial u}{\partial t} + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} = 0, & \text{for } t < T \\ u(x, T) = \psi(x). \end{cases} \quad (43)$$

Now let $(W_t)_{t \geq 0}$ be Brownian motion and consider $u(W_t, t)$. Performing a small variation yields

$$u(W_t + dW_t, t + dt) = u(W_t, t) + \frac{\partial u}{\partial t} dt + \frac{\partial u}{\partial x} dW_t + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} (dW_t)^2 + O(dt^2, dW_t^3). \quad (44)$$

From Itô’s lemma (1) we have the rule of thumb

$$(dW_t)^k = \begin{cases} dW_t, & \text{if } k = 1, \\ dt, & \text{if } k = 2, \\ 0, & \text{if } k > 2. \end{cases}$$

Moreover, $(dt)^k = 0$ for $k \geq 2$. Thus, we will ignore the last terms in (44). Passing the term $u(W_t, t)$ to the left-hand side we see that (44) becomes

$$\begin{aligned} u(W_t + dW_t, t + dt) - u(W_t, t) &= du(W_t, t) \\ &= \frac{\partial u}{\partial t} dt + \frac{\partial u}{\partial x} dW_t + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} (dW_t)^2 \\ &= \left(\frac{\partial u}{\partial t} + \frac{1}{2} \frac{\partial^2 u}{\partial x^2} \right) dt + \frac{\partial u}{\partial x} dW_t. \end{aligned}$$

By (43) we see that the term involving dt is zero, and therefore we are left with $du(W_t, t) = \frac{\partial u}{\partial x} dW_t$. Integrate this from t to T so that

$$\int_t^T du(W_s, s) = u(W_T, T) - u(W_t, t) = \int_t^T \frac{\partial u}{\partial x}(W_s, s) dW_s,$$

which, after re-arrangement, leads to

$$u(W_t, t) = \psi(W_T) - \int_t^T \frac{\partial u}{\partial x}(W_s, s) dW_s, \quad (45)$$

where we used the final boundary condition of (43). The key part is that we must now take expectations, conditioned on $W_t = x$, i.e. $u(x, t) = \mathbb{E}[\psi(W_T) | W_t = x]$, so that picking up from (45) we get

$$u(x, t) = \mathbb{E}[\psi(W_T) | W_t = x] + \mathbb{E}\left[\int_t^T \frac{\partial u}{\partial x}(W_s, s) dW_s | W_t = x\right]. \quad (46)$$

Let us deal with the second term first. Using tools from analysis, such as a generalization of Fubini's theorem, we can swap the expectation and the integral and therefore

$$\mathbb{E}\left[\int_t^T \frac{\partial u}{\partial x}(W_s, s) dW_s | W_t = x\right] = \int_t^T \mathbb{E}\left[\frac{\partial u}{\partial x}(W_s, s) dW_s | W_t = x\right].$$

From the properties of Brownian motion we know that dW_s is independent of W_s or, for that matter, any other function of W_s , hence by independence we can separate the expectation of the product into the product of the expectations as

$$\mathbb{E}\left[\frac{\partial u}{\partial x}(W_s, s) dW_s | W_t = x\right] = \mathbb{E}\left[\frac{\partial u}{\partial x}(W_s, s) | W_t = x\right] \times \mathbb{E}[dW_s | W_t = x].$$

Invoking again the independence of Brownian increments we get $\mathbb{E}[dW_s | W_t = x] = \mathbb{E}[dW_s] = 0$, since the Brownian motion is associated with a normal distribution with mean zero. Thus the expectation of the integral in (46) is simply zero and we are left with $u(x, t) = \mathbb{E}[\psi(W_T) | W_t = x]$. The task at hand now is to compute this expectation and to this end, we use $W_T = W_t + (W_T - W_t)$ inside the expectation so that

$$\begin{aligned} \mathbb{E}[\psi(W_T) | W_t = x] &= \mathbb{E}[\psi(W_t + (W_T - W_t)) | W_t = x] \\ &= \mathbb{E}[\psi(x + (W_T - W_t)) | W_t = x] \\ &= \mathbb{E}[\psi(x + (W_T - W_t))]. \end{aligned}$$

This is because the expressions $W_T - W_t$ and W_t are independent, therefore conditioning on $W_t = x$ is irrelevant. The conclusion is that if $u(x, t)$ is a solution to (43), then $u(x, t) = \mathbb{E}[\psi(x + W_T - W_t)]$. Now we need to remove the expectation. For that, recall that $W_T - W_t$ is normally distributed and therefore $W_T - W_t = \sqrt{T-t}Z$ with $Z \sim \mathcal{N}(0, 1)$. Therefore, using the probability distribution function of the Gaussian distribution, we see that

$$u(x, t) = \mathbb{E}[\psi(x + \sqrt{T-t}Z)] = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\infty} \psi(x + \sqrt{T-t}z) e^{-z^2/2} dz,$$

this is to be compared to (5). This is the solution to the heat equation that would have been obtained through Fourier transform methods or through similarity reductions.

B The full Feynman-Kac formula in one dimension

This idea can be generalized with very little work to the following situation. Suppose we have

$$\frac{\partial u}{\partial t} + a(x) \frac{\partial^2 u}{\partial x^2} + b(x) \frac{\partial u}{\partial x} = c(x)u, \quad \text{for } t < T,$$

and $u(x, T) = \psi(x)$. We can associate to this PDE a stochastic process X_t manufactured from these coefficients as the SDE $dX_t = b(X_t)dt + \sqrt{2a(X_t)}dW_t$. Then Feynman-Kac

is the result that $u(x, t) = \mathbb{E}[\exp(-\int_t^T c(X_u)du)\psi(X_T) | X_t = x]$. All issues of existence and uniqueness can be found in Shreve and Karatzas [31, §4]. The sketch of the argument leading to Feynman-Kac formula is to first set

$$Y_t := \exp \left[- \int_0^t c(X_s)ds u(X_t, t) \right],$$

and to then employ Itô to show that

$$\frac{dY_t}{\exp[-\int_0^t c(X_s)ds]} = \left(\frac{\partial u}{\partial t} + a(X_t) \frac{\partial^2 u}{\partial x^2} + b(X_t) \frac{\partial u}{\partial x} - c(X_t)u \right) dt + \frac{\partial u}{\partial x} dW_t,$$

with the derivatives of u evaluated at (X_t, t) . This allows us to establish that if u does solve the PDE, then Y_t above has zero drift (i.e. it is a martingale) and then write $\mathbb{E}[Y_t | X_t = x] = u(x, t)$. The details can be found in [65].

C Linking the Feynman-Kac PDE to the Schrödinger equation

Let us make the substitution $u(x, t) = e^{g(x, t)}v(x, t)$, for some analytic functions g and v . Substituting the newly transformed derivatives appearing into (10) and re-grouping terms, we obtain

$$\begin{aligned} -\frac{\partial v}{\partial t} &= \mathfrak{a}(x, t) \frac{\partial^2 v}{\partial x^2} + \left(2\mathfrak{a}(x, t) \frac{\partial g}{\partial x} + \mathfrak{b}(x, t) \right) \frac{\partial v}{\partial x} \\ &\quad + \left(\frac{\partial g}{\partial t} + \mathfrak{a}(x, t) \left(\frac{\partial g}{\partial x} \right)^2 + \mathfrak{a}(x, t) \frac{\partial^2 g}{\partial x^2} + \mathfrak{b}(x, t) \frac{\partial g}{\partial x} - \mathfrak{c}(x, t) \right) v. \end{aligned} \quad (47)$$

Setting the coefficient of the unwanted term to be zero yields

$$2\mathfrak{a}(x, t) \frac{\partial g}{\partial x} + \mathfrak{b}(x, t) = 0 \Leftrightarrow \frac{\partial g}{\partial x} = -\frac{\mathfrak{b}(x, t)}{2\mathfrak{a}(x, t)}. \quad (48)$$

We may integrate this to

$$g(x, t) = - \int \frac{\mathfrak{b}(x, t)}{2\mathfrak{a}(x, t)} dx + h(t), \quad (49)$$

where h is an arbitrary function of t only. Using (48) on the coefficient of the linear term v appearing in (47) we get the expression

$$\begin{aligned} \mathfrak{w}(x, t) &:= \frac{\partial g}{\partial t} + \mathfrak{a}(x, t) \left(\frac{\partial g}{\partial x} \right)^2 + \mathfrak{a}(x, t) \frac{\partial^2 g}{\partial x^2} + \mathfrak{b}(x, t) \frac{\partial g}{\partial x} - \mathfrak{c}(x, t) \\ &= \frac{\partial h}{\partial t} - \frac{\partial}{\partial t} \int \frac{\mathfrak{b}(x, t)}{2\mathfrak{a}(x, t)} dx - \frac{\mathfrak{b}^2(x, t)}{4\mathfrak{a}(x, t)} - \frac{1}{2} \mathfrak{a}(x, t) \left(\frac{\partial}{\partial x} \frac{\mathfrak{b}(x, t)}{\mathfrak{a}(x, t)} \right) - \mathfrak{c}(x, t). \end{aligned}$$

Since h was an arbitrary function we may take it to be $h(t) = 0$ so that $\frac{dh}{dt} = 0$ as well. Effectively, this implies that

$$g(x, t) = - \int^x \frac{\mathfrak{b}(y, t)}{2\mathfrak{a}(y, t)} dy,$$

where the integration sign \int^x indicates that the result of the integration with respect to y is a function of x without a constant of integration. This implies that

$$\mathfrak{w}(x, t) = -\frac{\partial}{\partial t} \int \frac{\mathfrak{b}(x, t)}{2\mathfrak{a}(x, t)} dx - \frac{\mathfrak{b}^2(x, t)}{4\mathfrak{a}(x, t)} - \frac{1}{2} \mathfrak{a}(x, t) \left(\frac{\partial}{\partial x} \frac{\mathfrak{b}(x, t)}{\mathfrak{a}(x, t)} \right) - \mathfrak{c}(x, t). \quad (50)$$

In a substantial number of interesting cases the coefficients \mathfrak{a} , \mathfrak{b} and \mathfrak{c} are functions of x only, which will further simplify (50). Therefore, using these auxiliary results, (10) becomes

$$0 = \frac{\partial v}{\partial t} + \mathfrak{a}(x, t) \frac{\partial^2 v}{\partial x^2} + \mathfrak{w}(x, t)v.$$

There are two additional substitutions we need to make. First, we need to turn this into an initial value problem by setting $\tau = T - t$, in which case we end up with

$$\frac{\partial v}{\partial \tau} = \mathfrak{a}(x, \tau) \frac{\partial^2 v}{\partial x^2} + \mathfrak{w}(x, \tau)v. \quad (51)$$

Lastly, the Wick rotation $\xi = -i\tau$ transforms (51) into

$$-i \frac{\partial v}{\partial \xi} = \mathfrak{a}(x, \xi) \frac{\partial^2 v}{\partial x^2} + \mathfrak{w}(x, \xi)v.$$

Note that some straightforward modifications to the time variable need to be applied in these last two transformations to the functions u and v to keep consistency.

D Additional norm plots

To demonstrate the efficacy of the ℓ_1 -norm enforcement procedure as explained in Sec 3.4, we provide additional plots in Figure 11 comparing the VarQITE method against Monte Carlo and forward Euler methods.

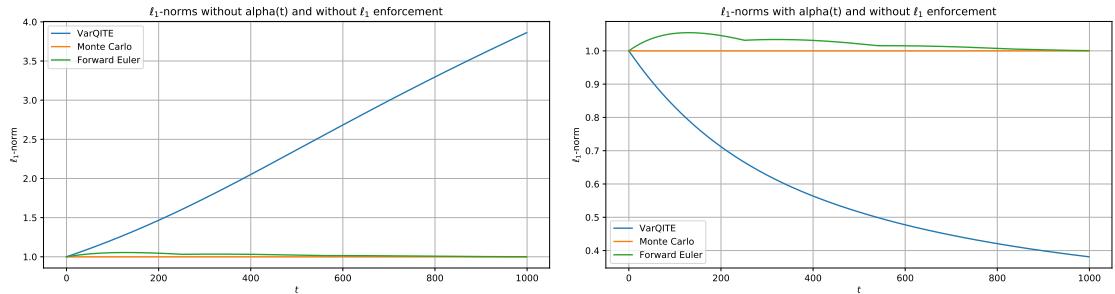


Figure 11: Plots of ℓ_1 norms for the first 100 evolutions with six qubits and $n = 1$ layers of ansatz, left: without $\alpha(t)$ and without the ℓ_1 -norm enforcement procedure and right: with $\alpha(t)$ but without the ℓ_1 -norm enforcement procedure.

E Backward propagation and option pricing

To apply the introduced technique to option pricing, a different application of the Feynman-Kac formula is required: one considers propagation backwards in time instead of forward, i.e., the final payoff is used as initial condition and we propagate back in time to the starting point of the process. Every element of the result then corresponds to the fair option price for the corresponding initial state.

For instance, one may wish to solve the heat equation by specializing $\mathfrak{a}(x, \xi) = \frac{1}{2}$ and $\mathfrak{w}(x, \xi) = 0$ as well as the payoff function $\psi(X_T) = \max(X_T - K, 0)$, where K now represents the strike of a European option with maturity T . The rationale behind is that the Black-Scholes equation can be written as the heat equation using suitable transformations. Stepping into the domain of path-dependent options, another popular financial derivative

is the Asian option, see [63]. To price it, we specialize the coefficients of the Hamiltonian to $\mathfrak{a}(x, \xi) = \frac{(q(\xi) - x)^2}{2}$ and $\mathfrak{w}(x, \xi) = 0$, where $q(t) = \frac{1 - e^{-r(T-t)}}{rT}$ if $r \neq 0$ and $q(t) = 1 - \frac{t}{T}$ if $r = 0$. The payoff is given by the average of process X_t , i.e.

$$\psi(X_T, T) = \max \left(\frac{1}{T} \int_0^T X_u du - K, 0 \right). \quad (52)$$

In this case, (52) represents an Asian option with strike K and maturity T . We remark that the payoff function ψ now depends on T . These cases are studied in [11, 10, 23, 18], not always under the optic of variational quantum imaginary time evolution. The European case is showcased in [9, 30] by amplitude estimation and without applying variational principles.

Much like in the case of forward propagation, cf. Section 3.3, the performance of the algorithm when going backward in time needs to be carefully analyzed. First, the payoff function, which is now considered as the initial condition, needs to be loaded into a quantum state, i.e., it needs to be normalized and we need to prepare a corresponding quantum circuit. Let us consider a European call option for illustration with $X_T \in [X_l, X_u]$, $X_u - X_l = B$, and $K = (X_l + X_u)/2$. If we discretize X_T using n qubits, i.e. with 2^n grid points and grid size $h = B/2^n$, then the normalization factor of the quantum state scales as $\theta(B\sqrt{2^n})$. As in forward propagation, this limits the potential advantage already to a quadratic speed-up, since we need to re-scale any result –including the estimation error– by a factor of size $\theta(B\sqrt{2^n})$ to get back to the original scale. In addition to the normalization, constructing a circuit to prepare arbitrary quantum states can be prohibitive [66]. For the simple payoffs considered efficient circuits should exist, and for more generic functions variational approach as suggested in [11] or [67] may be used.

Suppose the initial state has been loaded, the algorithm has been applied to estimate a prepared quantum state that corresponds to the solution and one is interested in the result. The impact of the α -factor might be negligible here, since we already start in a quantum state with rather uniform amplitudes. However, the option price for a particular target initial price corresponds now to the amplitude of a single computational basis state. The readout of this can be achieved via QAE up to accuracy $\varepsilon > 0$ at cost $\mathcal{O}(1/\varepsilon)$. Since the amplitude of interest likely will be of order $\mathcal{O}(1/\sqrt{2^n})$, the required ε will be of equal scale. Combining the requirements coming from the normalization likely renders this approach impractical for option pricing.

F Analytical solution

The equation we need to solve is

$$\frac{\partial u}{\partial t} = A \frac{\partial^2 u}{\partial x \partial y} + B \frac{\partial^2 u}{\partial x^2} + C \frac{\partial^2 u}{\partial y^2} - ru, \quad (53)$$

with $u = u(x, y, t)$, and initial condition $u(x, y, 0) = \delta(x - x_0)\delta(y - y_0)$, with $x_0, y_0 > 0$. Here A, B, C and r are all real constants such that $4BC - A^2 > 0$. The idea is to apply a Laplace transform on the time variable and a Mellin transform on each of the space variables. This will allow us to change the PDE into an algebraic equation which we can solve trivially. Then we proceed to invert the Laplace transforms as well as the Mellin transforms to recover the solution of (53).

Let us start with

$$\frac{\partial u}{\partial t} = Axy \frac{\partial^2 u}{\partial x \partial y} + Bx^2 \frac{\partial^2 u}{\partial x^2} + Cy^2 \frac{\partial^2 u}{\partial y^2} - ru + Dx \frac{\partial u}{\partial x} + Ey \frac{\partial u}{\partial y}, \quad (54)$$

with D and E real constants. Make the changes

$$X = \log x, \quad Y = \log y. \quad (55)$$

Then from (55) we see that

$$\frac{\partial}{\partial x} = \frac{1}{x} \frac{\partial}{\partial X}, \quad \frac{\partial^2}{\partial x^2} = \frac{1}{x^2} \left(\frac{\partial^2}{\partial X^2} - \frac{\partial}{\partial X} \right) \quad \text{and} \quad \frac{\partial}{\partial y} = \frac{1}{y} \frac{\partial}{\partial Y}, \quad \frac{\partial^2}{\partial y^2} = \frac{1}{y^2} \left(\frac{\partial^2}{\partial Y^2} - \frac{\partial}{\partial Y} \right). \quad (56)$$

Using (56) on (54) yields

$$\begin{aligned} \frac{\partial u}{\partial t} &= Axy \frac{1}{x} \frac{\partial}{\partial X} \frac{1}{y} \frac{\partial}{\partial Y} u + Bx^2 \frac{1}{x^2} \left(\frac{\partial^2}{\partial X^2} - \frac{\partial}{\partial X} \right) u + Cy^2 \frac{1}{y^2} \left(\frac{\partial^2}{\partial Y^2} - \frac{\partial}{\partial Y} \right) u - ru \\ &\quad + Dx \frac{1}{x} \frac{\partial}{\partial X} u + Ey \frac{1}{y} \frac{\partial}{\partial Y} u \\ &= A \frac{\partial^2 u}{\partial X \partial Y} + B \frac{\partial^2 u}{\partial X^2} + C \frac{\partial^2 u}{\partial Y^2} + (D - B) \frac{\partial u}{\partial X} + (E - C) \frac{\partial u}{\partial Y} - ru. \end{aligned} \quad (57)$$

Now, (57) will have the same structure as (53) if

$$D = B \quad \text{and} \quad E = C, \quad (58)$$

minding that x, y are related to X, Y by (55). Thus, we need to solve (54) with D and E satisfying the conditions given by (58). The solution to (54) when (58) holds is given by

$$u(x, y, t) = \frac{1}{2\pi\sqrt{\Delta}} \frac{e^{-rt}}{x_0 y_0 t} \exp \left[-\frac{B}{\Delta t} \left(\frac{A}{2B} \log \frac{x}{x_0} - \log \frac{y}{y_0} \right)^2 - \frac{1}{4Bt} \log^2 \frac{x}{x_0} \right]$$

where $\Delta = 4BC - A^2 > 0$. Using $X_0 = \log x_0$ and $Y_0 = \log y_0$ in addition to (55) (that is, reversing the change of variables), then the solution to (53) when (58) holds is given by

$$u(x, y, t) = \frac{e^{-rt-x_0-y_0}}{2\pi t \sqrt{\Delta}} \exp \left[-\frac{B}{\Delta t} \left(\frac{A}{2B} (x - x_0) - (y - y_0) \right)^2 - \frac{1}{4Bt} (x - x_0)^2 \right].$$

We start by taking the Laplace transform of u with respect to t so that

$$\begin{aligned} \hat{u}(x, y, s) &= \mathcal{L}\{u(x, y, t)\}(s) = \int_0^\infty e^{-st} u(x, y, t) dt \\ &= -\frac{1}{s} e^{-st} u(x, y, t) \Big|_{t=0} + \frac{1}{s} \int_0^\infty e^{-st} \left(\frac{\partial}{\partial t} u(x, y, t) \right) dt \\ &\quad + \frac{1}{s} \left(Axy \frac{d^2}{dxdy} + Bx^2 \frac{d^2}{dx^2} + Cy^2 \frac{d^2}{dy^2} - r + Dx \frac{d}{dx} + Ey \frac{d}{dy} \right) \\ &\quad \times \int_0^\infty e^{-st} u(x, y, t) dt \\ &= \frac{1}{s} \delta(x - x_0) \delta(y - y_0) \\ &\quad + \frac{1}{s} \left(Axy \frac{d^2}{dxdy} + Bx^2 \frac{d^2}{dx^2} + Cy^2 \frac{d^2}{dy^2} - r + Dx \frac{d}{dx} + Ey \frac{d}{dy} \right) \hat{u}(x, y, s). \end{aligned}$$

Next take a Mellin transform of \hat{u} with respect to x . This gives us

$$\tilde{u}(w_1, y, s) = \mathcal{M}\{\hat{u}(x, y, s)\}(w_1) = \int_0^\infty x^{w_1-1} \hat{u}(x, y, s) dx$$

$$\begin{aligned}
&= \frac{1}{s} \delta(y - y_0) \mathcal{M}\{\delta(x - x_0)\}(w_1) + \frac{1}{s} A y \frac{d}{dy} \mathcal{M}\left\{x \frac{d}{dx} \hat{u}(x, y, s)\right\}(w_1) \\
&\quad + \frac{1}{s} B \mathcal{M}\left\{x^2 \frac{d^2}{dx^2} \hat{u}(x, y, s)\right\}(w_1) + \frac{1}{s} C y^2 \frac{d^2}{dy^2} \mathcal{M}\{\hat{u}(x, y, s)\}(w_1) \\
&\quad - \frac{r}{s} \mathcal{M}\{\hat{u}(x, y, s)\}(w_1) + \frac{1}{s} D \mathcal{M}\left\{x \frac{d}{dx} \hat{u}(x, y, s)\right\}(w_1) \\
&\quad + \frac{1}{s} E y \frac{d}{dy} \mathcal{M}\{\hat{u}(x, y, s)\}(w_1).
\end{aligned} \tag{59}$$

We shall need to employ [68]

$$\mathcal{M}\{xf'(x)\}(w) = -w\mathcal{M}\{f(x)\}(w) \quad \text{and} \quad \mathcal{M}\{x^2 f''(x)\}(w) = w(1+w)\mathcal{M}\{f(x)\}(w)$$

to evaluate the terms appearing in (59). Using these Mellin transforms leads us to

$$\begin{aligned}
\tilde{u}(w_1, y, s) &= \frac{x_0^{w_1-1}}{s} \delta(y - y_0) - \frac{Ay}{s} w_1 \frac{d}{dy} \tilde{u}(w_1, y, s) + \frac{B}{s} w_1 (1+w_1) \tilde{u}(w_1, y, s) \\
&\quad + \frac{C}{s} y^2 \frac{d^2}{dy^2} \tilde{u}(w_1, y, s) - \frac{r}{s} \tilde{u}(w_1, y, s) - \frac{D}{s} w_1 \tilde{u}(w_1, y, s) + \frac{E}{s} y \frac{d}{dy} \tilde{u}(w_1, y, s).
\end{aligned}$$

Another Mellin transform of $\tilde{u}(w_1, y, s)$, this time with respect to y , yields

$$\begin{aligned}
\tilde{\tilde{u}}(w_1, w_2, s) &= \mathcal{M}\{\tilde{u}(w_1, y, s)\}(w_2) = \int_0^\infty y^{w_2-1} \tilde{u}(w_1, y, s) dy \\
&= \frac{x_0^{w_1-1} y_0^{w_2-1}}{s} + \frac{A}{s} w_1 w_2 \tilde{\tilde{u}}(w_1, w_2, s) + \frac{B}{s} w_1 (1+w_1) \tilde{\tilde{u}}(w_1, w_2, s) \\
&\quad + \frac{C}{s} w_2 (1+w_2) \tilde{\tilde{u}}(w_1, w_2, s) - \frac{r}{s} \tilde{\tilde{u}}(w_1, w_2, s) - \frac{D}{s} w_1 \tilde{\tilde{u}}(w_1, w_2, s) \\
&\quad - \frac{E}{s} w_2 \tilde{\tilde{u}}(w_1, w_2, s).
\end{aligned}$$

Now we solve for $\tilde{\tilde{u}}(w_1, w_2, s)$ in the above equation

$$\tilde{\tilde{u}}(w_1, w_2, s) = \frac{x_0^{w_1-1} y_0^{w_2-1}}{s - (Aw_1 w_2 + w_1(B - D + Bw_1) + w_2(C - E + Cw_2) - r)}. \tag{60}$$

We shall now bring in (58). Strictly speaking it is not necessary to use them and there is some loss of generality. However, it makes the problem much easier and using more generality in the coefficients is straightforward. Thus using (58) we see that (60) reduces to

$$\tilde{\tilde{u}}(w_1, w_2, s) = \frac{x_0^{w_1-1} y_0^{w_2-1}}{s - (Aw_1 w_2 + Bw_1^2 + Cw_2^2 - r)}. \tag{61}$$

The task, now that we have found $\tilde{\tilde{u}}(w_1, w_2, s)$ in terms of w_1, w_2 and s , is to revert back all the transforms and translate the answer to the space of solutions of (53). Apply the inverse Laplace transform $\mathcal{L}^{-1}\{\frac{1}{s-a}\}(t) = e^{at}$ to (61) so that we end up with

$$\begin{aligned}
\tilde{\tilde{u}}(w_1, w_2, t) &= x_0^{w_1-1} y_0^{w_2-1} \exp\left[\left(Aw_1 w_2 + Bw_1^2 + Cw_2^2 - r\right)t\right] \\
&= e^{-rt} x_0^{w_1-1} y_0^{w_2-1} \exp\left[Ctw_2^2\right] \exp\left[\alpha w_1 + \beta w_1^2\right],
\end{aligned}$$

where $\alpha = Atw_2$, and $\beta = Bt$. To invert the Mellin transform with respect to x we use the following $G(s) = e^{\alpha s + \beta s^2} = e^{\alpha s} e^{\beta s^2} = G_1(s)G_2(s)$. Then the inverses of G_1 and G_2 are given by [68]

$$g_1(x) = \mathcal{M}^{-1}\{e^{\alpha s}\}(x) = \delta(\alpha - \log x)$$

as well as

$$g_2(x) = \mathcal{M}^{-1}\{e^{\beta s^2}\}(x) = \frac{1}{2\sqrt{\pi\beta}} \exp\left(-\frac{\log^2 x}{4\beta}\right).$$

Thus the inverse of G can be found by computing the convolution

$$\begin{aligned} g(x) &= \int_0^\infty g_1(z)g_2\left(\frac{x}{z}\right)\frac{dz}{z} = \int_0^\infty \delta(\alpha - \log z)\frac{1}{2\sqrt{\pi\beta}} \exp\left(-\frac{\log^2 \frac{x}{z}}{4\beta}\right)\frac{dz}{z} \\ &= \frac{1}{2\sqrt{\pi\beta}} \exp\left[-\frac{(\alpha - \log x)^2}{4\beta}\right], \end{aligned}$$

since $\alpha \in \mathbb{R}$. Moving on from $\tilde{u}(w_1, w_2, t)$ we invert w_1 back into x and see that

$$\begin{aligned} \tilde{u}(x, w_2, t) &= e^{-rt} y_0^{w_2-1} \exp\left[Ctw_2^2\right] \mathcal{M}^{-1}\left\{x_0^{w_1-1} \exp\left[\alpha w_1 + \beta w_1^2\right]\right\}(x) \\ &= e^{-rt} y_0^{w_2-1} \exp\left[Ctw_2^2\right] \int_0^\infty \delta(z - x_0) \frac{1}{2\sqrt{\pi\beta}} \exp\left[-\frac{(\alpha - \log \frac{x}{z})^2}{4\beta}\right] \frac{dz}{z} \\ &= \frac{e^{-rt}}{2\sqrt{\pi\beta}} y_0^{w_2-1} \exp\left[Ctw_2^2\right] \frac{1}{x_0} \exp\left[-\frac{(\alpha - \log \frac{x}{x_0})^2}{4\beta}\right] \\ &= \frac{e^{-rt}}{2x_0\sqrt{B\pi t}} \exp\left[-\frac{\log^2 \frac{x}{x_0}}{4Bt}\right] y_0^{w_2-1} \exp\left[\alpha w_2 + \beta w_2^2\right], \end{aligned}$$

where we have recycled α and β to now be $\alpha = \frac{A}{2B} \log \frac{x}{x_0}$ and $\beta = Ct - \frac{A^2 t}{4B}$. Lastly, we Mellin invert w_2 back into y and we arrive at

$$\begin{aligned} u(x, y, t) &= \frac{e^{-rt}}{2x_0\sqrt{B\pi t}} \exp\left[-\frac{\log^2 \frac{x}{x_0}}{4Bt}\right] \mathcal{M}^{-1}\left\{y_0^{w_2-1} \exp\left[\alpha w_2 + \beta w_2^2\right]\right\}(y) \\ &= \frac{e^{-rt}}{2x_0\sqrt{B\pi t}} \exp\left[-\frac{\log^2 \frac{x}{x_0}}{4Bt}\right] \int_0^\infty \delta(z - y_0) \frac{1}{2\sqrt{\pi\beta}} \exp\left[-\frac{(\alpha - \log \frac{y}{z})^2}{4\beta}\right] \frac{dz}{z} \\ &= \frac{e^{-rt}}{2x_0y_0\sqrt{B\pi t}} \frac{1}{2\sqrt{\pi\beta}} \exp\left[-\frac{\log^2 \frac{x}{x_0}}{4Bt}\right] \exp\left[-\frac{(\alpha - \log \frac{y}{y_0})^2}{4\beta}\right]. \end{aligned}$$

By setting $\Delta = 4BC - A^2$, and doing some mild algebraic re-arrangements we end up with

$$u(x, y, t) = \frac{1}{2\pi\sqrt{\Delta}} \frac{e^{-rt}}{x_0y_0t} \exp\left[-\frac{B}{\Delta t} \left(\frac{A}{2B} \log \frac{x}{x_0} - \log \frac{y}{y_0}\right)^2 - \frac{1}{4Bt} \log^2 \frac{x}{x_0}\right],$$

which is the solution we were seeking.

G Moments

While obtaining the full solution u to the PDF (19) generated by the Feynman-Kac algorithm is an expensive problem, familiar within the field of quantum metrology, we may nevertheless extract information about the moments of u with a quadratic advantage using

QPE which are deep circuits, see Section 3.3. Alternatively we could use Hadamard tests which are not as efficient but they consist of shallower circuits. This section generalizes to several dimensions some of the concepts introduced in [12], which then get applied to the system of SDEs introduced in (15). Along the way, we take this chance to amend some inaccuracies from [12]. To avoid confusion with exponentiation, the labeling $i = 1, 2, \dots, D$ of the processes X_t^i will now be subscripted and the time variable t will be suppressed.

Recall that $n_m = 2^{n_q/D}$, where n_q is the total number of qubits and D the number of dimensions, represents the number of grid points allocated to each dimension. Suppose we have a function $f : \mathbb{R}^D \rightarrow \mathbb{R}$ and consider the expectation

$$\begin{aligned} & \mathbb{E}[f(X_1(T), X_2(T), \dots, X_D(T))] \\ &= \sum_{i_1=0}^{n_m-1} \sum_{i_2=0}^{n_m-1} \dots \sum_{i_D=0}^{n_m-1} f(x_{i_1}, x_{i_2}, \dots, x_{i_D}) \\ & \quad \times \mathbb{P}_{X_1, X_2, \dots, X_D}[X_1(T) = x_{i_1}, X_2(T) = x_{i_2}, \dots, X_D(T) = x_{i_D}]. \end{aligned} \quad (62)$$

Here $u(x_{i_1}, x_{i_2}, \dots, x_{i_D}, t) = \mathbb{P}_{X_1, X_2, \dots, X_D}$ denotes the joint probability distribution function of the random variables X_1, X_2, \dots, X_D in situations where we have ℓ_1 normalization. We have D dimensions of equal length $[0, x_{i,\max}] = [0, x_{\max}]$ and in each one of them we split the interval $[0, x_{i,\max}]$ into $N_{x_i} = N_x = d$ sub-intervals spaced by the same lattice spacing $\Delta x_i = \Delta x = h$ for $i = 1, 2, \dots, D$ and the total number of qubits gets evenly allocated to each of the D dimensions. The splitting of $[0, x_{i,\max}]$ is

$$\{[0, a_{i,1}], [a_{i,1}, a_{i,2}], \dots, [a_{i,d-1}, x_{i,\max}]\} =: \Upsilon(i), \quad (63)$$

that is, we have d intervals in each dimension D , yielding a total of d^D intervals in all dimensions altogether. For an integer n we have $x_n = n\Delta x$. We further set the notation

$$\alpha_{i,k} := [a_{i,k}, a_{i,k+1}] \in \Upsilon(i), \quad \text{for } k = 0, 1, 2, \dots, d-1, \quad \text{and } i = 1, 2, \dots, D. \quad (64)$$

Let us recall Taylor's expansion in several dimensions. In order to do so, we need to introduce some notation first. If $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_n)$ denotes an n -tuple of nonnegative integers, then $\mathbf{x}^\alpha = x_1^{\alpha_1} x_2^{\alpha_2} \dots x_n^{\alpha_n}$ where $\mathbf{x} = (x_1, x_2, \dots, x_n) \in \mathbb{R}^n$. The sum $|\alpha| = \alpha_1 + \alpha_2 + \dots + \alpha_n$ is the order, or the degree, of α . Moreover $\partial^\alpha = \partial_1^{\alpha_1} \partial_2^{\alpha_2} \dots \partial_n^{\alpha_n} f = \frac{\partial^{|\alpha|} f}{\partial x_1^{\alpha_1} \partial x_2^{\alpha_2} \dots \partial x_n^{\alpha_n}}$ and $\alpha! = \alpha_1! \alpha_2! \dots \alpha_n!$. Thus the order of α is the same as the order of \mathbf{x}^α as a monomial or the order of ∂^α as a partial derivative.

In the interval $\alpha_{i,k}$, the function $f(\mathbf{x}) = f(x_1, x_2, \dots, x_D)$ is assumed to be well approximated by its L th order Taylor series (i labels dimension and k labels interval in the dimension)

$$f_{i,k}(\mathbf{x}) = \sum_{|m| \leq L_{i,k}} \frac{\partial^m f_{i,k}(\mathbf{0})}{m!} \mathbf{x}^m =: \sum_{|m| \leq L_{i,k}} f_{i,k,m} \mathbf{x}^m. \quad (65)$$

A natural assumption that we shall use towards the end is that $L_{i,k}$ is the same for each sub-interval of each dimension and hence $L_{i,k} = L$. The error in the expectation (62) due to this approximation will be discussed shortly. Due to the finiteness of \mathbf{x} , we can find an appropriate shift to have the range of f be positive and this will ensure that the expectation (62) will also be positive.

In D dimensions the unnormalized state that we are working with is given by

$$|\tilde{\psi}(t)\rangle = \sum_{i_1=0}^{n_m-1} \sum_{i_2=0}^{n_m-1} \dots \sum_{i_D=0}^{n_m-1} \mathbb{P}_{X_1, X_2, \dots, X_D}[X_1(t) = x_{i_1}, X_2(t) = x_{i_2}, \dots, X_D(t) = x_{i_D}] |\mathbf{i}\rangle \quad (66)$$

where $\mathbf{i} = |i_1, i_2, \dots, i_D\rangle$. To compute the expectation (62) we first consider the non-unitary operator satisfying

$$S_f |\mathbf{0}\rangle = \sum_{i_1=0}^{n_m-1} \sum_{i_2=0}^{n_m-1} \cdots \sum_{i_D=0}^{n_m-1} f(x_{i_1}, x_{i_2}, \dots, x_{i_D}) |\mathbf{i}\rangle. \quad (67)$$

This operator can be decomposed into easily implementable unitaries Q , i.e.

$$S_f = \bigotimes_{d=1}^D \sum_{k_d} \xi_{k_d} Q_{k_d}, \quad (68)$$

where $\xi_{k_d} \in \mathbb{C}$. Taking $|\mathbf{0}\rangle \langle \mathbf{0}| = |0, 0, \dots, 0\rangle \langle 0, 0, \dots, 0|$, a calculation shows that

$$\begin{aligned} \langle \tilde{\psi}(T) | (S_f |\mathbf{0}\rangle \langle \mathbf{0}| S_f^\dagger) |\tilde{\psi}(T)\rangle &= \prod_{i=1}^D \sum_{a_i=0}^{n_m-1} \mathbb{P}_{X_1, \dots, X_D}[X_1(T) = x_{a_1}, \dots, X_D(T) = x_{a_D}] \\ &\quad \times \langle \mathbf{a} | \prod_{j=1}^D \sum_{b_j=0}^{n_m-1} f(x_{b_1}, \dots, x_{b_D}) |\mathbf{b}\rangle \\ &\quad \times \prod_{k=1}^D \sum_{c_k=0}^{n_m-1} \mathbb{P}_{X_1, \dots, X_D}[X_1(T) = x_{c_1}, \dots, X_D(T) = x_{c_D}] \\ &\quad \times \langle \mathbf{c} | \prod_{\ell=1}^D \sum_{d_\ell=0}^{n_m-1} f^*(x_{d_1}, \dots, x_{d_D}) |\mathbf{d}\rangle \\ &= \sum_{a_1=0}^{n_m-1} \sum_{a_2=0}^{n_m-1} \cdots \sum_{a_D=0}^{n_m-1} f(x_{a_1}, x_{a_2}, \dots, x_{a_D}) \\ &\quad \times \mathbb{P}_{X_1, X_2, \dots, X_D}[X_1(T) = x_{a_1}, X_2(T) = x_{a_2}, \dots, X_D(T) = x_{a_D}] \\ &\quad \times \sum_{c_1=0}^{n_m-1} \sum_{c_2=0}^{n_m-1} \cdots \sum_{c_D=0}^{n_m-1} f(x_{c_1}, x_{c_2}, \dots, x_{c_D}) \\ &\quad \times \mathbb{P}_{X_1, X_2, \dots, X_D}[X_1(T) = x_{c_1}, X_2(T) = x_{c_2}, \dots, X_D(T) = x_{c_D}] \\ &= (\mathbb{E}[f(X_1(T), X_2(T), \dots, X_D(T))])^2, \end{aligned} \quad (69)$$

since f was a real function. We may now use some of the results presented in 3.2. For instance $|0\rangle \langle 0| = (\mathbb{I} - X^{\otimes n} \cdot C^{n-1} Z \cdot X^{\otimes n})/2$ is a sum of easily implementable unitaries, and therefore $|\mathbf{0}\rangle \langle \mathbf{0}|$ will also be implementable as such a decomposition by the use of $|\mathbf{0}\rangle \langle \mathbf{0}| = (\mathbb{I} - X^{\otimes nD} \cdot C^{nD-1} Z \cdot X^{\otimes nD})/2$.

Using (68) we can write the first and last terms of the left-hand side of (69) as

$$S_f |\mathbf{0}\rangle \langle \mathbf{0}| S_f^\dagger = \left(\bigotimes_{d=1}^D \sum_{i_d} \xi_{i_d} Q_{i_d} \right) \left(\frac{\mathbb{I} - X^{\otimes nD} \cdot C^{nD-1} Z \cdot X^{\otimes nD}}{2} \right) \left(\bigotimes_{d'=1}^D \sum_{i'_{d'}} \xi_{i'_{d'}}^* Q_{i'_{d'}}^\dagger \right) \quad (70)$$

This is the sum of tensor products of unitaries. Thus in (69) we will only have expressions of the type

$$\langle \tilde{\psi}(T) | Q_{i_d} Q_{i'_{d'}}^\dagger | \tilde{\psi}(T)\rangle \quad \text{and} \quad \langle \tilde{\psi}(T) | Q_{i_d} (X^{\otimes nD} \cdot C^{nD-1} Z \cdot X^{\otimes nD}) Q_{i'_{d'}}^\dagger | \tilde{\psi}(T)\rangle. \quad (71)$$

Two techniques will yield the quantities in (71). One could either use the Hadamard test circuits depicted in this section or use QPE [69, 70]. The circuit for the Hadamard test is given by Fig 12.

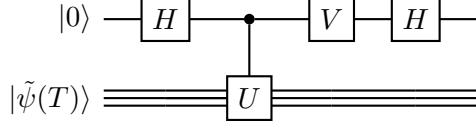


Figure 12: This circuit evaluates the Hadamard tests for (62), with U from (71) and V selects the real and imaginary parts.

Here the U 's are gates such that $U \in \{Q_{i_d}Q_{i'_{d'}}^\dagger, Q_{i_d}(X^{\otimes nD} \cdot C^{nD-1}Z \cdot X^{\otimes nD})Q_{i'_{d'}}^\dagger\}$ and

$$V := \begin{cases} \mathbb{I} & \text{for } \operatorname{Re} \langle \tilde{\psi}(T) | U | \tilde{\psi}(T) \rangle, \\ S^\dagger & \text{for } \operatorname{Im} \langle \tilde{\psi}(T) | U | \tilde{\psi}(T) \rangle. \end{cases}$$

We shall get to the pros and cons of each technique towards the end of this section, however, it is noteworthy that the circuit has the advantage of being shallow but not efficient in the terms of the number of measurements needed.

The next step is to define the operator

$$\begin{aligned} S_{\chi[a_1,b_1][a_2,b_2]\dots[a_D,b_D]} |\mathbf{0}\rangle &= \sum_{i_1=0}^{n_m-1} \sum_{i_2=0}^{n_m-1} \dots \sum_{i_D=0}^{n_m-1} \chi[a_1, b_1](x_{i_1}) \\ &\quad \times \chi[a_2, b_2](x_{i_2}) \dots \chi[a_D, b_D](x_{i_D}) |\mathbf{i}\rangle \\ &= \sum_{i_1=0}^{n_m-1} \chi[a_1, b_1](x_{i_1}) \sum_{i_2=0}^{n_m-1} \chi[a_2, b_2](x_{i_2}) \dots \sum_{i_D=0}^{n_m-1} \chi[a_D, b_D](x_{i_D}) |\mathbf{i}\rangle \\ &= \sum_{x_{i_1} \in [a_1, b_1]} \sum_{x_{i_2} \in [a_2, b_2]} \dots \sum_{x_{i_D} \in [a_D, b_D]} |\mathbf{i}\rangle \\ &= \sum_{\substack{j=1,2,\dots,D \\ x_{i_j} \in [a_j, b_j]}} |\mathbf{i}\rangle, \end{aligned} \tag{72}$$

where the interval indicator function χ is defined by

$$\chi[a, b](x) := \begin{cases} 1 & \text{if } x \in [a, b], \\ 0 & \text{otherwise.} \end{cases}$$

We must now connect S_f coming from (67) to $S_{\chi[0,a_1][0,a_2]\dots[0,a_D]}$ from (72). Note that

$$\begin{aligned} S_f |\mathbf{0}\rangle &= \sum_{i_1=0}^{n_m-1} \sum_{i_2=0}^{n_m-1} \dots \sum_{i_D=0}^{n_m-1} f(x_{i_1}, x_{i_2}, \dots, x_{i_D}) |\mathbf{i}\rangle \\ &= \prod_{j=1}^D \sum_{i_j=0}^{n_m-1} \sum_{k_j=0}^{d-1} \chi[\alpha_{j,k_j}](x_{i_j}) f_{j,k_j}(x_{i_1}, x_{i_2}, \dots, x_{i_D}) |\mathbf{i}\rangle \\ &= \prod_{j=1}^D \sum_{k_j=0}^{d-1} \sum_{i_j=0}^{n_m-1} \sum_{\substack{j=1,2,\dots,D \\ x_{i_j} \in \alpha_{j,k_j}}} f_{j,k_j}(x_{i_1}, x_{i_2}, \dots, x_{i_D}) |\mathbf{i}\rangle. \end{aligned} \tag{73}$$

In order to perform the sums over x_{i_j} 's we use a multidimensional Taylor series

$$S_f |\mathbf{0}\rangle = \prod_{j=1}^D \sum_{k_j=0}^{d-1} \sum_{i_j=0}^{n_m-1} \sum_{\substack{j=1,2,\dots,D \\ x_{i_j} \in \alpha_{j,k_j}}} \sum_{|\beta| \leq L_{i_j, k_j}} f_{j,k_j, \beta} \mathbf{x}^\beta |\mathbf{i}\rangle$$

$$\begin{aligned}
&= \prod_{j=1}^D \sum_{k_j=0}^{d-1} \sum_{i_j=0}^{n_m-1} \sum_{\substack{j=1,2,\dots,D \\ x_{i_j} \in \alpha_{j,k_j}}} \sum_{|\beta| \leq L_{j,k_j}} f_{j,k_j,\beta} x_{i_1}^{\beta_1} x_{i_2}^{\beta_2} \cdots x_{i_D}^{\beta_D} |\mathbf{i}\rangle \\
&= \prod_{j=1}^D \sum_{k_j=0}^{d-1} \sum_{i_j=0}^{n_m-1} \sum_{|\beta| \leq L_{j,k_j}} f_{j,k_j,\beta} (i_1 \Delta x)^{\beta_1} (i_2 \Delta x)^{\beta_2} \cdots (i_D \Delta x)^{\beta_D} \sum_{\substack{j=1,2,\dots,D \\ x_{i_j} \in \alpha_{j,k_j}}} |\mathbf{i}\rangle \\
&= \prod_{j=1}^D \sum_{k_j=0}^{d-1} \sum_{i_j=0}^{n_m-1} \sum_{|\beta| \leq L_{j,k_j}} f_{j,k_j,\beta} (i_1 \Delta x)^{\beta_1} (i_2 \Delta x)^{\beta_2} \cdots (i_D \Delta x)^{\beta_D} S_{\chi \prod_{j=1}^D \alpha_{j,k_j}} |\mathbf{0}\rangle \\
&= \prod_{j=1}^D \sum_{k_j=0}^{d-1} \sum_{i_j=0}^{n_m-1} f_{j,k_j}(x_{i_1}, x_{i_2}, \dots, x_{i_D}) S_{\chi \prod_{j=1}^D \alpha_{j,k_j}} |\mathbf{0}\rangle. \tag{74}
\end{aligned}$$

The operator S_f is extracted from this and seen to be

$$S_f = \prod_{j=1}^D \sum_{k_j=0}^{d-1} \sum_{|\beta| \leq L_{j,k_j}} f_{j,k_j,\beta} (\Delta x)^{|\beta|} (D(n))^{|\beta|} S_{\chi \prod_{j=1}^D \alpha_{j,k_j}}, \tag{75}$$

where $D(n)$ was given by (31). We now note that if $D = 1$, then (74) becomes

$$S_f |\mathbf{0}\rangle = \sum_{k_1=0}^{d-1} \sum_{i_1=0}^{n_m-1} f_{1,k_1}(x_{i_1}) S_{\chi \alpha_{1,k_1}} |\mathbf{0}\rangle = \sum_{k=0}^{d-1} \sum_{i=0}^{2^n-1} f_k(x_i) S_{\chi \alpha_k} |\mathbf{0}\rangle, \tag{76}$$

since in this case there is only one dimension so that $f_{1,k}$ can simply be written as f_k , $\alpha_{1,k}$ is α_k and all qubits are assigned to the same dimension. Moreover for one dimension (75) reduces to

$$S_f = \sum_{k_1=0}^{d-1} \sum_{|\beta| \leq L_{1,k_1}} f_{1,k_1,\beta} (\Delta x)^{|\beta|} (D(n))^{|\beta|} S_{\chi \alpha_{1,k_1}} = \sum_{k=0}^{d-1} \sum_{\beta=0}^L f_k(\beta) (\Delta x)^\beta (D(n))^\beta S_{\chi \alpha_k}, \tag{77}$$

where we are now dealing an ordinary Taylor series of L th order in one dimension $f_k(x) = \sum_{\beta=0}^L \frac{f_k^\beta(0)}{\beta!} x^\beta =: \sum_{\beta=0}^L f_k \beta x^\beta$. This means that $S_f |\mathbf{0}\rangle \langle \mathbf{0}| S_f^\dagger$ is the sum of $\mathcal{O}(d^2 n^{2(L+1)})$ unitaries and each Q_i is composed of at most $\mathcal{O}(n^4)$ gates because $S_{\chi \alpha_k}$, $(D(n))^\beta$ and $|\mathbf{0}\rangle \langle \mathbf{0}|$ are the sum of $\mathcal{O}(n)$, $\mathcal{O}(n^\beta)$ and $\mathcal{O}(1)$ unitaries, respectively, each composed of $\mathcal{O}(n)$, $\mathcal{O}(1)$ and $\mathcal{O}(n^2)$ gates, respectively. Equations (75), (76) and (77) are inaccurately given in [12, Eqs. (39), (40) and (B24)].

Let us now return to the error term of the expectation (62) resulting from the Taylor expansion of the function f . We now assume, for simplicity of notation, that all the L 's in each dimension are the same. The Lagrange form of the error term of the L th order multidimensional Taylor series around $\mathbf{a} = \{hk_1, hk_2, \dots, hk_D\}$ is given by

$$R_{L,k}(\mathbf{x}, \mathbf{a}) = \sum_{|\gamma|=L+1} \frac{\partial^\gamma f_k(\mathbf{c})}{\gamma!} (\mathbf{x} - \mathbf{a})^\gamma, \tag{78}$$

where $\mathbf{c} = (c_1, c_2, \dots, c_D)$ is such that $c_i \in [x_i, (k_i + 1)h]$. For every $x_i \in \mathbf{x}$ and every hk_i with same index i we have that $x_i - hk_i = x_i - ak_i \leq h$. Hence we see that $R_{L,k}(\mathbf{x}, \mathbf{a}) = \mathcal{O}(\prod_{i=1}^D h^{L+1}) = \mathcal{O}(h^{D(L+1)})$. Let \mathbb{E}_L be the expectation that would result from approximating f with the L th order Taylor series, then

$$|\mathbb{E} - \mathbb{E}_L| = \left| \sum_{k_1=0}^{d-1} \cdots \sum_{k_D=0}^{d-1} \int_{k_1 h}^{(k_1+1)h} \cdots \int_{k_D h}^{(k_D+1)h} f(\mathbf{x}) p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \right|$$

$$\begin{aligned}
& - \sum_{k_1=0}^{d-1} \cdots \sum_{k_D=0}^{d-1} \int_{k_1 h}^{(k_1+1)h} \cdots \int_{k_D h}^{(k_D+1)h} f_L(\mathbf{x}) p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \Big| \\
& = \left| \sum_{k_1=0}^{d-1} \cdots \sum_{k_D=0}^{d-1} \int_{k_1 h}^{(k_1+1)h} \cdots \int_{k_D h}^{(k_D+1)h} R_{L,k}(\mathbf{x}, \mathbf{a}) p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \right| \\
& \leq \left(\max_{\mathbf{k}} \max_{i=1, \dots, D: hk_i \leq x_i \leq h(k_i+1)} R_{L,k}(\mathbf{x}, \mathbf{a}) \right) \\
& \quad \times \left| \sum_{k_1=0}^{d-1} \cdots \sum_{k_D=0}^{d-1} \int_{k_1 h}^{(k_1+1)h} \cdots \int_{k_D h}^{(k_D+1)h} p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \right| \\
& = \left(\max_{\mathbf{k}} \max_{i=1, \dots, D: hk_i \leq x_i \leq h(k_i+1)} R_{L,k}(\mathbf{x}, \mathbf{a}) \right) \\
& = \mathcal{O}(h^{D(L+1)}), \tag{79}
\end{aligned}$$

where $p_{\mathbf{X}}(\mathbf{x})$ denotes the continuous joint probability distribution function. In order to secure the error below a desired threshold ε , we need to take $d > x_{\max} \varepsilon^{-\frac{1}{D(L+1)}}$.

The final issue we need to resolve is the construction of the operator S_f and its complexity. This construction is explained in [12] for one dimension, and we now give the D dimensional generalization needed for the rest of the bounds on the complexity of the algorithm. For any given $i = 1, 2, \dots, D$ we have that $a_i \in [0, x_{i,\max}]$ and hence there exists an integer k_{a_i} with $0 < k_{a_i} \leq \lfloor n/D \rfloor$ such that $\Delta x 2^{k_{a_i}-1} \leq a_i \leq \Delta x 2^{k_{a_i}}$. The binary expansion of $a_i/\Delta x$ is $a_i/\Delta x = \sum_{j=0}^{k_{a_i}-1} s_{i,j} 2^j$ with $s_{i,j} \in \{0, 1\}$. Next, we define the matrix of elements ℓ

$$\mathbf{L} = \begin{pmatrix} \ell_{1,1} & \ell_{1,2} & \cdots & \ell_{1,k_{a_1}-1} \\ \ell_{2,1} & \ell_{2,2} & \cdots & \ell_{2,k_{a_2}-1} \\ \vdots & \vdots & \ddots & \vdots \\ \ell_{D,1} & \ell_{D,2} & \cdots & \ell_{D,k_{a_D}-1} \end{pmatrix} \tag{80}$$

satisfying $s_{i,\ell} = 1$ in ascending order for each i , and we define the collection of intervals

$$\chi_{\ell_{i,j}}^{a_i} = \left[2^{\ell_{i,k_{a_i}-1}} + \sum_{k=0}^{\ell_{i,j}-1} s_{i,k} 2^k + 1, 2^{\ell_{i,k_{a_i}-1}} + \sum_{k=0}^{\ell_{i,j}} s_{i,k} 2^k \right] \tag{81}$$

for $\ell_{i,j}$ a member of the matrix \mathbf{L} . These intervals allow us to divide a dimension interval $[0, a_i/\Delta x]$ into disjoint intervals as

$$[0, a_i/\Delta x] = [0, 2^{k_{a_i}-1}] \cup \bigcup_{m=1}^{k_{a_i}-1} \chi_{\ell_{i,m}}^{a_i}. \tag{82}$$

The effect of the operator $S_{\chi_{\ell_{i,j}}^{a_i}}$ on the interval $\chi_{\ell_{i,j}}^{a_i}$ is defined as

$$S_{\chi_{\ell_{i,j}}^{a_i}} |0\rangle = 2^{\ell_{i,j}/2} \mathbb{I}^{\otimes k_{a_i}-1} \otimes X \otimes \bigotimes_{k=0}^{\lfloor n/D \rfloor - k_{a_i} - 1 - \ell_{i,j} - 1} \mathbb{X}_{i,s_{\lfloor n/D \rfloor - k_{a_i} - m}} \otimes H^{\otimes \ell_{i,j}} |0\rangle \tag{83}$$

where the operator \mathbb{X} takes two different values according to

$$\mathbb{X}_s = \begin{cases} X, & \text{if } s = 1, \\ \mathbb{I}, & \text{if } s = 0. \end{cases}$$

For the last interval on the right, the action of the operator is defined as

$$S_{\chi_{[0,2^{k_{a_i}}-1]}} |0\rangle = 2^{(k_{a_i}-1)/2} \mathbb{I}^{\otimes \lfloor n/D \rfloor - k_{a_i} + 1} H^{\otimes k_{a_i} - 1} |0\rangle. \quad (84)$$

Thus the interval $S_{\chi_{[0,a_i]}}$ is constructed by summing over $S_{\chi_{\ell_{i,j}}^{a_i}}$ and $S_{\chi_{[0,2^{k_{a_i}}-1]}}$. Lastly, we obtain $S_{\chi_{\alpha_{i,k}}} = S_{\chi_{[0,a_{i,k+1}]}} - S_{\chi_{[0,a_{i,k}]}}$, which is a sum of at most $\mathcal{O}(n)$ unitaries composed of $\mathcal{O}(n)$ gates.

The last part is the bound on the computational complexity in several dimensions. In order to upper bound the error ε coming from $E = \sqrt{\langle \tilde{\psi} | S_f | 0 \rangle \langle 0 | S_f^\dagger | \tilde{\psi} \rangle}$ we shall upper bound the error ε' of each of the terms in (70) and then find the number of measurements and gate complexity needed to achieve this error. Following [12, §A] we assume that $S_f | 0 \rangle \langle 0 | S_f^\dagger$ can be written as a linear combination of N_{U_j} unitaries for each dimension $j = 1, 2, \dots, D$, i.e.

$$S_f | 0 \rangle \langle 0 | S_f^\dagger = \bigotimes_{j=1}^D \sum_{i_j=1}^{N_{U_j}} \beta_{i_j} U_{i_j}, \quad \text{where } U_{i_j} \text{ are unitary operators.} \quad (85)$$

The normalized state is

$$\begin{aligned} |\psi\rangle &= \left(\sum_{j_1=0}^{n_m-1} \sum_{j_2=0}^{n_m-1} \cdots \sum_{j_D=0}^{n_m-1} \mathbb{P}^2[X_1(t) = x_{j_1}, X_2(t) = x_{j_2}, \dots, X_D(t) = x_{j_D}] \right)^{-1/2} |\tilde{\psi}\rangle \\ &= \left(\sum_{j_1=0}^{n_m-1} \sum_{j_2=0}^{n_m-1} \cdots \sum_{j_D=0}^{n_m-1} \mathbb{P}_{j_1, j_2, \dots, j_D}^2 \right)^{-1/2} |\tilde{\psi}\rangle. \end{aligned} \quad (86)$$

The error ε' is defined as the error of the expectation value of each term in a state $|\psi\rangle$. This implies that the estimated expectation value of each term $\tilde{u}_{i_1, i_2, \dots, i_D}$ satisfies

$$\left| \tilde{u}_{i_1, i_2, \dots, i_D} - \langle \psi | \bigotimes_{j=1}^D U_{i_j} | \psi \rangle \right| \leq \varepsilon' \quad (87)$$

The error in the linear combination of expectation values is determined using (87)

$$\begin{aligned} \left| \prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} \beta_{i_j} \tilde{u}_{i_1, i_2, \dots, i_D} - \langle \psi | \bigotimes_{j=1}^D \sum_{i_j=1}^{N_{U_j}} \beta_{i_j} U_{i_j} | \psi \rangle \right| &= \left| \prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} \beta_{i_j} (\tilde{u}_{i_1, i_2, \dots, i_D} - \langle \psi | \bigotimes_{j=1}^D U_{i_j} | \psi \rangle) \right| \\ &\leq \prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} \left| \beta_{i_j} (\tilde{u}_{i_1, i_2, \dots, i_D} - \langle \psi | \bigotimes_{j=1}^D U_{i_j} | \psi \rangle) \right| \\ &\leq \varepsilon' \prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} |\beta_{i_j}|. \end{aligned} \quad (88)$$

If we write \tilde{E} for the estimation of the expectation E , then by the use of (86) and (88) we arrive at

$$\begin{aligned} |\tilde{E} - E| &\leq \left(\sum_{j_1=0}^{n_m-1} \sum_{j_2=0}^{n_m-1} \cdots \sum_{j_D=0}^{n_m-1} \mathbb{P}_{j_1, j_2, \dots, j_D}^2 \right) \frac{\varepsilon'}{\tilde{E} + E} \prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} |\beta_{i_j}| \\ &\sim \left(\sum_{j_1=0}^{n_m-1} \sum_{j_2=0}^{n_m-1} \cdots \sum_{j_D=0}^{n_m-1} \mathbb{P}_{j_1, j_2, \dots, j_D}^2 \right) \frac{\varepsilon'}{2E} \prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} |\beta_{i_j}|. \end{aligned} \quad (89)$$

Therefore, in order to obtain an upper bound for ε in the expectation E , the error ε' must be such that

$$\varepsilon' \leq \frac{2E}{(\sum_{j_1=0}^{n_m-1} \sum_{j_2=0}^{n_m-1} \cdots \sum_{j_D=0}^{n_m-1} \mathbb{P}_{j_1,j_2,\dots,j_D}^2) \prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} |\beta_{i_j}|} \varepsilon =: \gamma \varepsilon. \quad (90)$$

In terms of the Hadamard test circuit above, one requires $\mathcal{O}(1/\varepsilon'^2)$ measurements to bound the error in the expectation value to ε' . Up to the preparation of the state, the depth of the unitaries in the circuit is $\mathcal{O}(1)$. The trade-off with QPE is that the same accuracy in the expectation is $\mathcal{O}(\log 1/\varepsilon')$ but the depth is now $\mathcal{O}(1/\varepsilon')$. With the L th order Taylor approximation, one has that $N_{U_j} = \mathcal{O}(d^2 n^{2L+2})$ for $j = 1, 2, \dots, D$, where d was the number of intervals. Moreover, the total number of measurements is the number of measurements for each term affected by a multiplication by N_{U_j} . Effectively, this yields that the total number of measurements by Hadamard test is $\mathcal{O}((\prod_{j=1}^D d^2 n^{2L+2}) / (\gamma \varepsilon^2))$, whereas using QPE, one would need $\mathcal{O}((\prod_{j=1}^D d^2 n^{2L+2}) \log(1/\gamma \varepsilon))$, but QPE needs $\mathcal{O}(1/\gamma \varepsilon)$ many U gates.

Lastly, we need the bound for γ from (90). Since $\sum_{j_1=0}^{n_m-1} \sum_{j_2=0}^{n_m-1} \cdots \sum_{j_D=0}^{n_m-1} \mathbb{P}_{j_1,j_2,\dots,j_D}^2 \leq 1$, we see that $\gamma \geq 2E / (\prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} |\beta_{i_j}|)$. Next, using (68) we get

$$\prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} |\beta_{i_j}| = 2^D \prod_{j=1}^D \sum_{\ell_j, \ell'_j} |\xi_{\ell_j} \xi_{\ell'_j}^*| = 2^D \prod_{j=1}^D \sum_{\ell_j, \ell'_j} |\xi_{\ell_j}| |\xi_{\ell'_j}|. \quad (91)$$

Moreover, from (31) we see that $(D(n))^{|\beta|} = \mathcal{O}(2^{n|\beta|}) = \mathcal{O}(x_{\max}^{|\beta|})$. For an interval in a given dimension $S_{\chi \alpha_k}$ is a linear combination of operators $S_{\chi_\ell^\alpha}$, therefore $|S_{\chi \alpha_k}| = \mathcal{O}(2^{(k_{a_i}-1)/2}) = \mathcal{O}(x_{\max}^{1/2})$ since k_{a_i} was such that $0 < k_{a_i} \leq \lfloor n/D \rfloor$. This implies that the largest $|\xi_{\ell_j}|$ is as large $[\max_{k_j} \max_{|\beta|} |f_{j,k_j,\beta}| x_{\max}^{|\beta|}] \mathcal{O}(x_{\max}^{1/2})$. Therefore, we get

$$\prod_{j=1}^D \sum_{i_j=1}^{N_{U_j}} |\beta_{i_j}| = \mathcal{O}\left(\prod_{j=1}^D [\max_{k_j} \max_{|\beta|} |f_{j,k_j,\beta}|] [x_{\max}^{|\beta|}]^2 d^2 n^{2L+2} x_{\max}\right), \quad (92)$$

which is the last bound we needed for γ .

The operator $S_f |\mathbf{0}\rangle \langle \mathbf{0}| S_f^\dagger$ will be the sum of $\mathcal{O}(d^{2D} n^{2D(L+1)})$ unitaries and each Q_k will be composed of $\mathcal{O}(n^4)$ gates. The number of gates will grow exponentially with the dimensions and this is a universal problem in many algorithms, such as the use of QAE when computing moments when the arguments of the function are multidimensional. In such cases one needs exponentially many resources to build a multidimensional oracle.

As L grows, $\varepsilon^{-\frac{2}{D(L+1)}}$ from (79) becomes smaller while $d^{2D} n^{D(2L+2)}$ grows larger. This means that the number of unitaries is not monotonic with respect to L and thus there could be an optimal L for the desired accuracy. In [12], the authors point out that the expectation values of those unitaries could be computed in parallel with a second quantum device.

H Future work

As discussed in Section 5, some of the following areas of research might have to be pursued using a combination of real and imaginary time evolutions.

H.1 Stochastic volatility and stochastic interest rates

A key weakness in the Black-Scholes is that the volatility term σ is constant. One can go around this problem by defining two SDEs: geometric Brownian motion for the underlying stock S

$$dS_t = \mu S_t dt + \sigma_t S_t dW_{1,t}, \quad (93)$$

where dW_1 is standard Brownian motion, as well as another SDE for σ

$$d\sigma_t = p(S, \sigma, t)dt + q(S, \sigma, t)dW_{2,t}, \quad (94)$$

where dW_2 is another Brownian motion whose correlation with dW_1 is given ρ , i.e. $dW_1 dW_2 = \rho dt$. Let $V(S, \sigma, t)$ be the value of an option on the underlying S at time t and with stochastic volatility σ . Portfolio replication coupled with Itô's lemma and non-arbitrage arguments yields the partial differential equation

$$\frac{\partial V}{\partial t} + rS \frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + (p - \lambda)q \frac{\partial V}{\partial \sigma} + \frac{1}{2}q^2 \frac{\partial^2 V}{\partial \sigma^2} + \rho\sigma S q \frac{\partial^2 V}{\partial S \partial \sigma} - rV = 0. \quad (95)$$

Here $\lambda(S, \sigma, t)$ is a universal function that is used in portfolio replication to ensure that options are only dependent on the variables S, σ and t . The function λ is called the market price of volatility risk. Comparing (95) with Black-Scholes

$$\frac{\partial V}{\partial t} + rS \frac{\partial V}{\partial S} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} - rV = 0, \quad (96)$$

we see the appearance of the additional linear and second order partial derivatives with respect to σ coming from (94) as well as the presence of the mixed term in S and σ due to the correlation ρ of the two Brownian motions dW_1 and dW_2 . Note that if the volatility were to be constant, then (95) would reduce to (96). Different choices of p and q lead to different named models, e.g.

- Hull and White: $d(\sigma^2) = a(b - \sigma^2)dt + c\sigma^2 dW_2$,
- Square-root model / Heston: $d(\sigma^2) = (a - b\sigma^2)dt + c\sqrt{\sigma^2}dW_2$,
- 3/2 model: $d(\sigma^2) = (a\sigma^2 - b\sigma^4)dt + c\sigma^{3/2}dW_2$,
- Ornstein–Uhlenbeck: $d(\log \sigma^2) = (a - b \log \sigma^2)dt + cdW_2$.

see [36, p. 861]. Similar equations also exist for stochastic interest rates [36].

H.2 American and decision-embedded options

The American option is also an involved one to price. Unlike in the European set-up, the holder of an American option may exercise at any time prior and up to expiration. The details of this set up can be found in [63, §2]. We are dealing with our usual geometric Brownian motion (93) as well as the additional SDE

$$dX_t^q = q_t dS_t + \mu(X_t^q - q_t S_t)dt, \quad X_0^q = X_0, \quad (97)$$

where q_t is the number of shares held at time t . The strategy of q_t is subject to the contractual constraint $q_t \in [\alpha_t, \beta_t]$, where $\alpha_t \leq \beta_t$. Financial arguments lead us to the conditional expectation

$$V^{[\alpha, \beta]}(t, S_t, X_t^q) = \max_{q_u \in [\alpha, \beta]} e^{-r(T-t)} \mathbb{E}[\max(x_T^q, 0) | \mathcal{F}_t], \quad t \in [0, T].$$

These expectations arise naturally in stochastic optimal control and are characterized by the corresponding Hamilton-Jacobi-Bellman equation

$$-rV + \frac{\partial V}{\partial t} + rs \frac{\partial V}{\partial s} + \max_{q \in [\alpha, \beta]} \left[(\mu x + qs(r - u)) \frac{\partial V}{\partial x} + \frac{1}{2} \sigma^2 s^2 \left(\frac{\partial^2 V}{\partial s^2} + 2q \frac{\partial^2 V}{\partial s \partial x} + q^2 \frac{\partial^2 V}{\partial x^2} \right) \right] = 0,$$

with associated boundary condition $V(T, s, x) = \max(x, 0)$. Note the typo in [63, Eq. (2.4)]. To model the American option using (97), one sets $\mu = 0$ and one allows only one switch in q_t , either from 1 to 0 for the American call, or from -1 to 0 for the American put.

H.3 The Poisson and wave equations

The heat equation is not the only candidate for this algorithm. In one of its simplest forms the Poisson equation reads as follows

$$\nabla^2 u(x, y, z) = f(x, y, z),$$

where ∇^2 is the Cartesian Laplacian in three dimensions. Note the similarity with the heat equation, except for the absence of the time derivative. Itô's lemma can be used [40, §4.1.1] to transform the conditional expectation

$$w(\mathbf{x}) = \mathbb{E} \left[\int_0^\infty \exp \left(- \int_0^s c(\mathbf{X}_\tau) d\tau \right) f(\mathbf{X}_s) \mid \mathbf{X}_t = \mathbf{x} \right]$$

arising from the vectorized SDE

$$d\mathbf{X}_t = b(\mathbf{X}_t) dt + \sum_j \sigma_{ij}(\mathbf{X}_t) dW_t^j$$

into the solution of the PDE

$$\sum_{i,j} \frac{1}{2} \Sigma_{ij}(x) \frac{\partial^2 w}{\partial x_i \partial x_j} + \sum_j b_j \frac{\partial w}{\partial x_j} - c(\mathbf{x}) w = f(\mathbf{x}).$$

Here $\Sigma_{ij} = \sum_k \sigma_{ik} \sigma_{kj}$. Note the absence of t in w , as w is now only a function of space \mathbf{x} .

Another venue would be to consider the wave or Klein-Gordon equations (or in general, other type of hyperbolic differential equations). In its simplest form the wave equation is written as

$$\frac{\partial^2 u}{\partial t^2} = c^2 \nabla^2 u(x, y, z), \quad c > 0. \quad (98)$$

The time derivative is now of second order, unlike in the heat equation, which was of first order. In these cases, other types of probability distributions and Brownian motions need to be added, specially by considering jumps and Poisson processes, see [64].

In these situations, different variational principles will be needed to account for the different orders of the time derivative that appears in the PDE. For example, when it comes to the wave equation (98), we will need a variational principle of the type

$$\delta \left\| \frac{d^2}{dt^2} |\tilde{v}(\boldsymbol{\theta}(t))\rangle - \mathcal{O}(t) |\tilde{v}(\boldsymbol{\theta}(t))\rangle \right\| = 0,$$

as opposed to (25). This procedure, along with more generalizations of the time derivative and related differential operators, will be carried out in subsequent research.