PAPER • OPEN ACCESS

Estimating truncation effects of quantum bosonic systems using sampling algorithms

To cite this article: Masanori Hanada *et al* 2023 *Mach. Learn.: Sci. Technol.* **4** 045021

View the article online for updates and enhancements.

MACHINE LEARNING
Science and Technology

PAPER

Estimating truncation effects of quantum bosonic systems using sampling algorithms

Masanori Hanada[1,2,*], Junyu Liu[2,3,4,5,11], Enrico Rinaldi[6,7,8,9] and Masaki Tezuka[10]

1 School of Mathematical Sciences, Queen Mary University of London, Mile End Road, London E1 4NS, United Kingdom
2 qBraid Co., Harper Court 5235, Chicago, IL 60615, United States of America
3 Pritzker School of Molecular Engineering, The University of Chicago, Chicago, IL 60637, United States of America
4 Chicago Quantum Exchange, Chicago, IL 60637, United States of America
5 Kadanoff Center for Theoretical Physics, The University of Chicago, Chicago, IL 60637, United States of America
6 Quantinuum K.K., Otemachi Financial City Grand Cube 3F, 1-9-2 Otemachi, Chiyoda-ku, Tokyo, Japan
7 Physics Department, University of Michigan, Ann Arbor, MI 48109, United States of America
8 Theoretical Quantum Physics Laboratory, RIKEN, Wako, Saitama 351-0198, Japan
9 Interdisciplinary Theoretical and Mathematical Sciences (iTHEMS) Program, RIKEN, Wako, Saitama 351-0198, Japan
10 Department of Physics, Kyoto University, Kyoto 606-8502, Japan
11 SeQure, Chicago, IL 60637, United States of America
* Author to whom any correspondence should be addressed.

E-mail: m.hanada@qmul.ac.uk

Abstract

To simulate bosons on a qubit- or qudit-based quantum computer, one has to regularize the theory by truncating infinite-dimensional local Hilbert spaces to finite dimensions. In the search for practical quantum applications, it is important to know how big the truncation errors can be. In general, it is not easy to estimate errors unless we have a good quantum computer. In this paper, we show that traditional sampling methods on classical devices, specifically Markov Chain Monte Carlo, can address this issue for a rather generic class of bosonic systems with a reasonable amount of computational resources available today. As a demonstration, we apply this idea to the scalar field theory on a two-dimensional lattice, with a size that goes beyond what is achievable using exact diagonalization methods. This method can be used to estimate the resources needed for realistic quantum simulations of bosonic theories, and also, to check the validity of the results of the corresponding quantum simulations.

1. Introduction

The Hilbert space for bosons is infinite-dimensional. To simulate bosons on a qubit- or qudit-based quantum computer, one has to introduce a finite-dimensional approximation of the theory by truncating the Hilbert space [1–8]. Sometimes this truncation is referred to as digitization and it is one of the necessary steps in the construction of efficient quantum algorithms to simulate quantum gauge theories [9–13], and to estimate quantum resources [14–16]. We will use truncation and digitization to mean the same thing in the rest of the manuscript, and this should not be confused with the discretization of the spacetime continuum on the lattice.

Some errors associated with digitization are inevitable, and one needs to know how big they can be. While previous studies on truncated bosonic Hilbert spaces [1–8] have looked at the scaling of the errors with the digitization spacing (or equivalently, the number of qubits), this does not seem to be a straightforward task in general. In fact, when the bosonic systems investigated are too large, one would need to run the digitized theory on a good (noiseless) quantum computer and also efficiently compute the same observable with classical algorithms in order to benchmark the result. So far, quantitative assessment of the digitization errors using classical devices is limited to rather small systems because the only known approaches applicable to generic theories are based on the explicit construction of Hilbert space whose

dimensionality increases exponentially with respect to the system size. A thorough numerical study of digitization errors for a ϕ^4 lattice scalar theory was done in [6], where the authors studied a single-site (one bosonic degree of freedom) and a two-site (two bosonic degrees of freedom) system. However, it was difficult to go to larger lattice sizes, due to the exponential growth of memory resources with the number of lattice sites (bosonic degrees of freedom). In this paper, we will resolve this issue for a rather generic class of bosonic theories and reduce the required resources from exponential to polynomial when estimating expectation values. We point out that this is the only known method able to do so for a rather generic class of bosonic systems in arbitrary dimensions. As a modest demonstration, we will study the digitization effect on a two-dimensional 16-site lattice model (16 bosonic degrees of freedom).

It is widely believed that the truncation effect vanishes exponentially, e.g. the low-lying energy eigenvalues have correction suppressed exponentially with respect to the truncation level [3–5, 17–21]. Although this scaling has been verified for concrete examples, there is no proof applicable to a wide class of theories. Furthermore, there is an implicit assumption, i.e. the wave function decays exponentially fast as $|x|$ becomes large. Note also that, even if the exponential accuracy of the digitization is valid, the precise value of the error for each theory is not immediately clear. However, it is practically impossible to study the digitization effects via the explicit construction of the Hilbert space except for small systems, e.g. lattice field theories with a few local degrees of freedom and less than a dozen lattice sites [5, 6]. It is a frustrating situation that prevents us from quantitative resource estimates for quantum simulations. A related issue is that it is not easy to check the validity of the results of the quantum simulation with specific digitization on NISQ devices: the presence of noise can quickly invalidate the results without a working error-correction strategy. It would be practically useful if we can do some calculations on classical devices to establish robust truncation error quantification which, as a byproduct, offers us some ways to cross-check quantum simulations on NISQ devices[12].

In this paper, we show that a Markov Chain Monte Carlo (MCMC) method can be used to estimate the digitization effects in a rather generic class of bosonic systems. MCMC methods [22] (see [23] for an elementary introduction) are straightforward in the original bosonic theory before digitization. Indeed, one can use the Euclidean path-integral method combined with MCMC to study some features of the quantum systems as long as there is no sign problem. If similar simulations are doable for the digitized theory, one can estimate the digitization effects.

A caveat is that the absence of the sign problem in the Euclidean path integral without digitization does not necessarily guarantee the absence of the sign problem in the digitized theory. In this work, we will show that we can apply MCMC to a wide class of theories without having the sign problem, at least for a certain digitization scheme. As a result, we are able to properly assess the amount of digitization effects even for large systems, well beyond the limits of exact diagonalization techniques.

We consider the generic system of N_{bos} bosons consisting of coordinate variables $\hat{x}_i$ ($i = 1, 2, \cdots, N_{\mathrm{bos}}$) and conjugate momentum variables $\hat{p}_i$ that satisfy the canonical commutation relation[13]

$$\left[\hat{x}_j, \hat{p}_k\right] = i\delta_{jk} \,. \tag{1}$$

We assume that the Hamiltonian is given by

$$\hat{H} = \frac{1}{2}\sum_{i=1}^{N_{\mathrm{bos}}} \hat{p}_i^2 + V(\hat{x}_1, \cdots, \hat{x}_{N_{\mathrm{bos}}}) \,, \tag{2}$$

where $V(x_1, \cdots, x_{N_{\mathrm{bos}}})$ is a real function bounded from below and represents the potential energy as a function of the coordinate variables only. This is a rather generic class and it even includes SU(N) Yang–Mills theories described by using the orbifold lattice formulation [24, 25]. We use the truncation in the coordinate basis defined in section 2. As we will see, this scheme admits the MCMC simulations without sign problem.

This paper is organized as follows. In section 2, we introduce the digitization scheme associated with the coordinate basis. Firstly, the case of a single bosonic variable is explained, and then it is generalized to the case of a generic number of variables. The digitization of $(2+1)$-dimensional scalar quantum field theory (QFT) on a lattice is explained as a concrete example. In section 3, the Monte Carlo technique is introduced and some numerical experiments are conducted. Section 4 is devoted to concluding remarks and discussion of future directions.

---

[12] Note that the MCMC method can be used only for a limited class of quantities. Although such quantities enable us to cross-check the validity of the simulation (e.g. we can check if the correct ground-state wave function is obtained), the MCMC method cannot replace the quantum simulation (e.g. it is impossible to add a small excitation to the ground state and determine the Hamiltonian time evolution).

[13] In our convention, $\hbar = 1$.

2. Truncation scheme

In this section, we specify the truncation scheme.

2.1. Coordinate-basis truncation for the single-boson system

Here, we introduce digitization in the coordinate basis [1, 2, 5], also called the field amplitude basis [6, 8]. Let us start with a review of the single boson field example. Let $\{|x\rangle | x \in \mathbb{R}\}$ be the coordinate basis for this particle that satisfies

$$\hat{x}|x\rangle = x|x\rangle. \tag{3}$$

A simple way to digitize it is to introduce a cutoff to the value of the eigenvalue x as

$$-R \leqslant x \leqslant R, \tag{4}$$

and introduce Λ points,

$$x(n) = -R + n a_{\text{dig}}, \qquad a_{\text{dig}} = \frac{2R}{\Lambda - 1}, \qquad n = 0, 1, \cdots, \Lambda - 1. \tag{5}$$

The digitization parameters Λ, a_{dig}, and R should be sent to infinity, zero, and infinity, respectively, to recover the action of the original operator $\hat{x}$. By using $|n\rangle$ to denote $|x(n)\rangle$, we can write

$$\hat{x} = \sum_{n=0}^{\Lambda-1} x(n)|n\rangle\langle n|. \tag{6}$$

The momentum operator $\hat{p}$ appears in the Hamiltonian only in the form of $\hat{p}^2$. A convenient way of regularizing it is

$$\hat{p}^2 = \frac{1}{a_{\text{dig}}^2}\left\{\sum_{n=0}^{\Lambda-1} 2|n\rangle\langle n| - \sum_{n=0}^{\Lambda-2}|n+1\rangle\langle n| - \sum_{n=0}^{\Lambda-2}|n\rangle\langle n+1|\right\}. \tag{7}$$

The dimension of the Hilbert space is Λ and the truncated Hamiltonian is expressed as a $\Lambda \times \Lambda$ matrix. For each concrete example, we can diagonalize the Hamiltonian up to a rather large value of Λ and confirm that the digitization effects below a fixed energy scale disappear exponentially as $\sim e^{-c\Lambda}$ with some $c > 0$. In terms of the number of qubits q, the truncation level is $\Lambda = 2^q$, and hence the suppression of the digitization effects is expected to be doubly exponential, $\sim e^{-c \cdot 2^q}$.

2.2. Coordinate-basis truncation for multi-boson system

Next, let us consider the more interesting case where we have multiple bosonic degrees of freedom. Suppose there are N_{bos} variables $\vec{x} = (x_1, \cdots, x_{N_{\text{bos}}}) \in \mathbb{R}^{N_{\text{bos}}}$. We introduce N_{bos} integers $\vec{n} = (n_1, \cdots, n_{N_{\text{bos}}}) \in \{0, 1, \cdots, \Lambda - 1\}^{N_{\text{bos}}}$ that are related to $x_i(n_i)$ as

$$x_i(n_i) = -R + n_i a_{\text{dig}}, \qquad a_{\text{dig}} = \frac{2R}{\Lambda - 1}, \qquad n_i = 0, 1, \cdots, \Lambda - 1. \tag{8}$$

Note that we can take different R, Λ, and a_{dig} for each bosonic coordinate x_i; here we use the same values for simplicity. For each $i = 1, 2, \cdots, N_{\text{bos}}$, the momentum operator $\hat{p}_i$ appears in the Hamiltonian only in the form of $\hat{p}_i^2$. This is a natural extension of the single-particle case presented in the previous section. Again we choose a regularization

$$\hat{p}_i^2 = \frac{1}{a_{\text{dig}}^2}\left\{\sum_{n=0}^{\Lambda-1} 2|\vec{n}\rangle\langle\vec{n}| - \sum_{n=0}^{\Lambda-2}|\vec{n}+\hat{i}\rangle\langle\vec{n}| - \sum_{n=0}^{\Lambda-2}|\vec{n}\rangle\langle\vec{n}+\hat{i}|\right\}. \tag{9}$$

The dimension of the Hilbert space is $\Lambda^{N_{\text{bos}}}$ and the truncated Hamiltonian is expressed as a $\Lambda^{N_{\text{bos}}} \times \Lambda^{N_{\text{bos}}}$ matrix. Unless N_{bos} is relatively small, it is difficult to directly determine the energy eigenvalues. Still, it is believed that the digitization effects below a fixed energy scale disappear exponentially as $\sim e^{-c\Lambda}$ with some $c > 0$ [5, 6]. As far as we know, there is no proof for this scaling. (Here, an implicit assumption is that the wave function decays exponentially fast as $|x|$ becomes large.) Note also that, even if the exponential accuracy of the digitization is valid, the precise dependence of the errors on a_{dig} for each theory is not immediately clear. Therefore, it is important to have a method to make a quantitative estimation of the truncation effect that is applicable to a wide class of theories. Our solution to this problem in presented later in section 3.

2.3. Scalar QFT on spatial lattice

As a particularly important example and prototypical QFT, we consider a scalar QFT on a d-dimensional spatial lattice. We consider the square lattice with equal lattice spacing a_{lat} in all d directions. The lattice Hamiltonian is defined by

$$\hat{H}_{\text{lat}} = a_{\text{lat}}\hat{H} = \sum_{\vec{n}_{\text{lat}}} \left(\frac{1}{2}\hat{\pi}_{\vec{n}_{\text{lat}}}^2 + \frac{1}{2}\sum_{\mu=1}^{d}\left(\hat{\phi}_{\vec{n}_{\text{lat}}+\hat{\mu}} - \hat{\phi}_{\vec{n}_{\text{lat}}}\right)^2 + V\left(\hat{\phi}_{\vec{n}_{\text{lat}}}\right)\right). \tag{10}$$

Fields $\hat{\phi}$ and $\hat{\pi}$ are dimensionless, and they correspond to the fields in the continuum theory by $\hat{\phi} = a_{\text{lat}}^{(d-1)/2}\hat{\phi}_{\text{cont.}}$ and $\hat{\pi} = a_{\text{lat}}^{(d+1)/2}\hat{\pi}_{\text{cont.}}$. Parameters such as mass are also made dimensionless, e.g. $m_{\text{lat}} = a_{\text{lat}} \times m$. A vector $\vec{n}_{\text{lat}} \in \mathbb{Z}^d$ labels the lattice sites. This is different from $\vec{n}$ used for the digitization, which we now denote by $\vec{n}_{\text{dig}}$. The canonical commutation relation is imposed, i.e.

$$\left[\hat{\phi}_{\vec{n}_{\text{lat}}}, \hat{\pi}_{\vec{n}_{\text{lat}}'}\right] = i\delta_{\vec{n}_{\text{lat}}, \vec{n}_{\text{lat}}'}. \tag{11}$$

The operators $\hat{\phi}$ and $\hat{\pi}$ are the same as $\hat{x}$ and $\hat{p}$ in the previous sections: here we have used a different notation to more easily connect with the traditional symbols used in the physics community.

$\hat{\mu}$ is the unit vector along the μth dimension of the spatial lattice ($\mu = 1, \cdots, d$). This is different from the unit vector $\hat{i}$ used for the digitization ($i = 1, \cdots, N_{\text{bos}}$). As infrared regularization, we introduce the periodic boundary condition with period L to all directions. Typically, the continuum limit ($a_{\text{lat}} \to 0$) is taken by fixing the physical volume $a_{\text{lat}}L$. The number of bosonic degrees of freedom is $N_{\text{bos}} = L^d$. Hence the dimension of the truncated Hilbert space is $\Lambda^{N_{\text{bos}}} = \Lambda^{L^d}$.

We digitize $\hat{\phi}$ just as before, by introducing R, Λ and a_{dig}. We use the same digitization parameters for all lattice points for sake of simplicity. Note that $a_{\text{lat}} \to 0$ and $a_{\text{dig}} \to 0$ do not necessarily commute: the correct order is $a_{\text{dig}} \to 0$ first, then $a_{\text{lat}} \to 0$.

3. Monte Carlo estimate for truncation effect

In this section, we show that the MCMC methods [22] (see [23] for an elementary introduction) can be used to estimate the digitization effects. Numerical demonstrations will be provided as well. A short review of the MCMC methods is provided in appendix. The crucial point is that, as long as the problem under consideration reduces to an average over non-negative weights, the MCMC methods allow efficient computations.

3.1. Formulation and algorithm

Let us estimate the truncation effect in the coordinate-basis scheme by using Markov Chain Monte Carlo simulation. For N_{bos} variables $\vec{x} = (x_1, \cdots, x_{N_{\text{bos}}})$, we define integers $\vec{n} = (n_1, \cdots, n_{N_{\text{bos}}})$ via the relation (8). For a Hamiltonian defined by (2), we consider the thermal partition function defined by

$$Z(\beta) = \text{Tr}\, e^{-\beta\hat{H}}. \tag{12}$$

Here, β is related to the temperature T by $\beta = \frac{1}{T}$. The trace is over the truncated Hilbert space. We rewrite it as

$$Z(\beta) = \sum_{\vec{n}^{(1)}, \cdots, \vec{n}^{(K)}} \langle\vec{n}^{(1)}|e^{-\Delta\cdot\hat{H}}|\vec{n}^{(2)}\rangle \cdot \langle\vec{n}^{(2)}|e^{-\Delta\cdot\hat{H}}|\vec{n}^{(3)}\rangle \cdots \langle\vec{n}^{(K)}|e^{-\Delta\cdot\hat{H}}|\vec{n}^{(1)}\rangle, \tag{13}$$

where $\beta = \Delta \times K$ has been divided up into K intervals. If Δ is sufficiently small, we can rewrite $\langle\vec{n}^{(j)}|e^{-\Delta\cdot\hat{H}}|\vec{n}^{(j+1)}\rangle$ as

$$\langle\vec{n}^{(j)}|e^{-\Delta\cdot\hat{H}}|\vec{n}^{(j+1)}\rangle \simeq \langle\vec{n}^{(j)}|e^{-\Delta\cdot\sum_i \frac{\hat{p}_i^2}{2}}e^{-\Delta\cdot V(\hat{x})}|\vec{n}^{(j+1)}\rangle$$

$$= \langle\vec{n}^{(j)}|e^{-\Delta\cdot\sum_i \frac{\hat{p}_i^2}{2}}|\vec{n}^{(j+1)}\rangle \cdot e^{-\Delta\cdot V(\vec{n}^{(j+1)})}. \tag{14}$$

To handle $\langle\vec{n}^{(j)}|e^{-\Delta\cdot\sum_i \frac{\hat{p}_i^2}{2}}|\vec{n}^{(j+1)}\rangle$, we can truncate the Fourier expansion of $e^{-\Delta\cdot\sum_i \frac{\hat{p}_i^2}{2}}$. If we keep order-$\Delta$ terms, we obtain

$$\simeq \langle \vec{n}^{(j)}| \left(1 - \Delta \cdot \sum_i \frac{\hat{p}_i^2}{2} \right) |\vec{n}^{(j+1)}\rangle \cdot e^{-\Delta \cdot V\left(\vec{n}^{(j+1)}\right)}$$

$$= \left\{ \left(1 - \frac{N_{\mathrm{bos}}\Delta}{a_{\mathrm{dig}}^2} \right) \delta_{\vec{n}^{(j)},\vec{n}^{(j+1)}} + \frac{\Delta}{2a_{\mathrm{dig}}^2} \sum_{i=1}^{N_{\mathrm{bos}}} \left(\delta_{\vec{n}^{(j)},\vec{n}^{(j+1)}+\hat{i}} + \delta_{\vec{n}^{(j)},\vec{n}^{(j+1)}-\hat{i}} \right) \right\} \cdot e^{-\Delta \cdot V\left(\vec{n}^{(j+1)}\right)} . \tag{15}$$

The crucial point that enables us to use MCMC methods is that the final form in (15) is non-negative for any $\vec{n}^{(j)}$ and $\vec{n}^{(j+1)}$, if $1 - \frac{N_{\mathrm{bos}}\Delta}{a_{\mathrm{dig}}^2} > 0$. In other words, we can write the partition function as a sum of non-negative weights.

Of course, when the digitization is removed and x_i are treated as continuous variables, we can easily rewrite the partition function in the form of the standard Feynman path integral, which is well known to be amenable to MCMC simulations. In some sense, we are solving a simple problem in a complicated manner, to estimate the digitization artifact. On the other hand, we are using a mature classical numerical simulation technique to understand an issue arising in the new growing field of quantum simulations for quantum field theories.

Note that $\frac{\Delta}{a_{\mathrm{dig}}^2}$ has to be small for (15) to be a good approximation. Therefore, as a_{dig} becomes smaller, we need smaller Δ and hence more Trotterization steps. For this reason, the classical simulation cost is sensitive to a_{dig}, but not to Λ. In the demonstrations shown in the later sections, we take Λ and R very large so that the finite-R effect is negligible and we can focus on the finite-a_{dig} effect.

Via the Monte Carlo simulation, expectation values of functions of $\vec{x}$ can be estimated from stochastic samples. We write the expectation value as

$$\langle f(\vec{x}) \rangle_\beta = \frac{1}{Z(\beta)} \sum_{\vec{n}^{(1)},\cdots,\vec{n}^{(K)}} \langle \vec{n}^{(1)}|e^{-\Delta \cdot \hat{H}}|\vec{n}^{(2)}\rangle \cdot \langle \vec{n}^{(2)}|e^{-\Delta \cdot \hat{H}}|\vec{n}^{(3)}\rangle \cdots \langle \vec{n}^{(K)}|e^{-\Delta \cdot \hat{H}}|\vec{n}^{(1)}\rangle f(\vec{x}) . \tag{16}$$

Here, $\vec{x}$ and $\vec{n}$ are related by (8). We use the approximation (15) for the simulation. In the limit of $\Delta \to 0$, the approximation becomes exact, and expectation values at finite temperature can be obtained including the digitization effects. By dialing the temperature, the digitization effects at various energy scales can be studied.

3.1.1. Metropolis algorithm

In the Metropolis algorithm, the chain of configurations $\{\vec{n}^{(1)},\cdots,\vec{n}^{(K)}\}$ is generated in such a way that the probability distribution of configurations is proportional to the right-hand side of (13), where the approximation (15) is understood. More explicitly, the probability distribution is proportional to $e^{-S\left(\vec{n}^{(1)},\cdots,\vec{n}^{(K)};\Delta\right)}$, where

$$\exp\left(-S\left(\vec{n}^{(1)},\cdots,\vec{n}^{(K)};\Delta \right) \right)$$
$$\equiv \prod_{j=1}^K \left\{ \left(1 - \frac{N_{\mathrm{bos}}\Delta}{a_{\mathrm{dig}}^2} \right) \delta_{\vec{n}^{(j)},\vec{n}^{(j+1)}} + \frac{\Delta}{2a_{\mathrm{dig}}^2} \sum_{i=1}^{N_{\mathrm{bos}}} \left(\delta_{\vec{n}^{(j)},\vec{n}^{(j+1)}+\hat{i}} + \delta_{\vec{n}^{(j)},\vec{n}^{(j+1)}-\hat{i}} \right) \right\} \cdot e^{-\Delta \cdot V\left(\vec{n}^{(j+1)}\right)} . \tag{17}$$

Note that

$$\lim_{K \to \infty} \sum_{\vec{n}^{(1)},\cdots,\vec{n}^{(K)}} \exp\left(-S\left(\vec{n}^{(1)},\cdots,\vec{n}^{(K)};\Delta \right) \right) \to Z(\beta) \qquad (\beta = K\Delta : \text{fixed}) . \tag{18}$$

In the following, we describe a naive update move for the Metropolis algorithm. The algorithm will start from an initial point, or configuration of all variables. For the initial configuration, we take all $\vec{n}^{(j)}$ to be the same and between 0 and $\Lambda - 1$. (Practically, we should take $n \sim \frac{\Lambda}{2}, x \sim 0$.) We perform the following procedure to $j = 1,2,\cdots,K = \frac{\beta}{\Delta}$ and $i = 1,2,\cdots,N_{\mathrm{bos}}$:

1. **Proposal**: as a candidate for the new value of $n_i^{(j)}$, $n' \equiv n_i^{(j)} \pm 1$ (equivalently, $\vec{n}' = \vec{n}^{(j)} \pm \hat{i}$) is proposed with probability $\frac{1}{2}$ for each.
2. The candidate n' is automatically rejected (i.e. $n_i^{(j)}$ remains unchanged) unless the following three conditions are satisfied:

$$0 \leqslant n' \leqslant \Lambda - 1 , \tag{19}$$

$$|\vec{n}' - \vec{n}^{(j+1)}| = 0 \text{ or } 1, \tag{20}$$

$$|\vec{n}' - \vec{n}^{(j-1)}| = 0 \text{ or } 1. \tag{21}$$

3. **Metropolis test**: the candidate n' is accepted (i.e. $n_i^{(j)}$ becomes n') with the probability $\min\left(1, e^{-\delta S}\right)$, where δS is the increment of the action. Otherwise the candidate n' is rejected (i.e. $n_i^{(j)}$ remains unchanged).

This is 'one sweep'. We repeat many sweeps and collect successive configurations along the Markov Chain. The above naive approach is not efficient because $\vec{x}$ is changed only locally (i.e. only one time slice at each step), while $\vec{x}$ should slowly change along the imaginary-time circle following dominant configurations.

In classical MCMC simulations of spin systems, local update rules for the Metropolis algorithm are known to perform poorly, in particular in regions of metastability or close to phase transitions. Inspired by cluster algorithms [26], we improve the algorithm by allowing the simultaneous update of a cluster as follows:

1. Choose $1 \leqslant B \leqslant B_{\max}$ randomly, where B labels the size of the 'clusters' (or blocks)
2. **Proposal**: as a candidate for the next configuration, we vary $n_i^{(j)}, n_i^{(j+1)}, \cdots, n_i^{(j+B)}$ simultaneously. For $l = j, j+1, \cdots, j+B$, $n'^{(l)} \equiv n_i^{(l)} \pm 1$ (equivalently, $\vec{n}'^{(l)} = \vec{n}^{(l)} \pm \hat{i}$) are proposed with probability $\frac{1}{2}$ for each. Note that we use the same sign for all l's.
3. The candidate n' is automatically rejected unless the following three conditions are satisfied for all $l = j, \cdots, j+B$:

$$0 \leqslant n' \leqslant \Lambda - 1, \tag{22}$$

$$\left|\vec{n}'^{(j+B)} - \vec{n}^{(j+B+1)}\right| = 0 \text{ or } 1, \tag{23}$$

$$\left|\vec{n}'^{(j)} - \vec{n}^{(j-1)}\right| = 0 \text{ or } 1. \tag{24}$$

4. **Metropolis test**: the candidate n' is accepted with the probability $\min\left(1, e^{-\delta S}\right)$, where δS is the increment of the action. Otherwise, the candidate n' is rejected.

$B_{\max}$ can be any number between 1 and K. The optimal value depends on the detail of the Hamiltonian and digitization parameters and in principle it can be found via a standard analysis of the autocorrelation time (see e.g. [23]). As an example, one would first try the algorithm with $B_{\max} = 1$ and measure the autocorrelation time of the observable of interest. Then, one would increase $B_{\max}$ by 1 unit and measure again the autocorrelation time. For some larger value of $B_{\max}$ the autocorrelation time will start decreasing because the update sweeps become more efficient at finding new configurations. For the simulations reported in this paper, we set $B_{\max} = \frac{K}{2}$, except for figure 2 in which $B_{\max} = 1$ is used for comparison. We chose $B = 1, 2, \cdots, B_{\max}$ with equal probability.

3.1.2. Simulation cost
Suppose that R is sufficiently large so that the support of the wave function is mostly contained in the truncated Hilbert space. As long as this loose condition is satisfied, the simulation cost is not sensitive to R. On the other hand, the cost is sensitive to a_{dig}, because it appears in the expansion parameter $\frac{\Delta}{a_{\mathrm{dig}}^2}$ in (15). To keep the expansion parameter small, we need to scale the number of Trotter steps K proportionally to a_{dig}^{-2}. Therefore, the number of variables in the Monte Carlo simulation scales as $N_{\mathrm{bos}} K \sim \frac{N_{\mathrm{bos}}}{a_{\mathrm{dig}}^2} \sim 2^{2n_q} N_{\mathrm{bos}}$. This can be considered the scaling of the memory size, because we need this amount to store all the variables. Note that this memory scaling is much better than $2^{N_{\mathrm{bos}} n_q}$, which is required for the direct computation of the eigenvalues with exact diagonalization methods. For practical applications, n_q does not increase with N_{bos} (each bosonic degree of freedom is represented with the same number of qubits), and hence the necessary memory size increase only linearly with the system size. In addition to the memory scaling, the computational cost for one sweep scales polynomially with N_{bos} and a_{dig}.

Strictly speaking, it is difficult to establish the exponential suppression of the truncation effect unless a clear exponential behavior sets in at a small truncation level because exponentially large statistics and hence exponentially large computational costs are needed to have exponentially small statistical error[14]. However, to establish that the error decays faster than a certain power, only a polynomially large cost is needed.
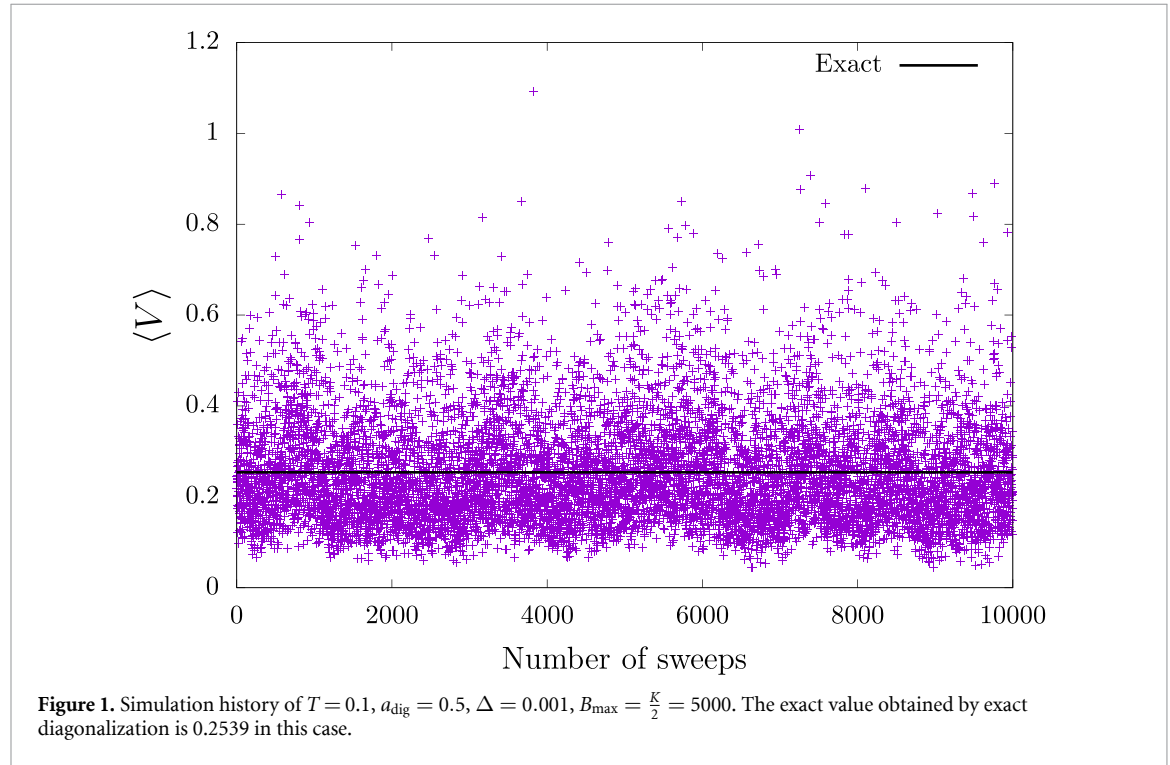
3.2. Single boson ($N_{\mathrm{bos}} = 1$)
As a sanity check of the method, let us consider the case of a single boson. In this case, the exact diagonalization method can be used to determine the low-energy spectrum rather accurately even for large

---

[14] In the Monte Carlo simulation, the statistical error scales as (number of configurations)$^{-1/2}$.

Table 1. $\langle V \rangle_{T=0.1}$ from Monte Carlo (MC) and exact diagonalization (Exact) for $V(x) = \frac{1}{4}x^4 + \frac{m^2}{2}x^2$, $m^2 = \pm 1.0$, $T = 0.1$, various choices of a_{dig}, with the infrared cutoff $R = 1000a_{\text{dig}}$ ($\Lambda = 2001$). For MC, $\Delta = 0.001$ was used and 10^4 sweeps were performed.

a_{dig}	$m^2 = 1.0$, MC	$m^2 = 1.0$, Exact	$m^2 = -1.0$, MC	$m^2 = -1.0$, Exact
0.3	0.2626(26)	0.2618	$-0.06326(68)$	-0.06354
0.5	0.2533(53)	0.2539	$-0.0664(27)$	-0.06633
0.7	0.2482(70)	0.2414	$-0.0717(18)$	-0.07024

Figure 1. Simulation history of $T = 0.1$, $a_{\text{dig}} = 0.5$, $\Delta = 0.001$, $B_{\max} = \frac{K}{2} = 5000$. The exact value obtained by exact diagonalization is 0.2539 in this case.

truncation level Λ. Therefore, we can confirm that the values obtained from the Monte Carlo simulations described in the previous sections are correct.

We considered the Hamiltonian with the following quartic interaction:

$$V(\hat{x}) = \frac{\lambda}{4}\hat{x}^4 + \frac{m^2}{2}\hat{x}^2. \tag{25}$$

Note that we can consider negative m^2 as well. We studied $\lambda = 1.0$, $m^2 = \pm 1.0$ and $a_{\text{dig}} = 0.3, 0.5, 0.7$. The step size Δ is set to 0.001 so that $\frac{\Delta}{a_{\text{dig}}^2}$ is small and the error associated with the truncation of the Taylor expansion of $e^{-\Delta \cdot \frac{\hat{p}^2}{2}}$ is suppressed. To focus on the errors coming from a_{dig} and not from R, we took R and Λ to be very large (specifically, $\Lambda = 2001$ and $R = 1000a_{\text{dig}}$).
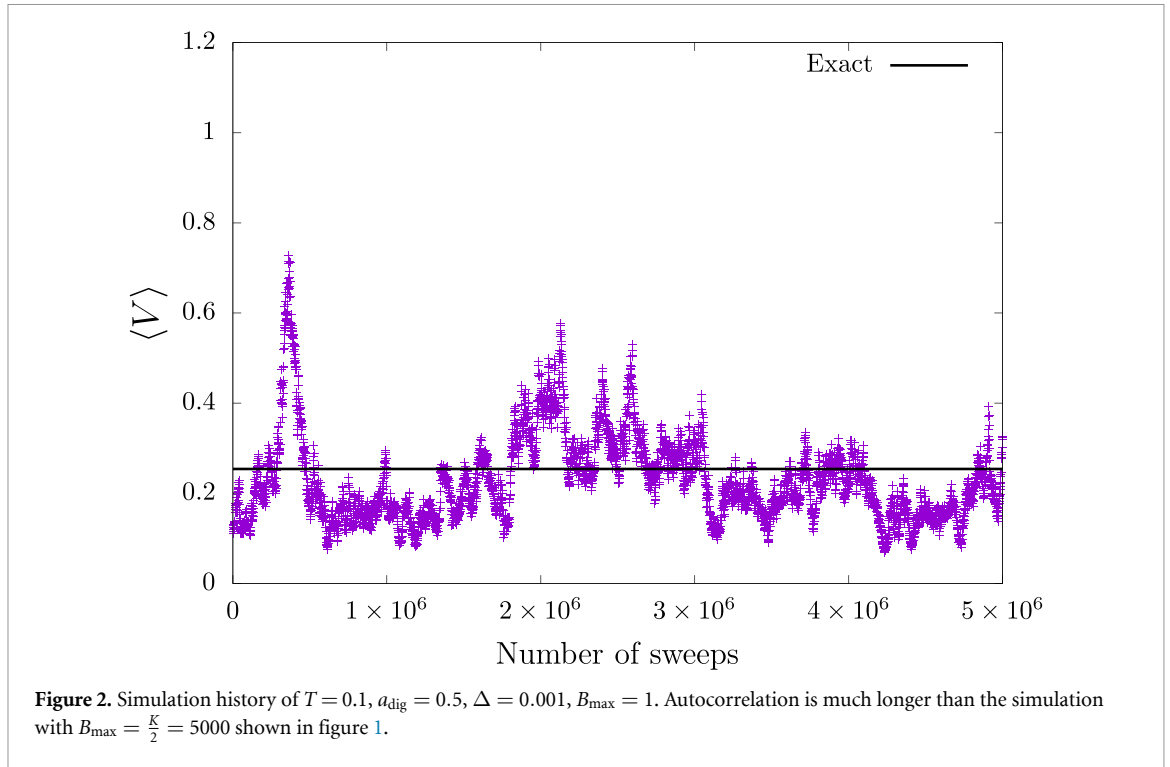
The results are summarized in table 1. 'Exact' values are obtained by exactly diagonalizing the corresponding Hamiltonian in the truncated coordinate basis, with a sufficiently large value of Λ such that the first four nonzero digits are obtained precisely. Both methods give consistent results by taking into account the stochastic (statistical) error from the MCMC simulation. The simulation history for $a_{\text{dig}} = 0.5$, $m^2 = +1.0$ is shown in figure 1. For this plot, we set the maximum cluster size that is updated simultaneously to be $B_{\max} = \frac{K}{2}$. The advantage of updating a large cluster simultaneously can be understood by comparing this plot with figure 2 corresponding to a simulation with $B_{\max} = 1$.

3.3. Scalar QFT ($N_{\text{bos}} = L^d$)

Let us consider the scalar QFT on a square d-dimensional lattice of L sites in each direction. The number of bosons is $N_{\text{bos}} = L^d$, which quickly grows and makes it impossible to construct the Hilbert space explicitly on classical devices.

To demonstrate the validity of the method, we consider the free theory with the potential:

$$V = \frac{m_{\text{lat}}^2}{2}\sum_{\vec{n}}\hat{\phi}_{\vec{n}}^2, \tag{26}$$

Figure 2. Simulation history of $T = 0.1$, $a_{\mathrm{dig}} = 0.5$, $\Delta = 0.001$, $B_{\mathrm{max}} = 1$. Autocorrelation is much longer than the simulation with $B_{\mathrm{max}} = \frac{K}{2} = 5000$ shown in figure 1.

where $m_{\mathrm{lat}} = a_{\mathrm{lat}} m$ in the Hamiltonian defined by (10). We focus on quantities that can be analytically computed for the case of the infinite-dimensional local Hilbert space (the full QFT without digitization effects) and we reproduce those results by taking the limit of $a_{\mathrm{dig}} \to 0$ in our Monte Carlo simulations. (We use this particular setup because the analysis is simpler and the essence of the MCMC approach can be conveyed efficiently. We will comment on the cases without analytic results later in this section.)

The Fourier transform on a lattice is defined by

$$\hat{\tilde{\phi}}_{\vec{q}} = \frac{1}{\sqrt{L^d}} \sum_{\vec{n}} e^{-i\vec{q}\cdot\vec{n}} \hat{\phi}_{\vec{n}}, \qquad \hat{\phi}_{\vec{n}} = \frac{1}{\sqrt{L^d}} \sum_{\vec{q}} e^{i\vec{q}\cdot\vec{n}} \hat{\tilde{\phi}}_{\vec{q}},$$

$$\hat{\tilde{\pi}}_{\vec{q}} = \frac{1}{\sqrt{L^d}} \sum_{\vec{n}} e^{-i\vec{q}\cdot\vec{n}} \hat{\pi}_{\vec{n}}, \qquad \hat{\pi}_{\vec{n}} = \frac{1}{\sqrt{L^d}} \sum_{\vec{q}} e^{i\vec{q}\cdot\vec{n}} \hat{\tilde{\pi}}_{\vec{q}}, \tag{27}$$

where $\vec{q} = (q_1, \cdots, q_d)$, $q_j = \frac{2\pi}{L} \ell_j$, $\ell_j = 1, 2, \cdots, L$. We can write the free Hamiltonian in terms of $\hat{\tilde{\phi}}_{\vec{q}}$ and $\hat{\tilde{\pi}}_{\vec{q}}$ as

$$
\begin{aligned}
\hat{H}_{\mathrm{lat, free}} &= \sum_{\vec{n}} \left(\frac{1}{2} \hat{\pi}_{\vec{n}}^2 + \frac{1}{2} \sum_{\mu=1}^{d} \left(\hat{\phi}_{\vec{n}+\hat{\mu}} - \hat{\phi}_{\vec{n}} \right)^2 + \frac{m_{\mathrm{lat}}^2}{2} \hat{\phi}_{\vec{n}}^2 \right) \\
&= \sum_{\vec{q}} \left(\frac{1}{2} \hat{\tilde{\pi}}_{\vec{q}} \hat{\tilde{\pi}}_{-\vec{q}} + \left(2 \sum_{\mu=1}^{d} \sin^2 \left(\frac{q_\mu}{2} \right) + \frac{m_{\mathrm{lat}}^2}{2} \right) \hat{\tilde{\phi}}_{\vec{q}} \hat{\tilde{\phi}}_{-\vec{q}} \right) \\
&= \sum_{\vec{q}} \left(\frac{1}{2} \hat{\tilde{\pi}}_{\vec{q}} \hat{\tilde{\pi}}_{-\vec{q}} + \frac{\omega_{\mathrm{lat}, \vec{q}}^2}{2} \hat{\tilde{\phi}}_{\vec{q}} \hat{\tilde{\phi}}_{-\vec{q}} \right),
\end{aligned}
\tag{28}
$$

where

$$\omega_{\mathrm{lat}, \vec{q}}^2 = m_{\mathrm{lat}}^2 + 4 \sum_{\mu=1}^{d} \sin^2 \left(\frac{q_\mu}{2} \right). \tag{29}$$

Each mode contributes $\frac{\omega_{\vec{q}}}{2}$ to the ground-state energy. The zero-point fluctuation of each mode is

$$\left\langle \hat{\tilde{\phi}}_{\vec{q}} \hat{\tilde{\phi}}_{-\vec{q}} \right\rangle = \frac{1}{2\omega_{\mathrm{lat}, \vec{q}}}. \tag{30}$$

Table 2. $\left\langle \hat{\tilde{\phi}}_{\vec{q}} \hat{\tilde{\phi}}_{-\vec{q}} \right\rangle$ for $\vec{q} = (q_x, q_y) = (0,0)$ and (π, π) on 2d 4×4 lattice, $a_{\text{lat}} = 1$, $m^2 = 1$, $\lambda = 0$, and $T = 1$ via MCMC. $R = a_{\text{dig}} \times 1000$, $\Lambda = 2001$. We chose Δ such that $\frac{\Delta}{2a_{\text{dig}}^2} \leqslant 0.01$. The exact value without digitization ($\Delta \to 0$ followed by $a_{\text{dig}} \to 0$) is $\frac{1}{2\tanh(0.5)} \simeq 1.081977$ and $\frac{1}{6\tanh(1.5)} \simeq 0.184131$. Simulations are conducted by using multiple (n_{stream}) streams with different random seeds. Each stream has $n_{\text{step}} \geqslant 3.93 \times 10^5$ steps. The auto-correlation length for each $\vec{q}$ is estimated by computing the integrated auto-correlation time for each stream, and for the largest length $d_{\vec{q}}$, the initial $10 d_{\vec{q}}$ steps are discarded as a burn-in period, and the following steps are split into $10 d_{\vec{q}}$-step samples. $d_{\vec{q}} < 10^3$ for $a \leqslant 0.8$, and for $a = 0.9$ and 1.0, stream sizes are greater than $2200 d_{\vec{q}}$. The number shown in the parenthesis indicates the unbiased estimation of the standard deviation from the averages of at least ten streams.

a_{dig}	Δ	$K = \beta/\Delta$	$\vec{q} = (0,0)$	$\vec{q} = (\pi, \pi)$	$d_{(0,0)}$	$d_{(\pi,\pi)}$	n_{stream}	n_{step}
0.20	0.0008	1250	1.0778(20)	0.18288(8)	37	4	40	3.93×10^5
0.25	0.00125	800	1.0800(13)	0.18230(5)	38	4	39	1.1×10^6
0.30	0.0015	667	1.0793(11)	0.18120(4)	37	4	22	2.57×10^6
0.40	0.002	500	1.0751(12)	0.17819(4)	41	4	19	2.59×10^6
0.50	0.002	500	1.0709(10)	0.17611(3)	59	5	10	1×10^7
0.60	0.005	200	1.0645(15)	0.16961(4)	130	7	10	1×10^7
0.70	0.005	200	1.0579(26)	0.16156(6)	369	15	10	1×10^7
0.80	0.005	200	1.0208(39)	0.15214(10)	968	55	10	1.1×10^7
0.90	0.005	200	0.9039(55)	0.14167(17)	2174	148	10	1.1×10^7
1.00	0.005	200	0.7038(70)	0.12792(25)	4491	319	10	1×10^7

This exact value should be obtained in the limits $\Delta \to 0$, $R \to \infty$, $a_{\text{dig}} \to 0$ and $T \to 0$. Note that we need to take both the digitization spacing to zero, and the energy scale to zero. On the other hand, at finite temperatures we have:

$$\left\langle \hat{\tilde{\phi}}_{\vec{q}} \hat{\tilde{\phi}}_{-\vec{q}} \right\rangle = \frac{e^{\beta\omega_{\vec{q}}/2} + e^{-\beta\omega_{\vec{q}}/2}}{e^{\beta\omega_{\vec{q}}/2} - e^{-\beta\omega_{\vec{q}}/2}} \times \frac{1}{2\omega_{\text{lat},\vec{q}}} = \frac{1}{2\omega_{\text{lat},\vec{q}} \tanh\left(\beta\omega_{\vec{q}}/2\right)} . \tag{31}$$

To reproduce this result, we only take the limits $\Delta \to 0$, $R \to \infty$ and $a_{\text{dig}} \to 0$. We use this relation for the numerical demonstrations that follow. Note that the digitization and Fourier transform do not commute in general. Therefore, even for the free theory, the estimation of the digitization effect is not trivial.

The ground-state wave function for each mode $\tilde{\phi}_{\vec{q}}$ is Gaussian with the width $\frac{1}{\sqrt{\omega_{\text{lat},\vec{q}}}}$. For the higher-momentum modes, the widths are narrower, and smaller a_{dig} will be needed for better approximation.
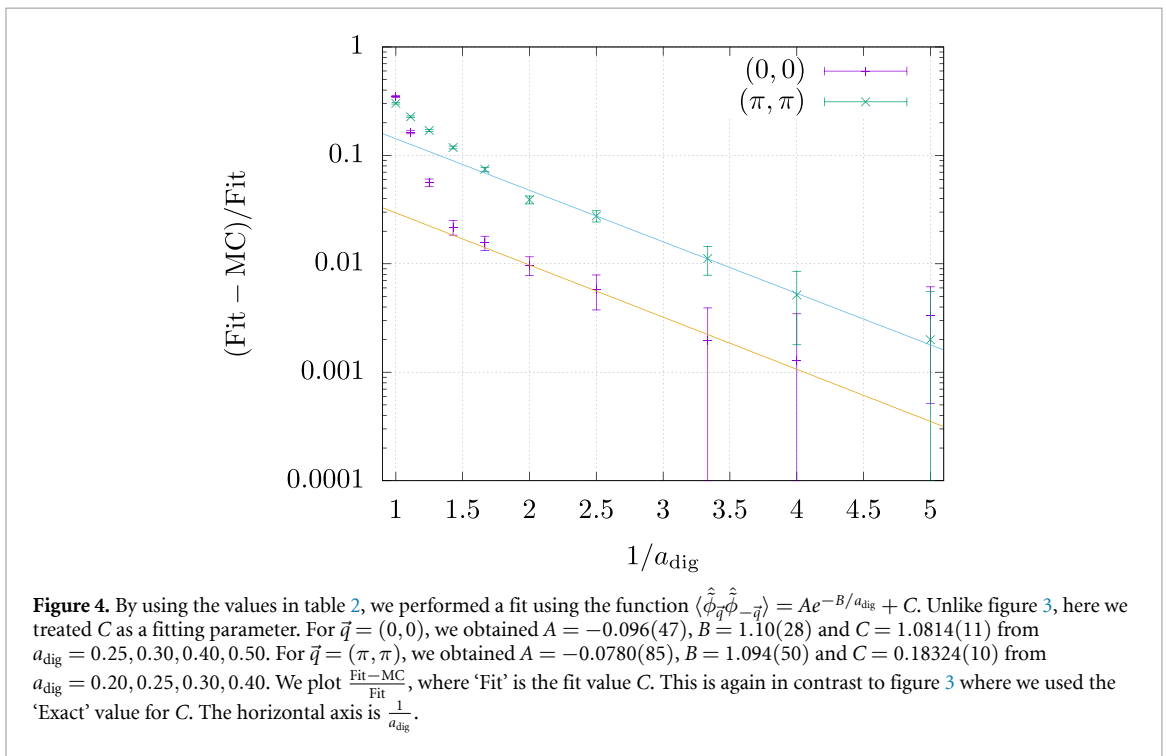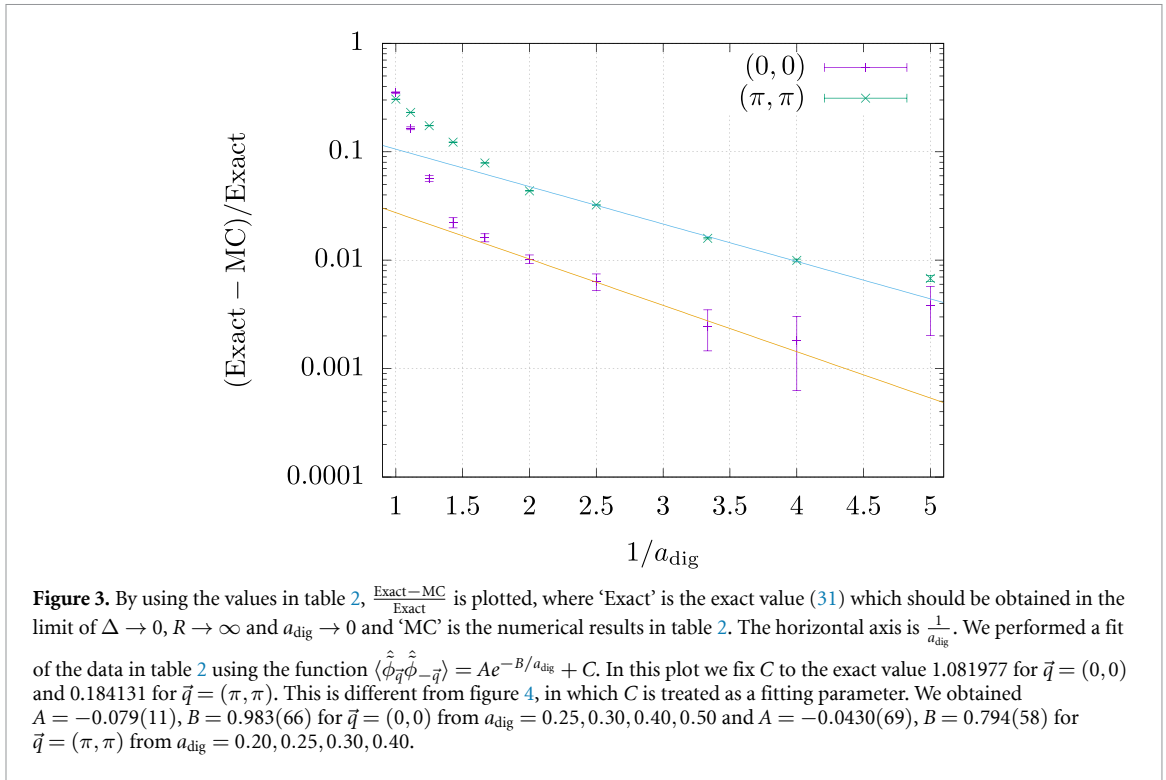
For numerical demonstration, we study a two-dimensional 4×4 lattice. The number of bosons is $N_{\text{bos}} = 4 \times 4 = 16$ and the dimension of the Hilbert space $\Lambda^{N_{\text{bos}}}$ increases so quickly with Λ that numerical analysis on classical devices is practically impossible if the Hilbert space is constructed explicitly.

We choose the parameters to be $a_{\text{lat}} = 1$ and $m^2 = 1$. For $\vec{q} = (q_x, q_y) = (0,0)$ and (π, π), $\frac{1}{2\omega_{\text{lat},\vec{q}}}$ is $\frac{1}{2}$ and $\frac{1}{6}$, respectively. Combined with (31), the values shift to $\frac{1}{2\tanh(0.5)} \simeq 1.0819$ and $\frac{1}{6\tanh(1.5)} \simeq 0.1841$, respectively, at $T = 1$. We estimate the truncation effects on these values.

We took $\frac{\Delta}{2a_{\text{dig}}^2}$ to be 0.01 at maximum. Therefore, we expect the error associated with Trotterization is at most a few percent. The values we obtained are shown in table 2. In figure 3, we plot $\frac{\text{Exact} - \text{MC}}{\text{Exact}}$, where 'Exact' is the exact value (31) which should be obtained in the limit of $\Delta \to 0$, $R \to \infty$ and $a_{\text{dig}} \to 0$, and 'MC' is the numerical result in table 2. The horizontal axis is $\frac{1}{a_{\text{dig}}}$. We can see that the error decreases to $\frac{\text{Exact} - \text{MC}}{\text{Exact}} \sim 0.01$, which is more or less the expected value of the Trotterization effect.

Very roughly, the width of the distribution of $\tilde{\phi}_{\vec{q}}$ is estimated by $\sqrt{\langle \hat{\tilde{\phi}}_{\vec{q}} \hat{\tilde{\phi}}_{-\vec{q}} \rangle}$, which is $\sqrt{1.0819} \simeq 1.04$ for $\vec{q} = (q_x, q_y) = (0,0)$ and $\sqrt{0.1841} \simeq 0.43$ for $\vec{q} = (q_x, q_y) = (\pi, \pi)$. A natural expectation is that the error associated with the digitization disappears exponentially fast once a_{dig} becomes smaller than these values. Qualitatively, we can see such an exponential decrease in figure 3.

For the example discussed above, we knew the analytic results in the limit of $a_{\text{dig}} \to 0$. Even without knowing the analytic result, the digitization error analysis is straightforward, because our method works for any R and a_{dig}. For simplicity, let us assume that $R = \infty$ and focus on the finite-a_{dig} effect, as above. For various values of a_{dig}, we can calculate $\langle \mathcal{O} \rangle$, where $\mathcal{O}$ is observable under consideration, say ϕ^2. Then, we can determine the a_{dig}-dependence by fitting the numerical results at finite a_{dig} values. If the analytic result at $a_{\text{dig}} = 0$ is known, such a fit is easier (has less free parameters), but the fit can be conducted even without knowing the value at $a_{\text{dig}} = 0$. In figure 4 we show how to perform a fit that includes the $a_{\text{dig}} = 0$ value of the observable as a free parameter. We compare the fitted result with the known $a_{\text{dig}} = 0$ result and find that they are statistically compatible. Alternatively, even if an analytical result is not available, one can determine the value at $a_{\text{dig}} = 0$ (without any digitization of the degrees of freedom) by performing the standard Euclidean path integral computation via MCMC.

Figure 3. By using the values in table 2, $\frac{\text{Exact}-\text{MC}}{\text{Exact}}$ is plotted, where 'Exact' is the exact value (31) which should be obtained in the limit of $\Delta \to 0$, $R \to \infty$ and $a_{\text{dig}} \to 0$ and 'MC' is the numerical results in table 2. The horizontal axis is $\frac{1}{a_{\text{dig}}}$. We performed a fit of the data in table 2 using the function $\langle \hat{\phi}_{\vec{q}} \hat{\phi}_{-\vec{q}} \rangle = Ae^{-B/a_{\text{dig}}} + C$. In this plot we fix C to the exact value 1.081977 for $\vec{q} = (0,0)$ and 0.184131 for $\vec{q} = (\pi, \pi)$. This is different from figure 4, in which C is treated as a fitting parameter. We obtained $A = -0.079(11)$, $B = 0.983(66)$ for $\vec{q} = (0,0)$ from $a_{\text{dig}} = 0.25, 0.30, 0.40, 0.50$ and $A = -0.0430(69)$, $B = 0.794(58)$ for $\vec{q} = (\pi, \pi)$ from $a_{\text{dig}} = 0.20, 0.25, 0.30, 0.40$.

Figure 4. By using the values in table 2, we performed a fit using the function $\langle \hat{\phi}_{\vec{q}} \hat{\phi}_{-\vec{q}} \rangle = Ae^{-B/a_{\text{dig}}} + C$. Unlike figure 3, here we treated C as a fitting parameter. For $\vec{q} = (0,0)$, we obtained $A = -0.096(47)$, $B = 1.10(28)$ and $C = 1.0814(11)$ from $a_{\text{dig}} = 0.25, 0.30, 0.40, 0.50$. For $\vec{q} = (\pi, \pi)$, we obtained $A = -0.0780(85)$, $B = 1.094(50)$ and $C = 0.18324(10)$ from $a_{\text{dig}} = 0.20, 0.25, 0.30, 0.40$. We plot $\frac{\text{Fit}-\text{MC}}{\text{Fit}}$, where 'Fit' is the fit value C. This is again in contrast to figure 3 where we used the 'Exact' value for C. The horizontal axis is $\frac{1}{a_{\text{dig}}}$.

4. Conclusion and discussion

In this paper, we introduced an MCMC-based technique to determine the digitization effects in a class of bosonic systems, which have the Hamiltonian of the form (2), in the coordinate-basis truncation scheme. This technique enables us to study the expectation values of various operators at finite temperature including their digitization effects. By dialing the temperature, various energy scales can be studied. As a prototypical QFT application, we studied the $(2+1)$-dimensional scalar QFT regularized on a lattice. To check its convergence to the right answer, we studied the weak-coupling limit. The inclusion of the interactions is straightforward, as we have demonstrated in the case of a single-boson system. Our method can be used to

estimate the resources needed for realistic quantum simulations, and also, to check the validity of the results of the quantum simulations. As a specific example relevant for the NISQ era, we can consider a variational algorithms to determine low-energy quantum states of a multi-dimensional lattice system. By using the MCMC method, one determines the ground-state energy of the digitized theory, including the scaling of the digitization effects. Whether this value can be reproduced is a good test of variational algorithms on NISQ devices when no other classical methods can be used due to the exponentially increasing requirements with the lattice size. Once we could find a quantum algorithm and quantum device that can pass this benchmark test, then we can use it to study more complicated observables which cannot be accessed by the MCMC method, such as excited-state energies or time-dependent correlation functions.

In this paper, we took the 'infrared'-cutoff parameter R very large and focused on the effects due to finite a_{dig}. In near-term quantum simulations the finite-R effects can also be relevant. It is straightforward to study small values of R and estimate the finite-R effects, by using the same simulation technique we have proposed.

To apply the method introduced in this paper, each bosonic variable must take values in $\mathbb{R}$. An interesting class of theories of this kind is Hermitian multi-matrix models, which are important in the context of the holographic duality. A nontrivial issue for those models is how to reconcile the digitization of the gauge field with the existence of the gauge symmetry. However, the study of the corresponding ungauged models [27–29], which are also relevant for the duality, is straightforward. It is not necessarily a problematic issue if one considers a quantum simulation on the extended Hilbert space, which is a rather common approach. For such an approach to work, one has to make sure that the low-energy states of the ungauged model in the extended Hilbert space are correctly described. Lattice gauge theories form another interesting class. Although unitary variables which appear in the Kogut–Susskind formulation [30] of lattice gauge theory cannot be studied by using the formulation given in this paper, it is straightforward to study the orbifold lattice [24, 25].

Whether it is possible to study other digitization schemes by Monte Carlo methods is an interesting question. For SU(N) gauge theories, digitization schemes based on discrete subgroups have been proposed [11–13, 31]. The effects of such schemes can be studied by using the standard lattice simulations [14, 16] and in principle, it is possible to take the continuum limit along the time directions both for gauged and ungauged theories. While such digitization schemes lack systematic un-digitization limits, they might be sufficiently good tools in the NISQ era. Another interesting approach that can be studied by Monte Carlo is the use of non-commutative geometry. For example, [7, 32] studied the truncation of the target space of the O(3) nonlinear sigma model by fuzzy sphere.

Another potentially useful direction is the use of variational Monte Carlo methods, specifically with the neural quantum states. Such an approach, which was confirmed to be valid for some theories before digitization [18, 33–35], may work even with digitization, and the introduction of fermions is straightforward at least conceptually. The introduction of the singlet constraint is also straightforward, for example by adding the square of the generators of gauge symmetry to the action as a penalty term. The MCMC technique introduced in this paper could be used for cross checking purposes.

Data availability statement

Acknowledgments

Data management

No additional research data beyond the data presented and cited in this work are needed to validate the research findings in this work. Simulation data are openly available at the following URL/DOI: https://zenodo.org/record/7766237.

Appendix. Short review of MCMC

Markov Chain Monte Carlo (MCMC) is a class of theories that are used to create a probability distribution efficiently. Specifically, let us consider n variables $x_1, \cdots, x_n$ which may be real numbers or integers. The goal is to obtain a sequence of 'configurations' $\{x^{(i)}\} = (x_1^{(i)}, \cdots, x_n^{(i)})$ $(i = 1, 2, \cdots)$ whose distribution converges to the target probability distribution $P(x_1, \cdots, x_n)$. Such a sequence allows us to calculate the expectation value of a function $f(x_1, \cdots, x_n)$ as

$$\langle f(x_1, \cdots, x_n) \rangle \equiv \int dx_1 \cdots dx_n f(x_1, \cdots, x_n) P(x_1, \cdots, x_n)$$
$$= \lim_{N_{\text{config}} \to \infty} \frac{1}{N_{\text{config}}} \sum_{i=1}^{N_{\text{config}}} f\left(x_1^{(i)}, \cdots, x_n^{(i)}\right). \tag{32}$$

The MCMC algorithms [22, 36] are designed so that the sequence is a Markov chain, i.e. the probability of obtaining $\{x^{(i+1)}\}$ after $\{x^{(i)}\}$ depends only on $\{x^{(i)}\}$ and does not depend on $\{x^{(i-1)}\}, \{x^{(i-2)}\}, \cdots$, and the transition probability between configurations denoted by $T(\{x\} \to \{x'\})$ satisfies the following three conditions:

1. **Irreducibility**. The Markov chain defined by $T(\{x\} \to \{x'\})$ is irreducible, i.e. transition between any pair $\{x\}$ and $\{x'\}$ is possible with a finite number of steps.
2. **Aperiodicity**. The greatest common divisor of the numbers of steps needed for a transition from $\{x\}$ to itself is called the period. The transition probability $T(\{x\} \to \{x'\})$ is chosen in such a way that the Markov chain defined is aperiodic, i.e, the period is 1 for any $\{x\}$.
3. **Detailed balance condition** is satisfied, i.e.

$$P(\{x\}) \cdot T(\{x\} \to \{x'\}) = P(\{x'\}) \cdot T(\{x'\} \to \{x\}). \tag{33}$$

for any $\{x\}$ and $\{x'\}$.

In the Metropolis algorithm [22, 36], transition from $\{x\}$ to $\{x'\}$ and that from $\{x'\}$ to $\{x\}$ are proposed with the same probability, and the proposal $\{x\}$ to $\{x'\}$ is accepted with the probability $\min(1, P(\{x'\})/P(\{x\}))$. It is straightforward to check that the detailed balance condition is satisfied. Whether the other two conditions are satisfied depends on the details of the transition probability $T(\{x\} \to \{x'\})$.

In the MCMC algorithms, most of the computational resources are used to create *important* configurations dominating the probability distribution. This feature, which is called importance sampling, allows us to drastically save simulation time.

ORCID iDs

Junyu Liu ⓘ https://orcid.org/0000-0003-1669-8039
Masaki Tezuka ⓘ https://orcid.org/0000-0001-7877-0839

References

[1] Jordan S P, Lee K S M and Preskill J 2014 Quantum computation of scattering in scalar quantum field theories *Quantum Inf. Comput.* **14** 1014–80
[2] Jordan S P, Lee K S M and Preskill J 2012 Quantum algorithms for quantum field theories *Science* **336** 1130–3
[3] Macridin A, Spentzouris P, Amundson J and Harnik R 2018 Digital quantum computation of Fermion-Boson interacting systems *Phys. Rev.* A **98** 042312
[4] Macridin A, Spentzouris P, Amundson J and Harnik R 2018 Electron-phonon systems on a universal quantum computer *Phys. Rev. Lett.* **121** 110504
[5] Klco N and Savage M J 2019 Digitization of scalar fields for quantum computing *Phys. Rev.* A **99** 052335
[6] Macridin A, Li A C Y, Mrenna S and Spentzouris P 2022 Bosonic field digitization for quantum computers *Phys. Rev.* A **105** 052405
[7] Alexandru A, Bedaque P F, Carosso A, Cervia M J and Sheng A 2023 Qubitization strategies for bosonic field theories *Phys. Rev.* D **107** 034503
[8] Li A C Y, Macridin A, Mrenna S and Spentzouris P 2022 Simulating scalar field theories on quantum computers with limited resources *Phys. Rev.* A **107** 032603
[9] Lamm H, Lawrence S and Yamauchi Y (NuQS Collaboration) 2019 General methods for digital quantum simulation of gauge theories *Phys. Rev.* D **100** 034518
[10] Ciavarella A, Klco N and Savage M J 2021 Trailhead for quantum simulation of SU(3) Yang-Mills lattice gauge theory in the local multiplet basis *Phys. Rev.* D **103** 094501

[11] Alexandru A, Bedaque P F, Harmalkar S, Lamm H, Lawrence S and Warrington N C (NuQS Collaboration) 2019 Gluon field digitization for quantum computers *Phys. Rev.* D **100** 114501

[12] Ji Y, Lamm H and Zhu S (NuQS Collaboration) 2020 Gluon field digitization via group space decimation for quantum computers *Phys. Rev.* D **102** 114513

[13] Ji Y, Lamm H and Zhu S 2023 Gluon digitization via character expansion for quantum computers *Phys.Rev.D* **107** 114503

[14] Hackett D C, Howe K, Hughes C, Jay W, Neil E T and Simone J N 2019 Digitizing gauge fields: lattice Monte Carlo results for future quantum computers *Phys. Rev.* A **99** 062341

[15] Mathis S V, Mazzola G and Tavernelli I 2020 Toward scalable simulations of lattice gauge theories on quantum computers *Phys. Rev.* D **102** 094501

[16] Alexandru A, Bedaque P F, Brett R and Lamm H 2022 Spectrum of digitized QCD: glueballs in a S(1080) gauge theory *Phys. Rev.* D **105** 114508

[17] Somma R D Quantum simulations of one dimensional quantum systems (arXiv:1503.06319 [quant-ph])

[18] Rinaldi E, Han X, Hassan M, Feng Y, Nori F, McGuigan M and Hanada M 2022 Matrix-model simulations using quantum computing, deep learning and lattice Monte Carlo *PRX Quantum* **3** 010324

[19] Liu J 2021 Does Richard Feynman dream of electric sheep? topics on quantum field theory, quantum computing, and computer science *PhD Thesis* Caltech

[20] Bauer C W *et al* 2023 Quantum simulation for high energy physics *PRX Quantum* **4** 027001

[21] Liu J, Li Z, Zheng H, Yuan X and Sun J 2022 Some variational recipes for quantum field theories *Mach. Learn.: Sci. Technol.* **3** 045030

[22] Metropolis N, Rosenbluth A W, Rosenbluth M N, Teller A H and Teller E 1953 Equation of state calculations by fast computing machines *J. Chem. Phys.* **21** 1087–92

[23] Hanada M and Matsuura S 2022 *MCMC From Scratch — A Practical Introduction to Markov Chain Monte Carlo Method* (Springer)

[24] Kaplan D B, Katz E and Unsal M 2003 Supersymmetry on a spatial lattice *J. High Energy Phys.* JHEP05(2003)037

[25] Buser A J, Gharibyan H, Hanada M, Honda M and Liu J 2021 Quantum simulation of gauge theory via orbifold lattice *J. High Energy Phys.* JHEP09(2021)034

[26] Swendsen R H and Wang J-S 1987 Nonuniversal critical dynamics in Monte Carlo simulations *Phys. Rev. Lett.* **58** 86–88

[27] Maldacena J and Milekhin A 2018 To gauge or not to gauge? *J. High Energy Phys.* JHEP04(2018)084

[28] Berkowitz E, Hanada M, Rinaldi E and Vranas P 2018 Gauged and ungauged: a nonperturbative test *J. High Energy Phys.* JHEP06(2018)124

[29] Pateloudis S, Bergner G, Bodendorfer N, Hanada M, Rinaldi E and Schäfer A 2022 Nonperturbative test of the Maldacena-Milekhin conjecture for the BMN matrix model *J. High Energy Phys.* JHEP08(2022)178

[30] Kogut J B and Susskind L 1975 Hamiltonian formulation of Wilson's lattice gauge theories *Phys. Rev.* D **11** 395–408

[31] Haase J F, Dellantonio L, Celi A, Paulson D, Kan A, Jansen K and Muschik C A 2021 A resource efficient approach for quantum and classical simulations of gauge theories in particle physics *Quantum* **5** 393

[32] Alexandru A, Bedaque P F, Lamm H and Lawrence S (NuQS Collaboration) 2019 σ models on quantum computers *Phys. Rev. Lett.* **123** 090501

[33] Han X and Hartnoll S A 2020 Deep quantum geometry of matrices *Phys. Rev.* X **10** 011069

[34] Stokes J, De S, Veerapaneni S and Carleo G 2023 Continuous-variable neural-network quantum states and the quantum rotor model *Quantum Mach. Intell.* **5** 12

[35] Pescia G, Han J, Lovato A, Lu J and Carleo G 2022 Neural-network quantum states for periodic systems in continuous space *Phys. Rev. Res.* **4** 023138

[36] Rosenbluth M 1953 Proof of validity on Monte Carlo method for canonical averaging *Tehnical Report* Los Alamos Scientific Lab Rosenbluth M 2003 Proof of validity on Monte Carlo method for canonical averaging *AIP Conf. Proc.* **690** 32–38