ACADÉMIE
DES SCIENCES
INSTITUT DE FRANCE

Comptes Rendus

Physique

Thomas Ayral

Classical and quantum algorithms for many-body problems

Volume 26 (2025), p. 25-89

Online since: 27 January 2025

https://doi.org/10.5802/crphys.229

CENTRE MERSENNE

The Comptes Rendus. Physique *are a member of the*
Mersenne Center for open scientific publishing
www.centre-mersenne.org — e-ISSN : 1878-1535

ACADÉMIE
DES SCIENCES
INSTITUT DE FRANCE

Comptes Rendus. Physique
2025, Vol. 26, p. 25-89
https://doi.org/10.5802/crphys.229

Research article / *Article de recherche*

Classical and quantum algorithms for many-body problems

Algorithmes classiques et quantiques pour le problème à N corps

Thomas Ayral [a]

[a] Eviden Quantum Laboratory, Les Clayes-sous-Bois, France
E-mail: thomas.ayral@eviden.com

Abstract. The many-body problem is central to many fields, such as condensed-matter physics and chemistry, but also to combinatorial optimization, which is nothing but a classical many-body problem. This manuscript, written as part of an Habilitation à Diriger des Recherches, presents the various algorithmic approaches, both classical and quantum, to solving this problem. We begin by reviewing the main existing classical and quantum methods, focusing on their successes as well as their current limitations. In particular, we present the state-of-the-art in quantum methods, distinguishing between perfect and noisy processors. We then present recent work on combining classical and quantum algorithms to overcome the limitations inherent to both paradigms. In particular, we begin by showing how tensor networks, often used as reference tools to gauge the interest of quantum methods, can also be used to initialize a quantum computation, in addition to simulating it realistically. We then turn to the special case of fermionic problems. After describing a method based on natural orbitals for shortening, and thus making more reliable, quantum circuits to prepare fermionic states, we present a method based on slave spins for using a platform of Rydberg atoms to simulate lattice models of fermions. Finally, we show how these same Rydberg platforms can be used to solve combinatorial problems, and how decoherence influences the quality of the results obtained. This leads to the definition of a new utility metric for quantum processors, the Q-score.

Résumé. Le problème à N corps est un problème central pour nombre de domaines comme la physique de la matière condensée ou la chimie, mais aussi celui de l'optimisation combinatoire, qui n'est autre qu'un problème à N corps classique. Ce manuscrit, rédigé dans le cadre d'une Habilitation à Diriger des Recherches, présente les différentes approches algorithmiques, qu'elles soient classiques ou quantiques, pour résoudre ce problème. Nous commenàçons par y passer en revue les principales méthodes classiques et quantiques existantes, avec un accent mis sur leurs succès ainsi que leurs limitations actuelles. En particulier, nousc présentons un état de l'art des méthodes quantiques, en distinguant processeurs parfaits et processeurs bruités. Ensuite, nous présentons des travaux récents permettant de combiner algorithmes classiques et quantiques pour surmonter les limitations inhérentes aux deux paradigmes. En particulier, nous commenàçons par montrer comment les réseaux de tenseurs, souvent utilisés comme outils de référence pour jauger de l'intérêt des méthodes quantiques, peuvent aussi être utilisés pour initialiser un calcul quantique, en plus de le simuler de faàçon réaliste. Nous passons ensuite au cas particulier des problèmes fermioniques. Après avoir décrit une méthode à base d'orbitales naturelles permettant de raccourcir, et donc de fiabiliser, des circuits quantiques pour préparer des états fermioniques, nous exposons une méthode à base de spins esclaves permettant d'utiliser une plateforme d'atomes de Rydberg pour simuler des modèles de fermions sur réseau. Nous montrons enfin comment ces mêmes plateformes de Rydberg peuvent être utilisées pour résoudre des problèmes combinatoires, et comment la décoherence influence la qualité des résultats obtenus. Ceci nous amène à la définition d'une nouvelle métrique d'utilité des processeurs quantiques, le Q-score.

ISSN (electronic): 1878-1535

Keywords. Many-body physics, Quantum computing, Algorithms, Condensed-matter physics, Numerical methods.

Mots-clés. Problème à N corps, Informatique quantique, Algorithmes, Physique de la matière condensée, Méthodes numériques.

Manuscript received 19 December 2024, accepted 20 December 2024.

1. Introduction

Quantum many-body problems—for instance, systems of interacting electrons in solids—are among the most difficult problems to solve on classical computers. This is directly due to the strongly interacting nature of the involved particles. These interactions in turn cause a failure of mean-field approaches, which essentially try to capture many-body physics within an effective single-body description—in fact, a classical description as it does not involve a key quantum property, entanglement. Such mean-field approaches must be replaced, or rather augmented, with "strongly-correlated" methods that capture the many-body nature of the phenomena at play, this time with entanglement at the center of the stage. This comes at a cost: whether exact diagonalization methods (within a carefully identified low-energy or active subspace), Monte-Carlo methods, or tensor-network methods—all these methods come with an exponential cost as a function of some parameter. This has not prevented huge algorithmic progress in the recent decades, with an increasing understanding of more and more exotic phases of matter. And yet some regimes still elude our understanding. Doped, low-temperature regimes of models like the Hubbard model, which are believed to capture the physics of high-temperature superconductors, are but one example. Another one is the dynamics of fermionic or spin models, whose study is nowadays limited to short times and/or small systems.

These limitations spurred physicists to envision, in the early 1980s, a new type of processors, called quantum processors. They experienced a boom in the 2010s thanks to the advent of experimental processors with tens to hundreds of controlled particles or even quantum bits. Since early on, quantum processors have been assigned to two different tasks. One is solving abstract math problems like factoring numbers, with exponential speedup guarantees in theory. The second one, which will be the main focus of this work, consists in using quantum processors as artificial quantum many-body systems that mimic Nature's quantum many-body systems. In this field, too, exponential speedup guarantees were established for simulating the dynamics of quantum systems compared to classical computers.

In practice, however, quantum computers are fragile systems. A phenomenon called decoherence introduces steep limitations to the quality of the outputs of quantum computations. These limitations accumulate exponentially with the complexity (size, duration) of the task at hand. A careful balance thus has to be stricken between the hoped-for acceleration compared to classical algorithms, and the penalty coming from decoherence—all this while taking into account the rules of quantum computing, which differ from those of classical computing.

In this work, we explore various ways to strike this balance, with the following overarching question as our horizon: *how to best combine classical and quantum algorithms to compute the properties of many-body systems?*

To answer this question, we will first identify the strengths and weaknesses of the main classical algorithms for the many-body problem (Section 2). Our main focus will be the Hubbard model, a prototypical model for strongly-correlated electron systems, although we will also extend the discussion to quantum chemistry problems as well as combinatorial optimization problems, which can be regarded as classical many-body problems. We will carry out a similar program for quantum algorithms for the many-body problem, with a focus on algorithms geared at current processors, namely variational quantum algorithms (Section 3).

After these introductory sections, we will explain how classical and quantum algorithms can be combined to optimize our usage of quantum processors (Section 4). We will first show that tensor network techniques are not only powerful diagnosis tools for quantum advantage and promising predictive tools to simulate noisy quantum computers, but that they can also be used in combination with quantum algorithms to reach accurate results in possibly hard parameter regimes. We will then present two hybrid methods for handling fermionic models with more robustness to decoherence, as well as quantitative criteria for the success of two important fermionic quantum algorithms. Finally, we will tackle a subclass of *classical* many-body problems encountered in the field of combinatorial optimization, and how they can be tackled with quantum algorithms. In particular, we investigate the effect of decoherence on the quality of the computed output.

2. Many-body problems: a few classical approaches, their strengths and their limitations

Many-body or strongly-correlated problems are encountered in many fields of physics including materials science, quantum chemistry and nuclear physics. Their commonality is the failure of conventional mean-field theories to describe some of their phases. This failure requires the development of advanced classical algorithms, all of which, as we shall see, have to deal with an exponential difficulty in some guise.

We will first focus on algorithms that directly tackle the problem at hand (Section 2.1). The goal is to shed light on the successes of these algorithms despite this exponential wall, and identify potential bottlenecks. We will mainly focus on condensed-matter many-body problems, but will also introduce a few notions of quantum chemistry, which is an important candidate application for quantum processors.

We will then introduce classical problem reduction techniques that have been developed to reduce the number of relevant degrees of freedom, leading to a smaller, yet still many-body subproblem (Section 2.2). Such a reduced model will in general be better suited to a quantum processing task.

2.1. *The Hubbard model and exponential walls*

Many-body problems come in many flavors. In materials science and quantum chemistry, the physics of the electronic degrees of freedom is well described (after neglecting the ionic motion, an approximation called the Born–Oppenheimer approximation) by the so-called electronic-structure Hamiltonian,

$$H = \sum_{\alpha,\beta=1}^{N_o} \sum_{\sigma=\uparrow,\downarrow} h_{\alpha\beta} c^\dagger_{\alpha,\sigma} c_{\beta,\sigma} + \frac{1}{2} \sum_{\sigma,\sigma'} \sum_{\alpha\beta\gamma\delta} v_{\alpha\beta\gamma\delta} c^\dagger_{\alpha,\sigma} c_{\beta,\sigma} c^\dagger_{\gamma,\sigma'} c_{\delta,\sigma'}, \tag{1}$$

which describes the kinetic and potential energy (first term) and Coulomb interaction (second term) of the electronic system. The kinetic and interaction tensors h and v are determined by the geometry of the problem and by the choice of single-particle basis set $\{\phi_\alpha(\boldsymbol{r})\}_{\alpha=1\dots N_o}$ (sometimes called atomic orbitals), created/annihilated by the creation and annihilation operators $c^\dagger_{\alpha,\sigma}$ and $c_{\alpha,\sigma}$ ($\sigma = \uparrow,\downarrow$ denotes the spin of electrons). Provided the basis set size N_o is large enough, the ground state energy of H will be the exact ground state energy of the solid. In practice, one must truncate the basis set to a small N_o. (The size N_o that allows keeping a reasonable "accuracy", namely a small enough error with respect to the exact ground state energy in the $N_o \to \infty$ limit, depends on the choice of basis).

In condensed-matter physics, and in particular strongly-correlated materials that typically encompass partially filled d and f shells, which correspond to localized atomic orbitals, one usually picks a localized basis set. Localized means that $\phi_\alpha(\boldsymbol{r})$ is centered around an atomic position $\boldsymbol{R}_i$, e.g. $\alpha \equiv (\boldsymbol{R}_i, a)$ with a an atomic orbital like 1s, 2s, 2p... Thus, $h_{\alpha\beta}$ can typically be interpreted as a tunneling matrix element between atomic sites. It is thus usually limited to neighboring sites in the material. As for $v_{\alpha\beta\gamma\delta}$, it also has a "local" structure that endows H with a special structure: it is not *any* large ($2^{N_o} \times 2^{N_o}$) Hermitian matrix. In condensed-matter systems, this locality is made even more obvious by high-energy electrons, which tend to screen the interaction between low-energy electrons. This screening phenomenon often suggests a simplification of Equation (1): one can keep only local interaction matrix elements, which yields the so-called Hubbard model [1]. In its single-band version, it reads

$$H = \sum_{i,j=1}^{n} \sum_{\sigma} t_{ij} c^{\dagger}_{i,\sigma} c_{j,\sigma} + U \sum_{i=1}^{n} n_{i\uparrow} n_{i\downarrow} = H_{\mathrm{kin}} + H_{\mathrm{int}}, \tag{2}$$

where the creation operators $c^{\dagger}_{i,\sigma}$ create electrons in orbitals $\{\phi_i(\boldsymbol{r})\}_{i=1\dots n}$ localized around site $\boldsymbol{R}_i$ of the lattice. The Hubbard interaction U is the value of the on-site interaction after screening. The Hubbard model (and other related "low-energy" models) can be derived, more or less approximately, from the electronic structure Hamiltonian. In particular, U can be computed in an ab-initio fashion under some assumptions using advanved techniques like the constrained random phase approximation (cRPA, [2]). This "downfolding" procedure can also be carried out using tensor network techniques (which we will discuss in further detail below). For instance, [3] derives a single-band Hubbard model for the cuprate materials (that are known to exhibit high-temperature superconductivity since the 1980s [4]) starting from a three-band model.

Although simple-looking, this minimal model is difficult to solve in regimes where both the kinetic energy (described by the tensor t_{ij}) and the interaction U are of comparable magnitude. In such regimes, the influence of the kinetic term, which favors delocalized states through tunnelling, competes with that of the interaction term, which favors localized states that minimize Coulomb repulsion, and form an exotic phase of matter called a "Mott insulator" [5].

2.1.1. *The failure of mean-field methods*

Standard mean-field theories like Hartree–Fock (HF) theory or more advanced ones (like density functional theory (DFT, [6]), which is not strictly speaking mean-field but reduces the problem to a single-particle Kohn–Sham problem [7]) take interactions into account only in an averaged fashion. This simplication allows for "efficient" classical algorithms. Here, and throughout this report, "efficient" will refer to algorithms whose run time and memory complexity is polynomial in the problem size n. For the case of HF or DFT, the complexity is $O(n^3)$, dominated by the diagonalization of a $n \times n$ (or $N_o \times N_o$ if dealing with the electronic structure problem Equation (1)) linear system of equations. Yet, these methods fail to describe the Mott insulating phase: they can predict the opening of antiferromagnetic gaps but not of paramagnetic Mott gaps.

One can (at least partially) account for this failure by looking at a simple two-site Hubbard problem (*aka* the Hubbard molecule) with two electrons. In this case, in the limit of large on-site interactions U (or zero hopping), the ground state is of the form

$$|\Psi_0\rangle = \frac{1}{\sqrt{2}} \left(|\uparrow,\downarrow\rangle + |\downarrow,\uparrow\rangle\right), \tag{3}$$

(with $|\uparrow,\downarrow\rangle = c^{\dagger}_{L\uparrow} c^{\dagger}_{R\downarrow} |\mathbf{0}\rangle$, meaning one electron on the left site with spin up, one electron on the right site with spin down), because the on-site Coulomb interaction penalizes states with double occupancies, like $|0,\uparrow\downarrow\rangle$ or $|\uparrow\downarrow,0\rangle$.

As we are about to see, mean-field theory will lead to a qualitatively different answer. Indeed, (unrestricted) Hartree–Fock theory amounts to replacing H by

$$H_{\mathrm{m.f}} = \sum_{i,j=1}^{n} \sum_{\sigma} \tilde{t}_{ij}^{\sigma} c_{i,\sigma}^{\dagger} c_{j,\sigma}, \tag{4}$$

with a modified hopping matrix

$$\tilde{t}_{ij}^{\sigma} = t_{ij} + U \langle n_{i\bar{\sigma}} \rangle \delta_{ij}, \tag{5}$$

that depends on the densities $\langle n_{i\sigma} \rangle$. A quadratic Hamiltonian, $H_{\mathrm{m.f}}$ will necessarily lead to a ground state in the form of a single Slater determinant, namely a state that can be written as $\prod_{\alpha\sigma} (\tilde{c}_{\alpha\sigma}^{\dagger})^{n_{\alpha\sigma}} |\mathbf{0}\rangle$, where the creation operators $\tilde{c}_{\alpha\sigma}^{\dagger}$ are defined as linear combinations of the original operators $\tilde{c}_{\alpha\sigma}^{\dagger} = \sum_i V_{\alpha i} c_{i\sigma}^{\dagger}$ (V is unitary to preserve the anticommutation relations of fermionic operators).

It turns out that the two (up and down) electrons will populate the "bonding" orbital $\tilde{c}_{B\sigma}^{\dagger} = (c_{L\sigma}^{\dagger} + c_{R\sigma}^{\dagger})/\sqrt{2}$, leading to a state of the form

$$|\Psi_{\mathrm{HF}}\rangle = \tilde{c}_{B,\uparrow}^{\dagger} \tilde{c}_{B,\downarrow}^{\dagger} |\mathbf{0}\rangle = \frac{1}{2} \left(|\uparrow, \downarrow\rangle + |\downarrow, \uparrow\rangle + |0, \uparrow\downarrow\rangle + |\uparrow\downarrow, 0\rangle \right). \tag{6}$$

This state has contributions from configurations with double occupancies (last two terms) that are not present in the true ground state Equation (3). This is due to the fact that HF (and DFT) underestimates the influence of local interactions by handling them in an approximate, mean-field way. From a technical point of view, it is also worth noting that the ground state (3) cannot be written as a Slater determinant: it is said to be (in a quantum chemistry context) multi-reference, as opposed to HF states, which are "single-reference" states (see [8] for an interesting discussion of multi-reference vs single-reference character). In yet another context, one could see that the state represented by Equation (3) is an entangled state, as opposed to (6), which can be factorized.

Describing multi-reference states (aka strongly-correlated, or states with "static" correlations) requires more advanced methods.

2.1.2. *Direct diagonalization*

A direct diagonalization (also called exact diagonalization, or full configuration interaction) of the $4^n \times 4^n$ matrix representation of H in the Fock basis (the basis made up of all states of the form $|n_{1\uparrow}, n_{1\downarrow}, \ldots, n_{n\uparrow}, n_{n\downarrow}\rangle = \prod_{i=1}^{n} \prod_{\sigma=\uparrow,\downarrow} (c_{i\sigma}^{\dagger})^{n_{i\sigma}} |\mathbf{0}\rangle$) is limited to very small lattice sizes n due to the exponential ($\propto (4^n)^3$) cost of diagonalizing this matrix.

More advanced diagonalization approaches are routinely used, like the Lanczos method [9]. It essentially amounts to a smart way of orthonormalizing the so-called Krylov basis $\{H|\Phi_0\rangle, H^2|\Phi_0\rangle, \ldots, H^k|\Phi_0\rangle\}$ (with $|\Phi_0\rangle$ a suitable starting state, e.g. the HF state), which spans a (Krylov) subspace that is a good approximation of the low-energy eigenspace of the full H (the Krylov basis is inspired from the power method, where powers of H, $H^k|\Phi_0\rangle$, are used to amplify a given eigenvector, here the lowest). If H is a sparse matrix (which is the case in Equation (2), as there are typically $O(n)$ nonzero terms per line in the $4^n \times 4^n$ H matrix), the cost of constructing the $k \times k$ matrix of H in this basis is $O(4^n)$ instead of $O(16^n)$: one can thus push the limit a bit further (and have some control on the accuracy by increasing the size k of the Krylov subspace). The use of symmetries also allows to extend exact diagonalization methods. For instance, [10] reach systems of 50 spins. Despite these improvements, one always faces an exponential bottleneck in the number n of sites.

2.1.3. *Quantum Monte-Carlo algorithms*

Another way to tackle the properties of the Hubbard model is to resort to so-called quantum Monte-Carlo methods. These methods come in several flavors, but boil down to rewriting e.g. expectation values of observables in the generic form

$$\langle O\rangle = \mathrm{Tr}(\rho\hat{O}) = \frac{\sum_{x\in\mathscr{C}} w(x)f(x)}{\sum_{x\in\mathscr{C}} w(x)} \tag{7}$$

with $\mathscr{C}$ a very large (exponential or worse) configuration space, $w(x)$ the weight of a configuration x and $f(x)$ the value of the configuration. (For instance, the right-hand side of (7) can obtained for a thermal state $\rho = \mathrm{e}^{-H/(k_\mathrm{B}T)}/Z$ by Taylor-expanding the exponential in powers of H_int). From (7), one can define a probability distribution $p(x) = |w(x)|/\sum_x |w(x)|$ and approximate the corresponding expression by sampling:

$$\langle O\rangle = \frac{\sum_{x\in\mathscr{C}} p(x)\mathrm{sign}(w(x))f(x)}{\sum_{x\in\mathscr{C}} p(x)\mathrm{sign}(w(x))} \approx \frac{\langle\mathrm{sign}\cdot f\rangle_\mathrm{MC}}{\langle\mathrm{sign}\rangle_\mathrm{MC}}. \tag{8}$$

Here, $\langle g\rangle_\mathrm{MC} = (1/\mathscr{N})\sum_{i=1}^{\mathscr{N}} g(x_i)$, where the $\mathscr{N}$ configurations x_i are sampled according to the $p(x)$ distribution (either through direct sampling or Markov chain sampling, see [11]). The finite number of samples $\mathscr{N}$ used in Monte-Carlo leads to a statistical error

$$\Delta g \approx \sqrt{\frac{\mathrm{Var}(g)}{\mathscr{N}}} \tag{9}$$

(for uncorrelated samples), which can be made systematically small by increasing the number $\mathscr{N}$ of samples. However, when the denominator $\langle\mathrm{sign}\rangle_\mathrm{MC}$ in (8) vanishes, the number of samples must increase dramatically to ensure a constant error ΔO: this is the sign problem phenomenon that generically plagues fermionic computations. For instance, for a thermal state, one obtains $\Delta O/\langle O\rangle = O(\mathrm{e}^{N_\mathrm{el}/(k_\mathrm{B}T)}/\sqrt{\mathscr{N}})$, with N_el the number of electrons [12]: keeping a constant error requires a number of samples that is exponential in the number of electrons and inverse temperature T.

In practice, there are ways to avoid or overcome, at least in some regimes, this exponential sign problem. Let us give three illustrative examples.

Diagrammatic Monte-Carlo. One example is diagrammatic Monte-Carlo (DiagMC, [13]), which consists in computing expectation values $\mathrm{Tr}(\rho\hat{O})$ of observables $\hat{O}$ (with e.g. thermal states $\rho = \mathrm{e}^{-H/(k_\mathrm{B}T)}/Z$) by expanding the exponential of H (e.g. (2)) in powers of the interaction term H_int of H. The corresponding series $\langle O\rangle = \mathrm{Tr}(\rho\hat{O}) = \sum_{m=0}^{\infty} O_m$ contains contributions O_m at the mth order in the interaction U. These contributions can in turn be written in the form $O_m = \int_X \sum_{T\in\mathscr{S}_m} D(T,X)$, with T a Feynman diagram of order m ($\mathscr{S}_m$ is the group of all topologies at order m) and X internal variables (e.g. space and imaginary time). DiagMC uses configurations $x \equiv (X,T)$ and weights $w(x) = D(T,X)$, while the more recent “connected determinant DiagMC” (CDet, [14]) uses configurations $x = X$ with weights $w(x) = \sum_{T\in\mathscr{S}_m} D(T,X)$.

Let us focus specifically on the more advanced CDet (the conclusions are similar, although slightly less powerful, for DiagMC). The computation of the contributions at mth order turn out to have a $O(3^m)$ complexity. While this method thus appears to have exponential complexity in the expansion order (the run time, $t = \mathrm{e}^{\alpha m}$, with $\alpha = \log(3)$, is exponential for a given order), the error decreases exponentially with the expansion order m: once one is beyond the convergence radius of the series, $m \sim \log(1/\epsilon)$ is enough to achieve a truncation error $\epsilon = |\langle O\rangle - \langle O\rangle_m|$, with $\langle O\rangle_m$ the truncated series. Therefore, the overall run time for a fixed error scales polynomially: we have $t = 1/\epsilon^\alpha$, a polynomially scaling run time as a function of $1/\epsilon$ [15]. (Of course, to compute beyond the convergence radius $U > U_\mathrm{cr}$ of the series, one recovers an exponential complexity: this approach is perturbative in nature).

Auxiliary-field quantum Monte-Carlo. An alternative to DiagMC or CDet—both unbiased methods—is auxiliary field quantum Monte-Carlo (AFQMC, [16]). In this method (like in diffusion Monte-Carlo (DMC) or projector techniques [17]), an imaginary-time evolution $\mathrm{e}^{-\tau H}$ starting from a initial wavefunction $|\Phi_0\rangle$ is performed: the time-evolved state

$$|\Psi(\tau)\rangle = \mathrm{e}^{-\tau H}|\Phi_0\rangle / \|\mathrm{e}^{-\tau H}|\Phi_0\rangle\|$$

converges to the ground state in the large imaginary time limit. This formal algorithm requires the costly (thus impossible in practice) exponentiation of the exponentially large matrix H. In practice, this evolution is performed by splitting $\mathrm{e}^{-\tau H}$ in N_t "Trotter" slices

$$\mathrm{e}^{-H\tau} = \prod_{k=1}^{N_t}\left(\mathrm{e}^{-H_{\mathrm{kin}}\Delta\tau}\prod_{i=1}^{n}\mathrm{e}^{-U\Delta\tau n_{i\uparrow}n_{i\downarrow}}\right) + O\left(\frac{\tau^2}{N_t}\right), \tag{10}$$

and by expanding each Trotter slice using a Hubbard–Stratonovich decoupling of the interaction term,

$$\mathrm{e}^{-U\Delta\tau n_{i\uparrow}n_{i\downarrow}} = \sum_{s_i=\pm 1}\mathrm{e}^{-f(U,\Delta\tau,s_i)H_{\mathrm{f}}}, \tag{11}$$

with H_{f} a quadratic Hamiltonian and $\Delta\tau = \tau/N_t$. The energy of the resulting wavefunction can thus be written in the requisite form (7):

$$E(\tau) = \frac{\langle\Phi_0|H\mathrm{e}^{-2\tau H}|\Phi_0\rangle}{\langle\Phi_0|\mathrm{e}^{-2\tau H}|\Phi_0\rangle} = \frac{\sum_x\langle\Phi_0|H|\Phi(x)\rangle}{\sum_x\langle\Phi_0|\Phi(x)\rangle},$$

with $x = \boldsymbol{s}_1,\dots,\boldsymbol{s}_n, \boldsymbol{s}_i = (s_i^1,\dots,s_i^{N_t})$, and the wavefunction

$$|\Phi(x)\rangle = \prod_{k=1}^{N_t}\left(\mathrm{e}^{-H_{\mathrm{kin}}\Delta t}\prod_{i=1}^{n}\mathrm{e}^{-f(U,\Delta\tau,s_i^k)H_{\mathrm{f}}}\right)|\Phi_0\rangle.$$

The exponential sums appearing in the numerator and denominator are sampled using a Markov chain Monte-Carlo algorithm. For a given configuration x, $|\Phi(x)\rangle$ is a Slater determinant provided the starting point $|\Phi_0\rangle$ is also one, since applying a quadratic evolution (via Hamiltonians H_{kin} and H_{f}) to a Slater determinant leads to a Slater determinant. Being a Slater determinant, it can be stored and computed efficiently (on a classical computer). Sampling this multidimensional sum a priori allows to perform the evolution to the ground state and thus compute ground state expectation values.

In practice, however, the sign of $\langle\Phi_0|\Phi(x)\rangle$ changes, which leads to a sign problem. To avoid it, importance sampling strategies are usually implemented in the form of a "trial wavefunction" that is used to avoid sign changes (this variant is dubbed constrained-path Monte-Carlo, CPMC, [18]). This removes the sign problem but introduces a bias in the method (contrary e.g. to the diagrammatic Monte-Carlo methods introduced above, which are unbiased). This bias is all the larger as the trial wavefunction is far away from the true ground state. Constructing a good enough trial wavefunction is thus essential... but brings additional classical cost that needs to be taken into account in the computational burden of the method.

Note that there also exists DMC-like methods that do not require a trial wavefunction and therefore do not suffer from a potential bias: this is the case, for instance, of the FCIQMC method [19]. There, the limiting factor is the number of MC "walkers" (which are also used in CPMC to represent the wavefunction in a stochastic manner), which is supposed to saturate when the full configuration interaction (FCI) limit is reached: the number of walkers at saturation can be very large.

Variational Monte-Carlo. A third and last widely used example is variational Monte-Carlo (VMC). It is based on a parametric representation

$$|\Psi(\boldsymbol{\theta})\rangle = \sum_{x=0}^{2^n-1} \psi_{\boldsymbol{\theta}}(x)|x\rangle \tag{12}$$

of the wavefunction, where $\boldsymbol{\theta}$ is a list of parameters that completely characterize the state, and $\{|x\rangle\}$ is the Fock basis. The Rayleigh–Ritz variational principle guarantees that

$$E(\boldsymbol{\theta}) = \frac{\langle\Psi(\boldsymbol{\theta})|H|\Psi(\boldsymbol{\theta})\rangle}{\langle\Psi(\boldsymbol{\theta})|\Psi(\boldsymbol{\theta})\rangle} \geq E_0$$

(with E_0 the exact ground state energy of H). The goal of VMC is to find the set of parameters $\boldsymbol{\theta}^\star$ that minimizes $E(\boldsymbol{\theta})$. For this, one needs to compute $E(\boldsymbol{\theta})$. This is achieved by inserting $I = \sum_x |x\rangle\langle x|$ in $E(\boldsymbol{\theta})$, which yields:

$$E(\boldsymbol{\theta}) = \sum_x \frac{|\langle\Psi(\boldsymbol{\theta})|x\rangle|^2}{\langle\Psi(\boldsymbol{\theta})|\Psi(\boldsymbol{\theta})\rangle} \frac{\langle x|H|\Psi(\boldsymbol{\theta})\rangle}{\langle x|\Psi(\boldsymbol{\theta})\rangle} = \sum_x p_{\boldsymbol{\theta}}(x) E_{\mathrm{loc}}(x)$$

with the probability distribution $p_{\boldsymbol{\theta}}(x) = |\psi_{\boldsymbol{\theta}}(x)|^2/\langle\Psi(\boldsymbol{\theta})|\Psi(\boldsymbol{\theta})\rangle$ and the local energy $E_{\mathrm{loc}}(x) = \langle x|H|\psi_{\boldsymbol{\theta}}\rangle/\psi_{\boldsymbol{\theta}}(x)$. This exponential sum is sampled as

$$\sum_x p_{\boldsymbol{\theta}}(x) E_{\mathrm{loc}}(x) \approx \frac{1}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} E_{\mathrm{loc}}(x_i)$$

with samples x_i drawn from $p_{\boldsymbol{\theta}}(x)$. VMC relies on an efficient computation of $E_{\mathrm{loc}}(x)$ for a given x, and the possibility to sample from $p_{\boldsymbol{\theta}}(x)$. The method has gained particular traction in the recent years with the emergence of deep neural networks (dubbed quantum neural networks, [20]) to represent $\psi_{\boldsymbol{\theta}}(x)$. Some allow direct sampling, some require the use of Markov chain Monte-Carlo.

As always in MC methods, the finite number $\mathcal{N}$ of samples induces statistical noise (see Equation (9)). One nice feature of VMC, though, is the behavior of the variance when $\psi_{\boldsymbol{\theta}}(x)$ is close to the sought-after ground eigenstate: then, $E_{\mathrm{loc}}(x) = \langle x|H|\psi_{\boldsymbol{\theta}}\rangle/\psi_{\boldsymbol{\theta}}(x) \approx E_0$, namely the energy is independent of x, therefore the variance vanishes. This means that as one gets closer to the solution, one needs fewer and fewer samples to reach the same statistical accuracy.

The main limitation of VMC is the representational power of $\psi_{\boldsymbol{\theta}}(x)$ and the optimization of its parameters to minimize its energy. We refer to reader to [21] and [22] for more in-depth discussions of Monte-Carlo algorithms (and their relation with quantum algorithms).

The pros and cons of the three MC methods we just presented may be summarized as follows:

- DiagMC and CDet are perturbative in nature: they will perform well in weak correlation regimes, but fail in stronger correlation regimes;
- AFQMC is part of the family of projector MC methods: it will perform well if a good trial state is provided. Otherwise, it will be plagued by a large bias (due to a wrong trial state) or a large variance (due to the sign problem);
- VMC is unbiased (to the extent that the variational ansatz is expressive enough) and not perturbative, but it relies on the optimization of a nonconvex function, a problem that is generically hard to solve (in fact, it is “NP hard”, a terminology we will discuss later, in Section 4.3.1).

2.1.4. *Tensor network methods*

A major alternative to Monte-Carlo methods is provided by so-called tensor network methods. They consist in representing the state of the system as a graph whose vertices are tensors, and whose edges are chosen to reflect e.g. the lattice topology (but not necessarily). The combined memory footprint of these tensors (of the order of $n\chi^d$, with d the number of neighbors of each vertex, and χ the dimension of the tensor legs) is meant to be much smaller than the exponential

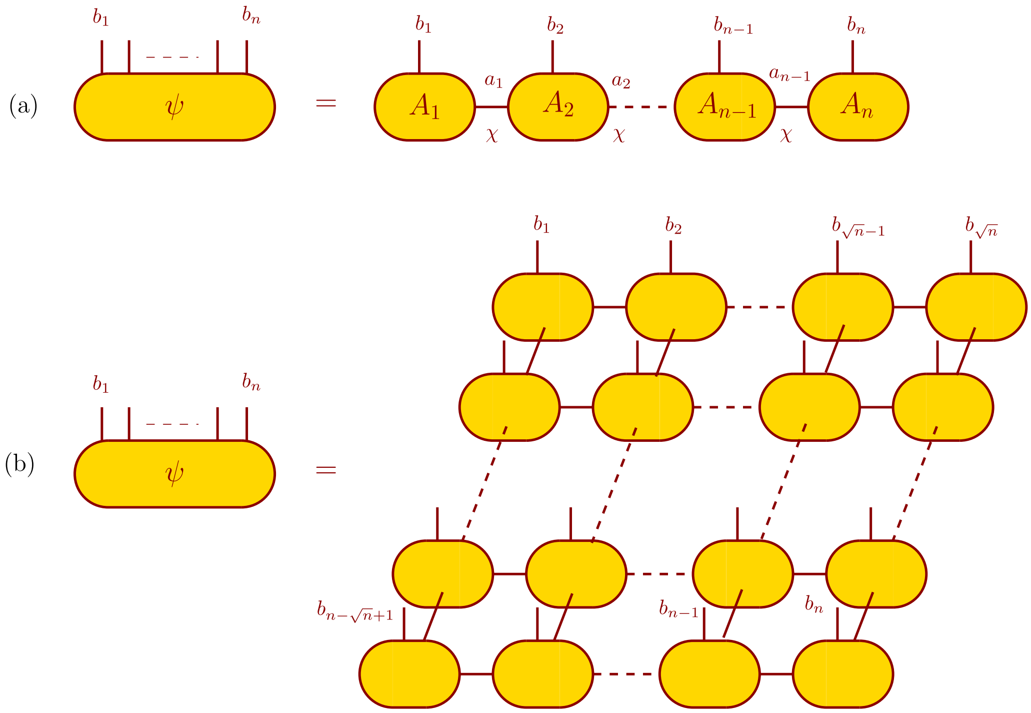

Figure 1. Two examples of tensor networks. (a) Matrix product state (MPS): linear graph. (b) Projected entangled pair state (PEPS): graph with a $\sqrt{n} \times \sqrt{n}$ square lattice topology.

footprint of storing the wavefunction $|\Psi\rangle$ (4^n) or thermal state ρ ($4^n \times 4^n$) of the Hubbard model. This exponential in the Hilbert space size is generically traded for an exponential in the entanglement of the state to be represented, as we shall see below.

The simplest example of such tensor networks are matrix product states (MPS, [23], Figure 1(a)), whose graph is simply a line graph, namely each tensor has two "neighbors" on its left and on its right (except for the first and last tensors). The main parameter of a MPS (and, for that matter, of other tensor networks) is the so-called bond dimension χ. $\chi = 1$ corresponds to a product (factorized) state, i.e. without entanglement. If the so-called von Neumann entanglement entropy of the state, which measures the level of entanglement of the state, is called S, then one must choose

$$\chi \geq 2^S \tag{13}$$

to represent this state with a MPS. Thus, a MPS with a small bond dimension, and thus a small storage cost ($\propto n\chi^2$), can potentially represent states with low entanglement. Algorithms to manipulate these states then generally scale as $O(\text{poly}(\chi))$ (usually $O(\chi^3)$, which corresponds to the cost of compressing the state through a singular value decomposition (SVD), or more advanced algorithms). This property is very useful for instance for studying ground states of Hamiltonians with local terms, which typically display a small entanglement entropy (S is even independent of system size for gapped systems in 1D, [24]).

Most importantly, tensor networks illustrate the fact that entanglement is a key quantity to consider when investigating the "hardness" of a computational problem for a classical algorithm. Currently, MPSs are widely used for studying 1D systems at equilibrium. Difficulties arise for higher dimensions or for out-of-equilibrium problems.

For higher dimensions, indeed, even when an "area law" is fulfilled, the entropy nevertheless still increases with dimension of the system: in 2D, the bond dimension is exponential in the linear dimension L of the $L \times L$ lattice. Tensor networks based on higher dimensional graphs than line (1D) graphs have been developed to try and overcome this challenge. For instance, projected entangled pair states (PEPS, [25], Figure 1(b)) are based on a 2D lattice graph. However, contrary to MPS, observable expectation values cannot be computed efficiently (polynomially in χ) for PEPS. In fact, "contracting" (i.e. summing over all internal indices of) a generic tensor network can be shown to scale (both in CPU and memory) as $\mathrm{e}^{T(G)}$, where $T(G)$ is the so-called treewidth of the TN's graph G, namely the minimum width over the tree decompositions of G ([26]; graphs that are trees have treewidth 1, a $M \times N$ square lattice has treewidth $\min(M,N)$). The recently introduced isometric tensor networks (IsoTNS, [27]) solve this issue, but with a quite prohibitive cost of $O(\chi^7)$. Recently, approximate contraction methods of PEPS have been introduced that use a belief propagation algorithm [28]. This type of approximate contraction gives accurate results for PEPS (or more generally tensor networks) defined on sparse graphs [29], namely graphs with a low connectivity (and is exact for one-dimensional graphs, i.e. MPS).

As for out-of-equilibrium physics, the challenge for MPS also comes from large entropies: the entropy tends to increase with time (linearly after a sudden change or quench in the Hamiltonian parameters, [30]), causing the requisite bond dimension to increase accordingly, following Equation (13). This difficulty is compounded by going to two or more dimensions for the reasons we just explained.

2.1.5. *Towards a quantitative handshake of classical methods for the two-dimensional Hubbard model?*

The Hubbard model in two dimensions has been tackled through a variety of methods including those we described above. Despite the difficulties associated with the strongly-correlated character of the model, agreement between various methods has been reached in some parameter regimes [31].

One important frontier is that of the doped, low-temperature phase of the model with strong interactions ($U/t \approx 8$). Recent comparative DMRG (MPS) and AFQMC studies of the two-dimensional Hubbard model with nearest-neighbor t and next-nearest-neighbor t' hopping on a square lattice have led to consistent pictures of the phenomenology of high-Tc cuprates in the zero-temperature regime [32]. In particular, the coexistence of partially filled stripe order with superconductivity in the ground states found by both methods on the hole-doped side has been observed in the high bond dimension regime (for MPS) and large-size limit (for AFQMC), suggesting that the Hubbard model with t' contains the main ingredients of high-Tc superconductivity. However, these studies were limited to zero temperatures (ground states), so that no direct computation of the superconducting temperature was possible. No dynamical properties were studied either.

2.1.6. *A word on quantum chemistry methods*

We have so far reviewed a few major methods used to tackle the Hubbard model, a model that captures the physics of certain solids. For chemical compounds, the simplification from the electronic structure Hamiltonian (Equation (1)) to the Hubbard model is not possible. In particular, the structure of the Coulomb interaction is in general more complicated. This makes Monte-Carlo methods based on a expansion in powers of the interaction term (like DiagMC above) much more costly, if not irrelevant.

We review below a few widespread algorithms designed for quantum chemistry problems, with the goal of identifying computational bottlenecks that could be overcome with quantum computers.

Configuration interaction (CI, [33]). CI is a variational method like VMC, where the variational state is a linear combination of a few states:

$$|\Psi(\boldsymbol{\theta})\rangle = \theta_0|\Phi_{\mathrm{HF}}\rangle + \sum_{a,i}\theta_i^a c_a^\dagger c_i|\Phi_{\mathrm{HF}}\rangle + \sum_{ab,ij}\theta_{ij}^{ab} c_a^\dagger c_b^\dagger c_i c_j|\Phi_{\mathrm{HF}}\rangle + \cdots \tag{14}$$

where $|\Phi_{\mathrm{HF}}\rangle$ is the Hartree–Fock state, $i, j, \dots$ refer to occupied molecular orbitals and $a, b\dots$ refer to unoccupied ones (we henceforth include the spin σ index in the orbital index). One usually truncates the expression to a finite set of excitations, for instance single (S) excitations $c_a^\dagger c_i|\Phi_{\mathrm{HF}}\rangle$ and double (D) excitations $c_a^\dagger c_b^\dagger c_i c_j|\Phi_{\mathrm{HF}}\rangle$. The method is efficiently tractable on classical computers: the computation of the variational energy requires the evaluation of terms like $\langle\Phi_{\mathrm{HF}}|c_{j'}^\dagger c_{i'}^\dagger c_{b'} c_{a'} H c_a^\dagger c_b^\dagger c_i c_j|\Phi_{\mathrm{HF}}\rangle$, which is a polynomial computation thanks to $|\Phi_{\mathrm{HF}}\rangle$ being a single Slater determinant (allowing the use of Wick's theorem). However, to make it accurate in strongly-correlated cases, one needs a very large number of excitations (which in practice is too large compared to the required accuracy), unless one uses several Slater determinants in the CI expansion (instead of only the HF state). Even in this "multi-reference" case, (truncated) CI will suffer from a lack of size extensivity, namely the energy of two uncoupled fragments will not be the sum of the energies of each fragment. (The untruncated, and thus exponentially costly, version of CI, called full configuration interaction (FCI), takes into account all Slater determinants: it is of course exact and as such extensive, but untractable beyond small sizes because exponential in N_o.)

Coupled cluster (CC, [34]). CC is a method that allows to use the same number of parameters as CI, but includes an infinite number of fluctuations around the HF state. This is achieved by putting the parameters into an exponential

$$|\Psi(\boldsymbol{\theta})\rangle = \mathrm{e}^{\sum_{a,i}\theta_i^a c_a^\dagger c_i + \sum_{ab,ij}\theta_{ij}^{ab} c_a^\dagger c_b^\dagger c_i c_j + \cdots}|\Phi_{\mathrm{HF}}\rangle \equiv \mathrm{e}^{T(\boldsymbol{\theta})}|\Phi_{\mathrm{HF}}\rangle. \tag{15}$$

Usually the cluster operator $T(\boldsymbol{\theta})$ is truncated to single (S) and double (D) excitations, in which case we still have $O(N_o^4)$ parameters, but by expanding the exponential we see that an infinite number of fluctuations (made up of "clusters" of excitations of finite order) is taken into account. A major advantage of the CC method is that is it size extensive.

Questions arise when implementing the method in practice. In principle, one would like to keep the variational nature of CI (this is formally advantageous as one knows that the obtained energy is an upper bound to the true energy). This entails minimizing

$$E(\boldsymbol{\theta}) = \frac{\langle\Phi_{\mathrm{HF}}|\mathrm{e}^{T^\dagger} H \mathrm{e}^{T}|\Phi_{\mathrm{HF}}\rangle}{\langle\Phi_{\mathrm{HF}}|\mathrm{e}^{T^\dagger}\mathrm{e}^{T}|\Phi_{\mathrm{HF}}\rangle}. \tag{16}$$

However, computing $E(\boldsymbol{\theta})$ is not efficient: the number of terms that arise from expanding the exponentials is factorial in the number of orbitals N_o (when truncating T so a finite excitation order, [35, 36]), and truncating to a smaller arbitrary number of terms would suppress size extensivity [37]. This is why, in most implementations, one does not use this variational formulation of coupled cluster (dubbed VCC).

Instead, one uses the fact that at convergence, one should have $H\mathrm{e}^T|\Phi_{\mathrm{HF}}\rangle = E_0\mathrm{e}^T|\Phi_{\mathrm{HF}}\rangle$. Multiplying by e^{-T}, this yields $\mathrm{e}^{-T}H\mathrm{e}^T|\Phi_{\mathrm{HF}}\rangle = E_0|\Phi_{\mathrm{HF}}\rangle$. Defining states $|\mu\rangle = \tau|\Phi_{\mathrm{HF}}\rangle$ for $\tau \in \{c_a^\dagger c_i, c_a^\dagger c_b^\dagger c_i c_j, \dots\}$, and noticing that these states are orthogonal to $|\Phi_{\mathrm{HF}}\rangle$, we can project:

$$\langle\mu|\mathrm{e}^{-T}H\mathrm{e}^T|\Phi_{\mathrm{HF}}\rangle = 0 \tag{17}$$

for all μ. We are thus facing a root finding problem.

It turns out that the Baker–Campbell–Hausdorff expansion of operator $\mathrm{e}^{-T}H\mathrm{e}^T$ (which is not Hermitian)

$$\mathrm{e}^{-T}H\mathrm{e}^T = H + [H,T] + \tfrac{1}{2!}[[H,T],T] + \tfrac{1}{3!}[[[H,T],T],T] + \cdots \tag{18}$$

terminates at a finite, polynomial order in T (when T itself is truncated to a finite excitation order) because the creation (annihilation) operators act only on unoccupied (occupied) orbitals. Therefore, CC, solved in this fashion (sometimes called projective CC, PCC, or truncated CC, TCC), remains a polynomial method (the cost is e.g. $O(N_o^6)$ for single and double excitations, CCSD).

In its "SD with perturbative triples" form (CCSD(T)), CC is considered to be the golden standard for weakly correlated molecules.

Two main limitations are the fact that it is a single-reference method (it builds upon a single Slater determinant, $|\Phi_{\mathrm{HF}}\rangle$), and it is not, in its projective form, variational (essentially because $\mathrm{e}^{-T}H\mathrm{e}^{T}$ is not hermitian). The first limitation entails inaccurate results e.g. in the dissociation limit of molecules. Whether the second limitation—non variationality—is an actual problem is a debated topic (see e.g. [38]), namely it is not sure that guaranteeing that $E(\boldsymbol{\theta})$ is an upper bound to E_0 is very important in practice. Yet, VCC does have the nice property to ensure that a generalized Hellmann–Feynman theorem holds [39] (the Hellmann–Feynman theorem states that if r is an external parameter (e.g. the bond distance), then $\partial_r E(\boldsymbol{\theta}) = \langle\Psi(\boldsymbol{\theta})|\partial_r H|\Psi(\boldsymbol{\theta})\rangle$, making it easy to compute forces in molecules without redoing the variational computation for several r's). Note that PCC also satisfies the Hellmann–Feynman theorem [40].

Unitary coupled cluster (UCC, [37]). UCC proposes an ansatz of the form:

$$|\Psi(\boldsymbol{\theta})\rangle = \mathrm{e}^{T(\boldsymbol{\theta})-T^{\dagger}(\boldsymbol{\theta})}|\Phi_{\mathrm{HF}}\rangle. \tag{19}$$

This expression, as VCC, has only a polynomial number of parameters, and preserves size extensivity, and can be implemented in a variational way. In practice, as in VCC, computing the corresponding energy is combinatorially difficult in the absence of truncation of the BCH expansion. The relative qualities of VCC and UCC are still a subject of discussion [36]. However, one interesting feature of UCC is the unitary nature of $\mathrm{e}^{T(\boldsymbol{\theta})-T^{\dagger}(\boldsymbol{\theta})}$: as we shall see later, it will make UCC easier to implement on a quantum processor.

All three methods, when applied to the HF state, are limited to weak correlations (around the reference). They typically fail in strongly-correlated regimes. Other methods, referred to as multireference methods, are necessary to handle the strongly-correlated regime properly (we can mention the complete active space self-consistent field method (CASSCF, [41, 42]), configuration interaction using a perturbative selection done iteratively (CIPSI, [43]), multi-reference CC). We refer the reader to [44] for a more comprehensive analysis of quantum chemical methods.

2.2. *Problem reduction: embedding approaches*

2.2.1. *Dynamical mean-field theory…*

In the previous section, we summarized the main methods used for the numerical treatment of strongly-correlated problems, with a focus on the Hubbard model. All these methods have limitations in some regimes. One key aspect of the Hubbard model (and of electronic structure problems in general, see [44] for a discussion) that we overlooked so far is the *locality* of the physics of interacting electrons (which is also the reason for taking the Hubbard model as a starting point).

Advanced classical methods were developed starting in the early 1990s to take advantage of this locality. In particular, the local nature of the Mott phenomenon at play in the Hubbard model (Equation 2) is central: at large interaction strengths, local interaction phenomena cause the charge degree of freedom to freeze and electrons to localize (this is the essence of Kondo physics). Based on this physical insight, and on more formal considerations about the infinite-dimensional limit of the Hubbard model, a theory called dynamical mean field theory (DMFT, [45]) was

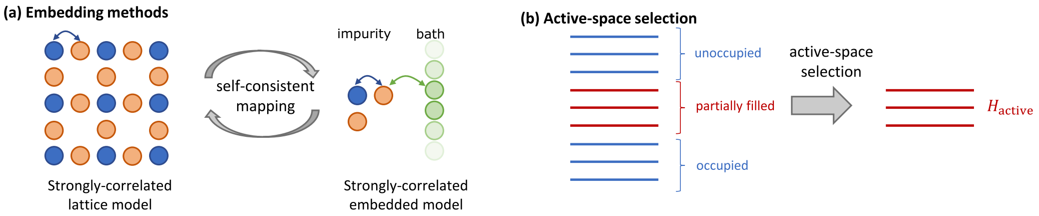

Figure 2. Embedding methods (a) and active space selection (b). Adapted from [46]. With kind permission of The European Physical Journal (EPJ).

developed to self-consistently map the Hubbard model onto a local effective model describing a single or a few interacting sites (usually called "impurities") embedded in a noninteracting bath mimicking the dynamical mean field surrounding them (Figure 2).

In analogy to plain-vanilla Weiss mean-field theories [47]—where a single spin S feels the influence of a mean field h created by its neighbors, which itself depends on the mean magnetization $\langle S \rangle$ of the spin—DMFT describes a single (for simplicity) impurity embedded in a (dynamical) mean field $\Delta(t)$, called the hybridization function. This mean field itself depends on the impurity's Green's function, defined as

$$G^{\mathrm{R}}(t) = -\mathrm{i}\theta(t)\langle\{c(t), c^{\dagger}(0)\}\rangle. \tag{20}$$

The dynamical character of the mean field $\Delta(t)$ is crucial in recovering Mott physics: it describes how surrounding electrons can hop on and off the impurity site, where the Hubbard interaction U can cause them to localize. This contrasts with the Hartree–Fock mean-field theory described in a previous section, where the effect of other electrons on the effective model is described by a static density $\langle c^{\dagger}_{i\bar{\sigma}} c_{i\bar{\sigma}} \rangle$ (Equation (5)). Essentially, DMFT replaces this static mean field with a dynamical quantity like $\langle c_{i\sigma}(t) c^{\dagger}_{i\sigma}(0) \rangle$.

Through this mapping, the Hubbard model gets mapped onto a so-called Anderson impurity model (AIM), defined by the Hamiltonian

$$H = U n_{\uparrow} n_{\downarrow} - \mu(n_{\uparrow} + n_{\downarrow}) + \sum_{p\sigma} V_{p\sigma}(c^{\dagger}_{\sigma} a_{p\sigma} + \mathrm{h.c}) + \sum_{p\sigma} \epsilon_{p\sigma} a^{\dagger}_{p\sigma} a_{p\sigma}, \tag{21}$$

where the impurity site $(c^{\dagger}_{\sigma}, c_{\sigma})$ is coupled to noninteracting bath sites $a^{\dagger}_{p\sigma}, a_{p\sigma}$ (represented as the green dots in Figure 2). The (a priori infinitely many) coupling parameters $V_{p\sigma}$ and bath levels $\epsilon_{p\sigma}$ are chosen so that the Fourier transform of $\Delta(t)$ verifies:

$$\Delta_{\sigma}(\omega) = \sum_{p=1}^{\infty} \frac{V_{p\sigma}}{\omega - \epsilon_{p\sigma} + \mathrm{i}\eta}. \tag{22}$$

In practice, this Anderson impurity model is simpler to solve than the Hubbard model, because only the impurity site (or the impurity sites in "cluster" DMFT) is interacting. A whole spectrum of "impurity solvers" has been developed to compute the impurity Green's function required by the self-consistent DMFT equations. Most of these solvers rely on techniques similar to those used for tackling the Hubbard model (see Section 2.1). However, because the interacting problem is of much lower dimension (even zero dimensional in the single-impurity case) compared to the Hubbard model, the exponential hurdles cause much less harm, and allow for a very accurate computation of $G(\omega)$ in many physical regimes. In particular, they allow to solve DMFT with a few impurities and observe a Mott transition, a major achievement compared to usual mean-field theories.

There are, however, three main limiting factors: imaginary time, the number of impurities, and out-of-equilibrium regimes. First, most advanced quantum Monte-Carlo impurity solvers [48] work on the imaginary-time axis, which requires mathematically ill-defined analytical continuation techniques to obtain real-time quantities. Second, increasing the number of impurities (or orbitals, when working with more realistic models than the Hubbard model) is crucial in regimes where long-range fluctuations play an important role (as is suspected in some high-temperature superconductors), and serves as a control parameter of the method. But larger sizes revive the aforementioned exponential issues. Third, going out of equilibrium comes with the same problems as for the Hubbard model, albeit at a smaller scale to thanks to the reduced dimension of the impurity model [49].

2.2.2. *… and its simplifications*

To circumvent these limitations or to tackle larger (usually multiorbital) problems, simplified methods have been developed at the price of less accuracy or less information. Similarly to DMFT, these alternative "embedding methods" also consist in self-consistently mapping strongly-correlated, extended models like the Hubbard model onto smaller effective models. One such method is rotationally-invariant slave bosons (RISB, [50]), equivalent to the Gutzwiller method [51], which can be regarded as a low-energy simplification of DMFT that gives access to low-energy properties of the spectrum like the quasiparticle renormalization weight Z and the static self-energy shift.

In RISB, the impurity model Equation (21) contains as many bath sites as correlated sites (or impurities) [52], as opposed to the infinite (or very large) number required in DMFT to mimic the dynamical features of the hybridization function $\Delta(\omega)$. The parameters are obtained via self-consistent equations that are similar to those of DMFT (see e.g. [53] for a unified view of DMFT, RISB and the DMET method below). Contrary to DMFT, RISB does not require the computation of the full time-dependent impurity Green's function, but only of static correlators of impurity sites $i, j\dots$ $[D_{\mathrm{imp}}]^{\sigma}_{ij} = \langle c^{\dagger}_{i\sigma} c_{j\sigma}\rangle$, bath sites $[D_{\mathrm{bath}}]^{\sigma}_{pq} = \langle a^{\dagger}_{p\sigma} a_{q\sigma}\rangle$, as well as mixed correlators $[D_{\mathrm{mixed}}]^{\sigma}_{ip} = \langle c^{\dagger}_{i\sigma} a_{p\sigma}\rangle$. Together, these correlators make up the impurity model's one-particle reduced density matrix

$$D = \begin{bmatrix} D_{\mathrm{imp}} & D_{\mathrm{mixed}} \\ D^{\dagger}_{\mathrm{mixed}} & D_{\mathrm{bath}} \end{bmatrix}. \tag{23}$$

These static correlators are easier to compute than the full Green's function (Equation (20)), in exchange for containing less information. RISB has been used, for instance, to explore non-local correlation properties (like momentum-dependent quasiparticle weights) in the Hubbard model [54], or multi-orbital models inaccessible to DMFT relevant to realistic materials like praseodymium and plutonium [52], at a fraction of the cost of DMFT.

Another widespread embedding method, density-matrix embedding theory (DMET, [55]), proposes a similar embedding as RISB, without giving access to information such as the quasiparticle weight [53]. It also relies on the mapping of an extended model onto a simplified effective model that is solved with more accurate methods than mean-field methods.

The embedding methods presented above all consist in iteratively defining a suitable effective model. The definition of the model itself is done thanks to an efficient (i.e. polynomial) computation, while the effective model, which is supposed to contain the relevant strongly-correlated degrees of freedom, is solved either exactly with an expensive (exponential) method—with the corresponding size or time limitations—or a simplified polynomial method—with the corresponding loss of accuracy.

The spirit of these embedding methods is quite similar to "active space" methods encountered in a quantum chemistry context: there, the important degrees of freedom, called the "active space" of the molecule, are selected among all the orbital degrees of freedom of the molecule. Indeed, tackling all the degrees of freedom would either be impossible for exact methods (like the full configuration interaction method) due to the exponential wall, or inaccurate when

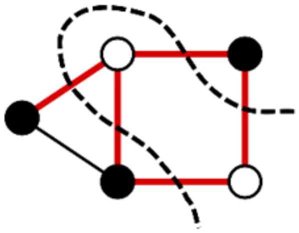

Figure 3. Example of graph. The maxcut bipartition is illustrated with the black and white vertices.

implemented with approximate polynomial methods like the configuration interaction (CI) or coupled cluster (CC) method with a finite number of excitations (like singles and doubles) for strongly-correlated (also called “multireference”) compounds (see Section 2.1.6 above).

In conclusion: with embedding methods and active-space methods, an efficient classical computation is done to (usually self-consistently) identify the relevant, hard degrees of freedom of the original extended model. This construction is very successful in reducing the problem size and therefore making it more accessible to exact, exponentially costly methods. Yet, they carry their own limitation as the size of the effective model is by construction limited by the exponential wall. For some regimes, the effective-model sizes that are required to converge the computation exceed the capacities of classical processors. In an upcoming section, we will explain how quantum processors could be used as a replacement or complement to some of the classical algorithms used for tackling these strongly-correlated problems.

Before this, we also introduce many-body problems that are not coming from physical systems.

2.3. *Beyond quantum: classical many-body problems*

Quantum many-body problems such as the ones that naturally arise in condensed-matter physics and quantum chemistry are but a subclass of many-body physics. Beyond the field of nuclear physics (which we did not elaborate on but shares many problems and methods with solid-state physics and chemistry, see our recent review, [56]), many-body problems are also encountered in classical computational problems known as combinatorial optimization problems.

One well-known combinatorial optimization problem is the maximum cut (MaxCut) problem. It is defined on a graph $G = (V, E)$ with a set V of vertices and a set E of edges. It consists in finding the “maximum cut” of the graph, namely the bipartition of vertices such that the most edges in E have one vertex in either partition. An example of graph and its maxcut bipartition are given in Figure 3.

If one labels the vertices $i = 1, \dots, n$, then a bipartition of the vertices can be encoded as a bitstring $b = (b_1, b_2, \dots, b_n)$ with $b_i \in \{0, 1\}$. If $b_i = 0$, the ith vertex belong to the first partition, otherwise it belongs to the second partition. Among the 2^n possible bipartitions (bitstrings), we want to find the one that maximizes

$$C(b) = -\sum_{i,j \in E} (1-2b_i)(1-2b_j). \tag{24}$$

Indeed, if $b_i = b_j$, we obtain a negative contribution -1 for each edge, otherwise (when the vertices belong to different partitions) we obtain a positive $+1$ contribution. Maximizing $C(b)$ should thus yield the solution.

Solving MaxCut on a classical computer is known to be exponentially hard (it is a NP complete problem, see Section 4.3.1 below for a more in-depth discussion). The connection to usual

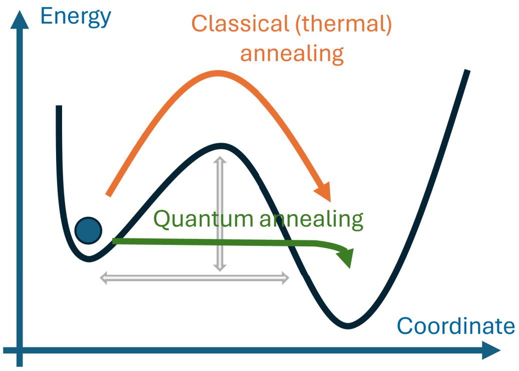

Figure 4. Sketch of the potential landscape and different annealing methods.

many-body problems can be made by redefining $S_i = 1-2b_i$. The S_i variables can take the values $+1$ or -1, they are like classical spins. The cost function, with an additional minus sign, is called the energy and reads:

$$E(S) = \sum_{ij\in E} S_i S_j = \sum_{ij} J_{ij} S_i S_j. \tag{25}$$

This form of energy defines the (antiferromagnetic) Ising model. Here, the "coupling" is $J_{ij} = 1$ if i, j is an edge of G and 0 otherwise. The Ising model belongs to the class of spin glass models. A well-known classical method to find the lowest energy configuration S^* of this model is to use a Markov chain Monte-Carlo algorithm with a Boltzmann distribution $p(S) \propto \mathrm{e}^{-E(S)/T}$ for the configurations, where the temperature T is slowly decreased (or "annealed") to 0. At this point, the system should be in its lowest configuration. The intuition behind the algorithm is that the nonzero temperature at the beginning is going to help the algorithm out of the local energy minima it can start in. This algorithm, called simulated annealing (SA) or thermal annealing, is illustrated in Figure 4. Of course, if the potential landscape is very rugged (with high barriers), the probability for attaining the global minimum will be very small.

One way to enhance this probability would be to create the possibility of jump through barriers, as sketched in Figure 4. For this, one needs to supplement our dynamics with possibilities to "hop" from one configuration to another. To achieve this, we turn our classical cost function into a quantum one: if we define the operator

$$H = \sum_{ij} J_{ij} Z_i \otimes Z_j, \tag{26}$$

with Z_i the Pauli-z matrix on the ith spin, we can check that the lowest energy of this operator corresponds to the bitstring S that minimizes E. We can now add a temporary perturbation to H that couples different eigenstates of H with each other, say, by defining

$$H(t) = \sum_{ij} J_{ij} Z_i \otimes Z_j + h(t) \sum_i X_i, \tag{27}$$

(with X_i the Pauli-x matrix). The second term (a transverse field term) accomplishes the wished-for hopping between configurations. As we shall argue in Section 3.1.2, a slow tuning of the coefficient of h from a large value to zero will lead to an enhanced probability of ending up in the ground state of H. This is the idea behind quantum annealing (QA). One can expect this algorithm to provide speedups when the potential barriers are tall and narrow.

While this quantum algorithm can be, and is, executed on the quantum processors we will introduce in the next section, it can also be "emulated" on classical computers. The corresponding "quantum-inspired" method is called "simulated quantum annealing" (SQA): the quantum time evolution can be mapped to a classical Ising model with one more dimension, which is solved using a Markov chain Monte-Carlo algorithm similar to the one used for simulated annealing. Such an algorithm turns out not to have a sign problem [57], contrary to fermionic problems, but its convergence to the exact ground state can be very slow. In general, however, it is faster than the SA algorithm [58]. Whether SQA is faster than a purely quantum algorithm (QA) is still a matter of debate (see [58] for a study in favor of SQA, [59] for a study in favor of QA).

In the next section, we introduce in more detail the main principles of quantum algorithms for many-body problems.

3. Quantum computers: promises for the many-body problem ... and reality

In this section, we summarize the reasons why quantum processors have been suggested as potential solutions to the limitations of the classical algorithms for solving the strongly-correlated problems we introduced in the previous section. We also discuss the limitations of these quantum processors, whether coming from the quantum nature of the processors or from hardware imperfections.

3.1. *The promises of the perfect quantum computer*

Given the challenges of solving many-body problems with classical processors, Richard Feynman famously advocated the use of quantum machines to simulate quantum problems [60]. In other words, he suggested the use of physical devices governed by the laws of quantum mechanics to mimic natural physical systems governed by the same laws.

3.1.1. *Two classes of quantum processors*

Two main strains of quantum processors emerged in the following decades. Inspired by Feynman's suggestion, physicists built "quantum simulators", namely artificial systems—ultracold atoms in optical lattices [61], ions trapped in electromagnetic traps, Rydberg atoms trapped by optical tweezers [62], etc.—whose Hamiltonian was similar to that of physical systems of interest. For instance, Fermi–Hubbard physics were studied via the physics of tens of ultracold atoms in optical lattices (see e.g. [63]), and hundreds of Rydberg atoms have been used to gain insights into the dynamics of spin models (see e.g. [64]). This "quantum simulation" approach is currently limited, among others, by the effective temperatures that can be reached, which are still too high to study phenomena like pseudogap physics, let alone superconductivity (they typically reach temperatures close to the antiferromagnetic exchange coupling [65]). In addition, these processors are limited to simulating a specific form of Hamiltonian, and are therefore meant to "solve" very specific problems. They are sometimes called "analog quantum computers", in reference to the analog computers that were built in the past to solve dedicated problems, before the advent of (classical) computers as we know them.

The second class of computers is inspired by the model of computation used in classical computers: there, all operations can be boiled down to logic gates (NOT, NAND, etc) acting on bit registers. This model was extended to quantum processors by introducing the notion of quantum bit (qubit) to refer to a spin-1/2 system. A quantum "computer" can then be regarded as a collection of interacting qubits. These qubits can be manipulated by quantum logic gates, which, owing to Schrödinger's equation, are unitary operations acting on (possibly several) qubits. The so-obtained quantum states are then "measured" to extract useful information.

It was soon realized that this mathematical model of quantum computation (sometimes called gate-based or "*digital* quantum computation") could be used to create algorithms with provable speedups over classical algorithms. The two most famous examples are Shor's factoring algorithm [66] and Grover's unstructured search algorithm [67], which respectively achieve an exponential and a quadratic speedup over their known classical counterparts. Another interesting feature of reusing a formalism similar to that of classical computers is that the error correction methods developed for classical machines were generalized to quantum computers [68, 69]: by using several ("physical") qubits to encode the information of a single ("logical") qubit, it was proven that, provided the error rate of physical qubits was below a certain threshold, combining more and more physical qubits together led to an exponential suppression of errors for the corresponding logical qubits, thereby guaranteeing an arbitrary level of accuracy. To date, the required error thresholds and numbers of qubits have not yet been attained by experimental quantum processors, although first promising proofs of concepts in this direction—with near-threshold errors, but still very small qubit counts—are emerging in all major technological platforms [70–73].

3.1.2. *Preparing a ground state: adiabatic annealing*

One major challenge of quantum many-body physics is to investigate the properties of the ground state of a complicated Hamiltonian. For this, one needs to generate such a state. A central method to generate such a state is the quantum adiabatic annealing method (see [74] for a review). It consists, in order to prepare the ground state $|\Psi_0\rangle$ of a Hamiltonian H, in starting from an easier Hamiltonian H_0, whose ground state $|\Phi_0\rangle$ is known and easy to prepare on the quantum hardware at hand. Then, one performs a time evolution, starting from $|\Phi_0\rangle$, under a Hamiltonian that interpolates between H_0 and H between the initial time and the final time T. For instance,

$$H(t) = (1 - t/T)H_0 + t/TH. \tag{28}$$

If the "annealing time" T is long enough ($T \gg V/\Delta_{\mathrm{min}}^2$ with V the time derivative of $H(t)$ in its instantaneous eigenstate and Δ_{min} the minimum gap in the spectrum of $H(t)$), the adiabatic theorem will guarantee that $|\Psi(T)\rangle$ will be very close to $|\Psi_0\rangle$. The overall run time of the algorithm is $O(T)$, so a major question for quantum annealing is the scaling of the minimum gap Δ_{min}. It depends on the problem at hand (H) as well as the mixer Hamiltonian (H_0) and the schedule itself. In particular, hard problems may feature a minimum gap that decreases exponentially with the system's size… leading to exponentially long run times.

The adiabatic annealing principle is underlying a number of quantum algorithms and methods. It is routinely used to prepare Rydberg atoms in the ground state of exotic Hamiltonians [75]. Quantum annealers built e.g. by the D-wave firm use this principle, in combination to the thermal annealing method we described above, to attempt to find ground states of Hamiltonians that correspond to complex optimization functions, as briefly discussed in Section 2.3.

Adiabatic annealing is also a guiding principle to design quantum programs on gate-based architecture. Since annealing consists in performing time evolutions governed by Hamiltonians like $H(t)$ (Equation (28)), tools have been introduced to implement these time evolutions with gate-based quantum computers, as we will describe in the next subsection.

3.1.3. *Performing a Hamiltonian evolution*

The time evolution of quantum systems is the first inspiration behind Feynman's idea [60]. While the first—analog—class of computers can be used to simulate the system they are built to reproduce, the second—digital—class of computers can also in principle be used as a substitute to classical computers for simulating *any* many-body problem. How to do it in practice was first described by [76]. The main building block for doing so is the realization that a quantum time evolution under the Schrödinger equation, which can be boiled down to a unitary evolution under the evolution operator $U = \mathrm{e}^{-\mathrm{i}Ht}$ (in the time-independent Hamiltonian case; the time-dependent case can also be dealt with, but we will focus on the time-independent case for ease of notation), can be expressed as a quantum circuit, namely a sequence of quantum logic gates.

The goal is to break down the $\mathrm{e}^{-\mathrm{i}Ht}$ unitary operator to the quantum logic gates of the quantum processor at hand. Formally, this breakdown is guaranteed to be feasible by the Solovay–Kitaev theorem. It states that any unitary operator can be approximated to precision ϵ by a sequence of gates of size polylogarithmic in $1/\epsilon$, namely a short sequence. These gates can be drawn from a universal gateset known as "Clifford+T"—namely one-qubit Clifford gates (essentially, single-qubit $\pi/2$ rotations) and the controlled-NOT (CNOT) gate—supplemented with the so-called T gate (a $\pi/4$ rotation along the Z axis).

In practice, the breakdown can be achieved by several methods. The simplest one [76] is to use the Trotter–Suzuki formula: supposing the Hamiltonian of interest has been decomposed as a weighted sum of Pauli operators,

$$H = \sum_{i=1}^{N_p} \lambda_i P_i, \tag{29}$$

(with $P_i = \sigma_1^{p_i(1)} \otimes \sigma_2^{p_i(2)} \otimes \cdots \otimes \sigma_n^{p_i(n)}$, $\sigma^p \in \{I, X, Y, Z\}$, $\lambda_i \in \mathbb{R}$) the first-order product formula reads

$$U = \mathrm{e}^{-\mathrm{i}Ht} = \prod_{k=1}^{N_t} \left(\prod_{i=1}^{N_p} \mathrm{e}^{-\mathrm{i}\frac{\lambda_i}{N_t} P_i t} \right) + O\left(\frac{t^2}{N_t} \right), \tag{30}$$

with N_t the number of time-slices (higher-order formulae and precise bounds involving the commutators of the P_i's are discussed in [77]).

Usually, the Pauli operators P_i that appear in the decomposition of physical Hamiltonians are local, namely they act only on a small subspace of the full Hilbert space. Thus, the operator $\mathrm{e}^{-\mathrm{i}\lambda_i/N_t P_i t}$ that appears in (30) can typically be decomposed as a short sequence of one- and two-qubit gates. Thus, the original unitary evolution operator U has been broken down, up to a controllable error $\epsilon \propto t^2/N_t$, as a sequence of one- and two-qubit gates that form a quantum circuit.

The ideal computational complexity for performing the corresponding computation is $O(N_t N_p N_c) \sim O(t^2/\epsilon)$. The scaling in the error ϵ is of the form poly$(1/\epsilon)$, a scaling regarded as "tractable".

The final relevant term is N_p, the number of Pauli terms, which we need to assess as a function of the system size. Typically, for spin Hamiltonians on a lattice, like the Ising (see Equation (25)), XY or Heisenberg models, N_p is proportional to the system size n. For fermionic models and in particular for the Hubbard model introduced previously (Equation (2)), we first need to transform the fermionic Hamiltonian in the form of Equation (29). There are several such fermion-qubit transformations or "encodings". A straightforward transformation is the Jordan–Wigner transformation, which maps the creation and annihilation operators onto Pauli spin operators as follows:

$$c_\alpha^\dagger = \prod_{\beta=1}^{\alpha-1} Z_\beta \left(\frac{X_\alpha - \mathrm{i}Y_\alpha}{2} \right), \quad c_\alpha = \prod_{\beta=1}^{\alpha-1} Z_\beta \left(\frac{X_\alpha + \mathrm{i}Y_\alpha}{2} \right). \tag{31}$$

The operator $X_\alpha - \mathrm{i}Y_\alpha/2 = \sigma_\alpha^+$ flips the αth qubit (corresponding to the αth spin-orbital), while the string of Z operators takes care of fermionic anticommutation relations. Thus e.g. a hopping term $c_\alpha^\dagger c_{\alpha'}$ maps to 4 Pauli strings of the form $\sigma_\alpha Z_{\alpha+1} \cdots Z_{\alpha'-1} \sigma_{\alpha'}$ (with $\sigma_{\alpha/\alpha'} \in \{X, Y\}$). Since the Hubbard model has $O(n)$ terms, the Jordan–Wigner transformation will lead to a number of Pauli terms $N_p = O(n)$. For the general electronic structure Hamiltonian (Equation (1)) used to model quantum chemical systems, N_p is dominated by the Coulomb interaction term, $N_p = O(N_o^4)$ (with N_o the number of orbitals).

Thus, generically, for physical systems written in a second-quantized form, $N_p \sim \text{poly}(n)$ where n is the size of the system. This gives a total scaling $O(t^2/\epsilon\text{poly}(n))$. Interestingly, replacing the first-order Trotter formula with a pth-order formula will lead to a scaling $O(t^{(1+p)/p}/\epsilon^{1/p}\text{poly}(n))$, which, in the limit of $p \to \infty$, becomes $O(t^{1+o(1)}/\epsilon^{o(1)}\text{poly}(n))$ (see e.g. [78, chapter 5]), namely in the absence of Trotter errors we get a linear run time, as expected.

This scaling, which is behind Feynman's original idea [60], is very favorable to quantum computers when compared to the exponential scaling (in size and/or time and/or other parameters) of the classical methods described above. This exponential advantage is the very reason for the strong push of quantum computing in the last decades. Of course, much remains to be done to improve the implementation of the unitary at hand. While the understanding of Trotter formulas and their errors is making progress (see e.g. [77]), alternatives to Trotter formulas have appeared in recent years (see e.g. [79–82]).

Let us stress that while we introduced Hamiltonian simulation as a means to implement the slow time evolution required in adiabatic annealing, it of course applies to any time evolution, like a sudden change (quench) of a parameter of a Hamiltonian, which leads to time evolutions that are hard to describe on classical computers.

3.1.4. *Computing ground state energies: quantum phase estimation*

With the adiabatic annealing algorithm and a circuit to implement time evolution, we can in principle prepare a quantum processor in the ground state of a given Hamiltonian. We are now facing the problem of measuring the energy of this state, or other observables like the many-particle correlators required e.g. in embedding techniques.

The energy estimation problem. Measurements in quantum processors are usually limited to measurements of Pauli operators Z_i on each qubit i. With these measurements, one can easily estimate expectation values of the form $\langle Z_{i_1} \cdots Z_{i_m} \rangle$ or, via single-qubit rotations, the expectation value of any string of Pauli operators. However, this expectation value estimation—which, at face value, is needed to compute the energy $\langle \Psi_0 | H | \Psi_0 \rangle$ of the state we prepared—is plagued with the same statistical variance as the classical Monte-Carlo algorithms we introduced previously (see Equation (9)). Indeed, it consists in approximating an expectation value with an empirical average on a finite number of samples. (This will turn out to be a major limitation of the near-term algorithms we will introduce later).

A key quantum algorithm, called quantum phase estimation (QPE, [83], that is also, incidentally, at the heart of the aforementioned Shor algorithm), avoids this statistical error by resorting to a circuit inspired by Mach–Zehnder interferometry [84].

QPE consists in extracting the phase φ of the eigenstate $|\psi\rangle$ of a unitary operator U such that $U|\psi\rangle = e^{i\varphi}|\psi\rangle$, given a circuit implementation of U. Applied to $U = e^{-iHt}$ and an eigenvector $|\psi\rangle$ of H, QPE allows to extract the eigenenergy of this eigenstate. This is achieved with an accuracy $\epsilon \sim 1/2^m$, where m is the number of auxiliary (or ancilla) qubits used in addition to those needed to encode $|\psi\rangle$. This comes at a computational cost of $O(2^m)$ (QPE requires the successive application of controlled U^{2^k} operators with $k = 0 \ldots m-1$, hence the $O(2^m)$ cost). QPE thus achieves a time complexity of $O(1/\epsilon)$, a quadratic gain over the time complexities of $O(1/\epsilon^2)$ that Monte-Carlo algorithms must cope with (recall Equation (9): a set error $\Delta g = \epsilon$ leads to a number of samples, and hence a run time, $\mathcal{N} \sim 1/\epsilon^2$).

QPE also comes with the useful property of projecting any incoming vector $|\phi\rangle$ to an eigenvector $|\psi\rangle$ of U with probability $P \propto \Omega = |\langle \phi | \psi \rangle|^2$. This can also be regarded as a downside: the success probability of the algorithm is proportional to Ω, the overlap with the sought-after eigenstate. Therefore, a number $1/\Omega$ of repetitions of QPE is needed to find the phase with high probability.

Hence, QPE can be used to prepare ground states and find ground state energies of Hamiltonians such as the Hubbard model: starting from a guess $|\phi\rangle$ for the ground state of H with sufficient overlap with the true ground state $|\psi_0\rangle$, one can in principle extract the ground state energy E_0 to great accuracy. The U^{2^k} operators can be implemented with the Trotter–Suzuki technique we just described for time evolution.

The preparation of an input state sufficiently close to the ground state is a key challenge. The simplest way is to find a classical method (such as the ones we reviewed in Section 2) to prepare an input state and then load it onto the quantum computer (how to do so depends on the classical representation of the state, and can be expensive). Another, more quantum way, is to use QPE in an annealing-like iterative procedure, dubbed adiabatic state preparation (ASP, [85]): one applies several QPEs with H interpolating, as in Equation (28), from an easy Hamiltonian H_0 with known ground state $|\Phi_0\rangle$ (e.g. $H_{\mathrm{m.f}}$, Equation (4)) to the Hamiltonian of interest H. As per the adiabatic theorem, the number of required steps for this ASP is of the order of $O(\mathrm{poly}(1/\Delta_{\mathrm{min}}))$, where Δ_{min} is the minimal gap between the ground state and the first excited state of the Hamiltonian that interpolates between H_0 and H [74]. The behavior of this gap with system size is difficult to predict.

Taking the first approach (using a classical heuristic), the complexity of QPE can thus be estimated by looking at the overlaps achieved by known methods in cases where the exact ground state is known. Reference [86] shows that, as expected, for weakly-correlated molecules (aka single-reference or dynamically correlated systems), the Hartree–Fock method does yield states with good overlaps with the true ground state. For more correlated systems, like the Hubbard and Anderson models with larger U/t ratios and small fillings, the overlaps become quite small ([86], see also [87]).

Computing other observables. Computing other observables, like the correlators needed for DMET or RISB (Equation (23)), or the Green's function for DMFT (Equation (20)), boils down (at zero temperature) to computing quantities of the form $\langle\psi|U_{\mathrm{meas}}|\psi\rangle$, with $|\psi\rangle$ e.g. the ground state wavefunction (typically prepared with the QPE-ASP procedure described above), and U_{meas} a unitary operator.

For instance, after a Jordan–Wigner transformation, a local $c_i(t)c_i^\dagger(0)$ term appearing in the Green's function becomes

$$\begin{aligned}U^\dagger(t)\frac{X_i+\mathrm{i}Y_i}{2}U(t)\frac{X_i-\mathrm{i}Y_i}{2} &= \frac{1}{4}\{U^\dagger(t)X_iU(t)X_i-\mathrm{i}U^\dagger(t)X_iU(t)Y_i\\ &\quad +\mathrm{i}U^\dagger(t)Y_iU(t)X_i+U^\dagger(t)Y_iU(t)Y_i\},\end{aligned}$$

where $U(t)=\mathrm{e}^{-\mathrm{i}Ht}$ denotes the time-evolution operator. Thus, computing $\langle\psi|c_i(t)c_i^\dagger(0)|\psi\rangle$ amounts to computing four expectation values. For instance, the first term is $\langle\psi|U^\dagger(t)X_iU(t)X_i|\psi\rangle$: it is of the form $\langle\psi|U_{\mathrm{meas}}|\psi\rangle$ and can be computed with two circuits (one for the real part, another for the imaginary part), shown in Figure 5. In addition to the state-preparation circuit needed to prepare the initial state $|\psi\rangle$, one must also implement a Trotterized version of the evolution operator $U(t)$, as well as controlled-Pauli operations.

Using these techniques, one can thus, at least in theory, hope to take advantage of quantum computers to overcome the exponential hurdles that classical methods all face. This was proposed in the DMFT context in [88–90].

However, in practice, the circuits involved in these tasks are potentially quite deep: the controls on the time evolutions lead to numerous entangling gates, and a large number of Trotter slices—needed to maintain a small Trotter error—leads to a high number of gates. For instance, the current noise levels make QPE-based approaches impractical. To obtain the ground state of the Hubbard model to accuracy $\Delta E/E_0 \leq 0.5\%$ for models up to 512 sites (16×16 lattice), the required number of T gates is of the order of 10^7–10^8 according to the estimates of [91].

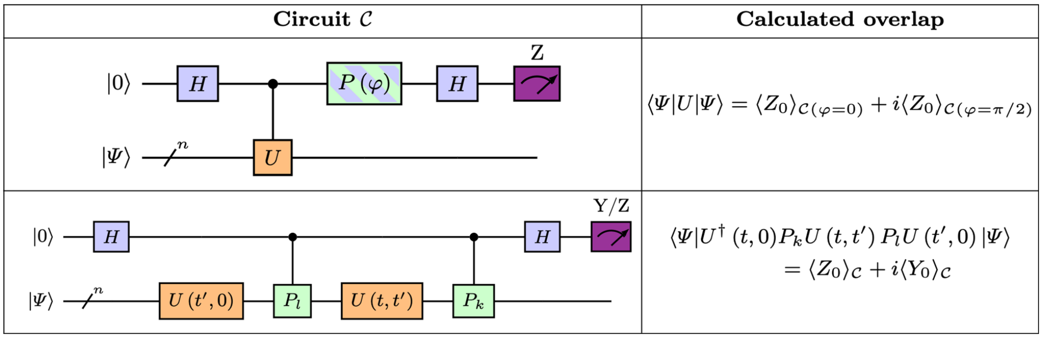
Circuit $\mathcal{C}$
Calculated overlap
$|0\rangle$
H
$P(\varphi)$
H
Z
$|\Psi\rangle$
n
U
$\langle\Psi|U|\Psi\rangle = \langle Z_0\rangle_{\mathcal{C}(\varphi=0)} + i\langle Z_0\rangle_{\mathcal{C}(\varphi=\pi/2)}$
$|0\rangle$
H
H
Y/Z
$|\Psi\rangle$
n
$U(t',0)$
P_l
$U(t,t')$
P_k
$\langle\Psi|U^\dagger(t,0)P_kU(t,t')P_lU(t',0)|\Psi\rangle$
$= \langle Z_0\rangle_{\mathcal{C}} + i\langle Y_0\rangle_{\mathcal{C}}$

Figure 5. Circuits for computing a Green's function. Top panel: circuit for computing $\langle\psi|U|\psi\rangle$. Bottom panel: circuit for computing $\langle\psi|P_k(t)P_l(t')|\psi\rangle$. Taken from [46]. With kind permission of The European Physical Journal (EPJ).

Using a surface code (a quantum error correction code suitable to superconducting architectures, see [92]), assuming a physical gate error rate of 0.1%, this requires about one million physical qubits and about half an hour of computation.

3.2. *Reality: noisy, intermediate-scale quantum computers*

3.2.1. *The exponential effect of decoherence*

Current quantum processors are far from perfect. They suffer from various imperfections: initialization errors, gate errors, readout errors, and idling errors (errors incurred by inactive qubits), to cite the most prominent sources of errors. Gate errors may come from miscalibration (operating a Rabi oscillation on a qubit requires exciting the qubit at its nominal frequency; if this frequency is known only approximately and/or varies with time, a systematic under- or overrotation will occur) or unwanted interaction with the environment.

Mathematically, noisy quantum states (dubbed mixed states) are described by density matrices, usually denoted as ρ, as opposed to the pure states $|\Psi\rangle$ we have used so far. The time evolution is no longer described by a unitary operator U, but by a *quantum channel*, namely a completely positive, trace-preserving linear map $\mathscr{E}(\rho)$. A unitary gate U is described by the channel $\mathscr{U}(\rho) = U\rho U^\dagger$, while generic noisy gates are described by

$$\mathscr{E}(\rho) = \sum_k E_k\rho E_k^\dagger,$$

with the Kraus operators E_k satisfying $\sum_k E_k^\dagger E_k = I$ to preserve the trace. A typical figure of merit of quantum gate is the average fidelity of the corresponding quantum channel, defined as

$$F(\mathscr{E},U) = \int \mathrm{d}\psi\langle\psi|\mathscr{U}^{-1}\circ\mathscr{E}(|\psi\rangle\langle\psi|)|\psi\rangle, \tag{32}$$

where $\mathrm{d}\psi$ denotes the Haar measure on quantum states and $\mathscr{U}$ denotes the ideal unitary operation of the gate. Error rates are defined as $\epsilon = 1-F$. (Similar metrics exist for qubit initialization and readout).

Typically, in today's superconducting processors [93, Supplementary Figure 4], readout errors of the order of 1%, two-qubit gate errors of the order of 0.5%, and one-qubit gate errors of the order of 0.1% are typically measured using various benchmarking protocols.

The cumulated effect of errors on the final state or final measurements is generically exponential in the number of operations. To understand this, we can consider a simple quantum channel called the (global) depolarizing channel,

$$\mathscr{E}_{\text{depol}}(\rho) = (1-p)\rho + p\frac{I}{2^n}. \tag{33}$$

Here the depolarizing probability p tells one how often the state ρ becomes "completely mixed". Upon applying this channel after each of the N_g gates U_k of a circuit, starting from a state ρ, one obtains

$$\mathscr{E}(\rho) = (1-p)^{N_g} U\rho U^\dagger + \left(1-(1-p)^{N_g}\right)\frac{I}{2^n},$$

where $U = \prod_{k=1}^{N_g} U_k$. Thus, for a given input state $|\psi\rangle$, $\langle\psi|\mathscr{U}^{-1}\circ\mathscr{E}(|\psi\rangle\langle\psi|)|\psi\rangle = (1-p)^{N_g} + (1-(1-p)^{N_g})1/2^n \approx (1-p)^{N_g}$ (using $1/2^n \ll 1$). Thus the process fidelity, for small p, can be expressed as a product of the fidelities $f = 1-p$ of the individual operations:

$$F(\mathscr{E}, U) \approx (1-p)^{N_g}. \tag{34}$$

Such an exponential decay of fidelity with the number of operators, proven here exactly for the depolarizing channel, is also observed in actual experiments, like the random circuit sampling task of Google [94].

This exponential decay of fidelity places a severe limitation of what can be achieved on near-term computers. We shall see, in Section 4.1, how this product law for fidelity can be used to quantitatively delineate the boundary to a potential "advantage" of quantum processors. More importantly, this stark decay has also spurred the development of algorithms requiring less deep circuits than, for instance, the quantum phase estimation circuit described in the previous section.

3.2.2. *Variational algorithms for near-term quantum processors*

The variational quantum eigensolver. The textbook quantum algorithms presented in a previous section (Section 3.1) require a number of gates N_g incompatible with the error levels (e.g. p in the depolarizing channel presented in the previous section, Equation (33)) available on current quantum processors. A simple way to reduce this number of gates was proposed in [95]. The idea was to use quantum computers to prepare, and compute the energy of, variational states $|\psi(\boldsymbol{\theta})\rangle$, and offload the rest of the computation to a classical processor. Given an expressive enough variational family, the variational principle

$$E(\boldsymbol{\theta}) = \langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle \geq E_0, \tag{35}$$

(with E_0 the ground-state energy of H) guarantees that a proper optimization, by a classical computer, of the parameters $\boldsymbol{\theta}$, will lead to a good enough approximation of the ground-state energy E_0. This is illustrated in Figure 6.

This approach, dubbed the variational quantum eigensolver (VQE, see [96] for a review, and [97] for an analog version), is robust to systematic errors like overrotations: if a rotation is of angle $\tilde{\theta} = \theta + \delta\theta$ instead of the wished-for θ, the classical optimization will converge to $\tilde{\theta}^* = \theta^* - \delta\theta$ without necessitating a knowledge of the miscalibration error $\delta\theta$. Yet, its main advantage over previous approaches to ground state estimation is that the variational family $|\psi(\boldsymbol{\theta})\rangle$ can be chosen at one's convenience. For instance, one may choose a quantum circuit $U_{\boldsymbol{\theta}}$ that is shallow enough to ensure that the fidelity of the final state $\rho(\boldsymbol{\theta}) = \mathscr{E}_{\boldsymbol{\theta}}(|0\rangle\langle 0|)$, with $\mathscr{E}_{\boldsymbol{\theta}}$ the noisy counterpart of $U_{\boldsymbol{\theta}}$, is close to 1. A major challenge therefore lies in finding short, but expressive enough variational state preparations. Methods in this direction are presented in Sections 4.2, and, to a lesser extent, 4.1.

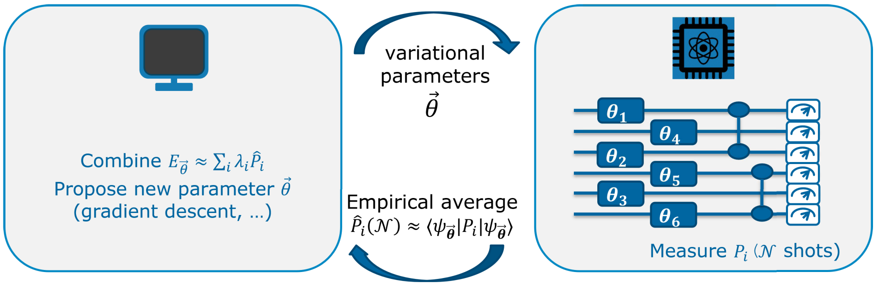
Combine $E_{\vec{\theta}} \approx \sum_i \lambda_i \hat{P}_i$
Propose new parameter $\vec{\theta}$
(gradient descent, ...)
variational parameters $\vec{\theta}$
Empirical average $\hat{P}_i(\mathcal{N}) \approx \langle \psi_{\vec{\theta}} | P_i | \psi_{\vec{\theta}} \rangle$
θ_1
θ_4
θ_2
θ_5
θ_3
θ_6
Measure P_i ($\mathcal{N}$ shots)

Figure 6. Sketch of the VQE algorithm. The left part is realized on a classical processor, while the right part is performed on a quantum processor.

Let us emphasize that switching to variational algorithms like VQE comes at a double price, even in the absence of decoherence.

First, the variational formulation of the ground state energy problem poses the question of the optimizability of the potential landscape $E(\boldsymbol{\theta})$ (Equation (35)). For uniformly random circuits (which have the largest expressivity), the phenomenon of concentration of measure—energy expectation values for a given variational circuit tend to gather around the circuit-average value as the size of the Hilbert space increases—leads to the so-called barren plateau problem ([98], see [99] for a review): gradients of $E(\boldsymbol{\theta})$ become exponentially small, even in the absence of noise (which further compounds this phenomenon). Physically motivated ansätze (like the unitary coupled cluster, UCC, ansatz, used in chemistry, see Section 2.1.6) could also suffer from this problem if they are deep enough [100].

In fact, the barren plateau problem is directly linked to the expressiveness of the ansatz: the variance $\langle E(\boldsymbol{\theta})^2\rangle - \langle E(\boldsymbol{\theta})\rangle^2$ (here $\langle\cdots\rangle$ denotes average over the parameter space) of the cost landscape, whose vanishing signals a barren plateau, is inversely proportional to the dimension of the so-called dynamical Lie algebra (DLA). If the parametric circuit is $U(\boldsymbol{\theta}) = \prod_k \mathrm{e}^{-\mathrm{i}\theta_k G_k}$, the DLA is the vector space generated by nested commutators of the generators G_k of the parametric circuit. (In fact, this statement holds for simple Lie algebras and more importantly, for circuits deep enough that they form a "2-design", namely an effectively random circuit, see [101] for the full result and its derivation). In other words, the larger the variational space being explored by VQE, the more likely the occurrence of flat landscapes.

A second price of VQE is that the estimation of the variational energy $\langle\psi(\boldsymbol{\theta})|H|\psi(\boldsymbol{\theta})\rangle$ at each step is performed, in practice, by splitting H as a sum of Pauli terms, $H = \sum_{i=1}^{N_p} \lambda_i P_i$, collecting the average value $\langle\psi(\boldsymbol{\theta})|P_i|\psi(\boldsymbol{\theta})\rangle$ of each local operator, and then summing the terms (see Figure 6). The second stage is essentially the construction of a classical estimator of $\langle\psi(\boldsymbol{\theta})|P_i|\psi(\boldsymbol{\theta})\rangle$: one generates $|\psi(\boldsymbol{\theta})\rangle$, does a projective measurement of P_i, which yields a ±1 outcome $\pi_s^{(i)}$, and repeats the procedure $\mathcal{N}$ times, yielding

$$\langle\psi(\boldsymbol{\theta})|P_i|\psi(\boldsymbol{\theta})\rangle \approx \frac{1}{\mathcal{N}} \sum_{s=1}^{\mathcal{N}} \pi_s^{(i)} = \hat{P}_i(\mathcal{N}). \tag{36}$$

The finite-sampling error ϵ between the left-hand side and the right-hand side scales as $1/\sqrt{\mathcal{N}}$, so that the computational time of VQE will generically scale as $O(\mathcal{N}) = O(1/\epsilon^2)$, as in classical Monte-Carlo techniques encountered in Section 2.1.3. This is a major drawback compared to QPE, whose run time scales as $1/\epsilon$ (albeit with many more gates and thus the need for an error-corrected quantum computer, as explained in Section 3.1). VQE also suffers from a major drawback compared to variational Monte-Carlo because, contrary to VMC, it does not benefit from the variance reduction phenomenon: as one gets closer to the sought-after ground state, the variance of the individual Pauli terms does not vanish (the eigenstate of H is not an eigenstate of the individual P_i's), as is the case in VMC for the local energy.

It must be pointed out that mathematical statements about barren plateaux are average statements. They do not preclude the existence of portions of the parametric space where gradients do not vanish. However, these "fertile valleys" are narrow ones [102]. One of the future challenges of VQE is to start the optimization process close to or in these narrow gorges. Using classical methods to find good starting points is most probably a promising avenue in this direction. The recently introduced adaptive ansatz generation strategies, which construct the ansatz by selecting the operators that maximize the gradients, are also promising in this respect [103, 104].

Of course, even in the absence of barren plateaux and if shot noise is small enough, decoherence also has to be reckoned with. A major challenge of VQE is thus to find the shallowest ansätze to curb the deleterious influence of noise.

The design of variational quantum circuits. Designing variational quantum circuits $U(\boldsymbol{\theta})$ must meet two contradictory goals: being expressive enough to be able to reach the sought-after ground state during optimization, and being short enough to limit the impact of decoherence.

To meet the expressivity requirement, a simple strategy, sometimes called the hardware-efficient ansatz [105], is to define a circuit that is "entangling enough" by using the available two-qubit gates on the hardware (hence the name), and random single-qubit rotations, so that the state created by the ansatz approximates well a random quantum state. These states are characterized by a high level of entanglement, and may therefore have a good chance of getting close to the ground state of the Hamiltonian. A main drawback of these circuits, however, is that they are very prone, by design, to the barren plateau problem.

The complementary strategy is to construct physically-motivated ansätze: they will cover the Hilbert space less extensively but will be easier to optimize. A well-known ansatz in this family is the unitary coupled cluster (UCC) ansatz we introduced in Equation (19). Since $T - T^\dagger$ is antihermitian, $\mathrm{e}^{T-T^\dagger}$ is unitary and can thus a priori be implemented on a quantum processor. In practice, this is achieved by finding a Pauli decomposition $T - T^\dagger = \mathrm{i}\sum_k \lambda_k P_k$ and Trotterizing the corresponding sum of terms as described above (Section 3.1.4). The length of the so-obtained circuit is polynomial in the number of orbitals, contrary to (untruncated) classical implementations of the UCC (or VCC) methods, raising hopes that VQE could bring an advantage compared to classical methods.

Among physically-inspired ansätze, one can also mention the Hamiltonian variational ansatz (HVA, [106]), which is inspired by the annealing method (see Section 3.1.4). Following the adiabatic theorem, if one starts from the ground state of the Hamiltonian H at hand (say the Hubbard model) where one has removed the hopping terms (such a Hamiltonian, H_0, commutes with $c_i^\dagger c_i$ and thus admits Fock states, namely factorized states, as eigenstates $|\Phi_0\rangle$), and deforms it (see Equation (28)) into the Hamiltonian of interest H, one is guaranteed to end up in the ground state of H. The HVA takes inspiration from the Trotterization of the interpolating Hamiltonian by proposing the following ansatz:

$$|\Psi(\boldsymbol{\theta})\rangle = \prod_{k=1}^{M} \mathrm{e}^{-\mathrm{i}\theta_{2k} H_0} \mathrm{e}^{-\mathrm{i}\theta_{2k+1} H} |\Phi_0\rangle, \tag{37}$$

with $2M$ parameters $\boldsymbol{\theta}$. The hope underlying the method is that the total duration of the circuit once the parameters have been optimized is shorter than the (long) duration dictated by the adiabatic theorem (HVA relies on the hope to perform a "shortcut to adiabaticity").

3.2.3. *Applications to fermionic many-body problems: from molecules to the Hubbard model to the Anderson model*

After its introduction by [95], VQE was tested on a number of toy systems that served as proofs of concept, with no aim to compete with classical methods (see e.g. [107] for an application for the H_2 molecule). In parallel, theoretical papers estimated the amount of computational resources required to reach a satisfactory level of accuracy for larger systems beyond toy models. In particular, [106] gave resource estimates for computing the ground state energy of small molecules to chemical accuracy, namely to within 1 mHa of the exact ground state energy (such levels of accuracy are needed to compute chemical rates: those depend on activation energies E_a through a factor $\propto \mathrm{e}^{-E_\mathrm{a}/k_\mathrm{B}T}$ via the Arrhenius or Eyring laws; only predicting whether a reaction is likely or not to happen requires that E_a, and therefore the ground-state energies involved in computing E_a, be known at least up to a $O(k_\mathrm{B}T)$ error; at room temperature, this corresponds to 1 mHa). To obtain a resource estimate, [106] used the fact that the shot noise error can be upper bounded as

$$\Delta H(\mathcal{N}) \leq \frac{\|H\|_1}{\sqrt{\mathcal{N}}}, \tag{38}$$

with the one-norm defined as $\|H\|_1 = \sum_i |\lambda_i|$. Such an estimate is based on a nonuniform allocation of shots among the Pauli terms: each Pauli term $\lambda_i P_i$ in the decomposition of H receives a fraction of the total budget proportional to $|\lambda_i|$ ([108]; more advanced strategies are possible, see e.g. [109], but do not alter the upper bound). At least $\mathcal{N} \approx \|H\|_1^2/\Delta H^2$ are thus needed to reach a given accuracy ΔH. For the H_2O molecule, since $\|H\|_1 = 36$ Ha (in the STO-6G basis), one obtains a number of $\mathcal{N} = 36^2/(10^{-3})^2 \approx 10^9$ samples. Assuming a clock cycle (processor rate) of 1 MHz (an optimistic estimate for superconducting processors, whose measurement durations are in the µs range), this corresponds to 1000 s per energy evaluation, without factoring in gradients (which basically require a multiplication of the above figure by the number of parameters). Counting gradients and the number of optimization steps, one easily reaches months [106].

Similar estimates for the Hubbard model lead to less dire estimates—of the order of days—due to a smaller number of terms and translation invariance [106]. This appears more promising, although this estimate completely leaves aside optimization issues (the barren plateau problem) and decoherence.

As argued in the introductory section, using embedding techniques like DMFT, RISB or DMET bring another level of simplification by shifting the focus to a simpler effective model, namely the Anderson impurity model (AIM, Equation (21)). Motivated by the success of these techniques and by the more modest effort of solving the AIM compared to the Hubbard model, a few works proposed to use a quantum processor to solve the impurity model.

References [89] and [88] introduced the general methodology of using quantum processors to solve DMFT equations. Reference [89] numerically simulated the effect of hardware errors in a trapped-ion implementation of a nonequilibrium DMFT scheme to describe a quench of the kinetic term of the Hubbard model, with 3 to 21 qubits (corresponding to 2 and 10 AIM bath sites, respectively). The Green's function was measured using a Trotter compilation of the time evolution (as described above). Reference [88] focussed on equilibrium DMFT, with a proposal to use quantum phase estimation to measure the Green's function. The following works, which were aimed at practical implementations, resorted to a simplified DMFT scheme called two-site DMFT [110], which consists in limiting the bath to only one bath site. This in turns means that only 4 qubits are needed. Reference [90] numerically simulated two-site DMFT with a (noiseless) superconducting platform implementation of the Trotter evolution required by the Green's function computation, while the implementation of the ground state preparation needed to initialize the Green's function circuit was not discussed. Reference [111] implemented two-site DMFT with VQE as the algorithm used to compute the energy of the ground state as well as of the $N+1$ and $N-1$ manifolds needed to compute the Green's function in a Lehmann representation. This VQE was implemented on superconducting as well as trapped-ion hardware. Reference [112] performed noisy simulations of two-site DMFT, with a standard Trotter time evolution to compute the Green's function, and a black box for the ground state preparation. The minimal levels of noise needed to get useful results for two-site DMFT were computed in this same work. Reference [113] implemented a Gutzwiller (RISB) method to study the periodic Anderson model (a variant of the Hubbard model), with a VQE-based, two-qubit implementation on a superconducting processor. More recent work [114] carried out a DFT+DMFT computation for a multiorbital Hubbard model with an AIM with up to 6 bath sites (and a single impurity), corresponding to 14 qubits. However, the determination of the optimal VQE parameters was done on a classical computer.

All the experimental runs comprising a full-blown state preparation were limited to at most 4 qubits, and should thus be considered as mere proof-of-principle experiments. The main question this state of affairs raises is how to increase the size of the AIM (whether in terms of number of impurities, or number of bath sites, or both) while remaining tractable. We will give some insights into this in the next section.

4. Paths towards useful quantum-classical algorithms

In the previous two sections, we have outlined the main methods used on classical computers (Section 2) and on the emerging quantum processors (Section 3) for solving fermionic quantum many-body problems. We have identified specific advantages and drawbacks of both computing paradigms, and in particular

(1) the difficulty of classical computers to handle equilibrium systems with long-distance correlations and/or a high degree of entanglement, limiting the reach of the most advanced methods to solve strongly-correlated models like the Hubbard model, or even simpler models, self-consistently "distilled" by embedding methods, like the Anderson impurity model;
(2) the difficulty of classical computers to handle out-of-equilibrium dynamics due to the fast growth of entanglement;
(3) the limitations of current noisy quantum hardware due to decoherence, which put a low cap to the maximum size of viable quantum circuits;
(4) the fundamental limitations of even noiseless quantum circuits, like shot noise, or the need for input states with high overlaps in quantum phase estimation.

The goal of this section is to outline recent research results to overcome some of these difficulties with a common pattern: the combination of classical and quantum algorithms to solve the many-body problems at hand.

Three main directions will be elaborated on: Section 4.1 explores how tensor network methods can be used to delineate, and perhaps reach, quantum advantage. Section 4.2 elaborates on hybrid methods for tackling fermionic many-body problems. Finally, Section 4.3 extends the scope to hard optimization problems, which can be regarded as classical many-body problems, and which are natural targets for quantum processors.

4.1. *Tensor networks: classical yardsticks of quantum supremacy claims… and antidotes to the plague of decoherence?*

As discussed in Section 4.1 above, tensor networks are powerful classical tools to represent many-body states. As it were, tensor networks are also deeply tied to quantum circuits: executing a

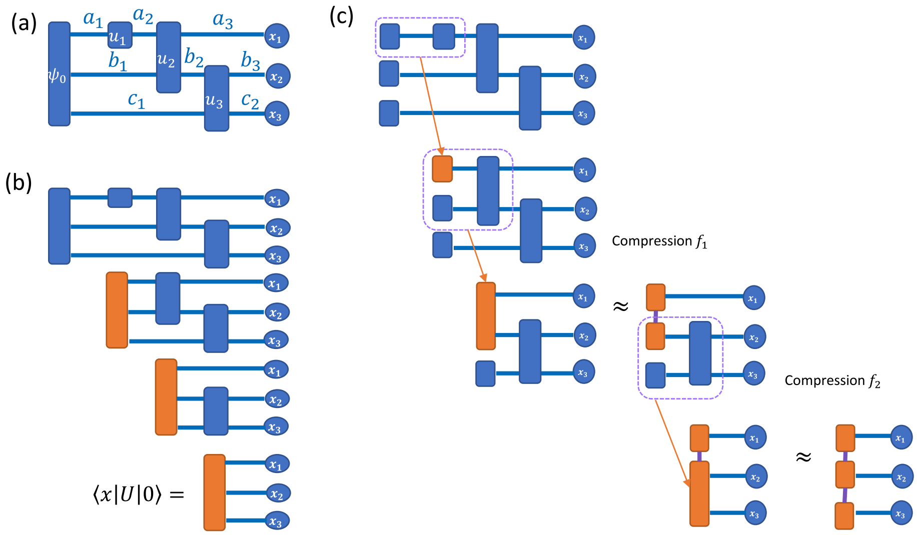

Figure 7. Tensor network representations of quantum circuits and contraction strategies. (a) General tensor network, with vertices ψ_0, u_1, u_2, u_3 and x_1, x_2, x_3, and edges a_1, a_2, a_3, b_1, b_2, b_3 and c_1, c_2. (b) “Schrödinger” strategy to contract the tensor network: the quantum state is stored as a dense $2 \times 2 \times 2$ array ψ_{b_1,b_2,b_3}. (c) Matrix-product-state strategy: the quantum state is represented as a matrix product state, e.g. with three connected tensors. After the application of a two-qubit gate, a SVD compression is performed.

quantum circuit on a gate-based quantum processor can be regarded as contracting a tensor network. Before we explain this statement, let us emphasize that this deep connection both opposes, and brings closer, tensor network methods and quantum computing: it opposes them as two competing methods, whose computational merits and shortcomings can be compared; it brings them closer as complementary methods, which could possibly be used in combination with each other depending on the hardware constraints of the computation.

Thus, tensor network provide a natural comparison point for the run time and memory cost of perfect quantum computations, namely the CPU and memory needed for contracting the corresponding tensor network. As we shall see in the next section, this comparison point has been used to support claims of quantum supremacy. However, we will also argue that for the noisy quantum computers that are available today, the reference point should be a *lossy* contraction of the tensor network (Section 4.1.1). We will also see that some tensor networks could potentially be used to construct quantum circuits that are more robust to noise (Section 4.1.2).

The connection between a quantum circuit execution and a tensor network contraction is the following. The expectation values $\langle\psi|\hat{O}|\psi\rangle$ or outcome probabilities $|\langle x|\psi\rangle|^2$ (where $x = (b_1, b_2, \ldots, b_n)$ represents a bitstring, and $|x\rangle$ an eigenstate of the $\hat{Z} \otimes \hat{Z} \cdots \hat{Z}$ operator), of a state $|\psi\rangle = \prod_{k=1}^{N_g} U_k |0\rangle^{\otimes n}$ generated by a quantum circuit with N_g gates on n qubits can be regarded as the outcome of contracting a tensor network. As a reminder, a tensor network is defined by a graph $G = (V, E)$ with vertices V and edges E, and tensors $\{T_k\}_{k \in V}$ associated to each vertex. In our situation, the tensors correspond to gates $\{U_k\}_{k=1 \ldots N_g}$, operators $\hat{O}$ or states $|\psi\rangle$, $|x\rangle$ or $|0\rangle^{\otimes n}$, and one edge (i, j) of the graph corresponds to a summation over the corresponding indices of tensors T_i and T_j. This is illustrated in Figure 7(a) for the computation of $\langle x|\psi\rangle = \langle x|U|0\rangle^{\otimes n}$ for $n = 3$ qubits. To compute the actual values $\langle\psi|\hat{O}|\psi\rangle$ or $|\langle x|\psi\rangle|^2$, one needs to perform the contraction of the tensor network, namely actually perform the summation over the internal variables. Two steps are necessary for this: the determination of the contraction order (some orders yield better computational complexities) and the contraction itself. The determination of the optimal contraction order (that leads to the smallest complexity) is a NP-complete problem. As already discussed in Section 2.1.4, finding this order generically scales as $O(\mathrm{e}^{T(G)})$, where $T(G)$ is the so-called treewidth of the graph. For graphs corresponding to, e.g. $l \times L$ lattices, $T(G) \sim \min(l, L)$. Thus, if a circuit topology roughly corresponds to a $n \times D$ lattice (where n is the number of qubits and D the depth of the circuit), the generic cost of finding the best tensor contraction order for this circuit is $O(\mathrm{e}^{\min(n,D)})$. The general space complexity of the contraction is also of order $\mathrm{e}^{T(G)}$. This means that tensor network contractions are in principle exponentially costly, making a strong case for (perfect) quantum computers: they "contract" this tensor network in polynomial time. However, real quantum computers come with their own exponential burden, namely decoherence (Equation (34)). The outcome of the comparison between tensor networks and quantum computers thus depends on the parameters of the circuit to be executed or simulated and on the hardware properties.

4.1.1. *Assessing quantum supremacy claims with grouped matrix product states*

Google's supremacy claim with random quantum circuits. The advent of large-scale quantum processors, with tens of qubits, in the 2010s, led to efforts to design special experiments geared at demonstrating the advantage of quantum computers over classical computers. The first such claim was made in 2019 by Google's team using transmon processors [94], essentially coupled anharmonic oscillators described by a Bose–Hubbard model. The task that Google proposed as a possible low-hanging fruit for demonstrating quantum advantage of these processors over classical algorithms was that of sampling bitstrings in random, two-dimensional quantum circuits with 53 qubits. Such circuits are known to lead to a fast growth of entanglement entropy [115]. They are therefore the best way to beat the exponential decay of fidelity presented in Section 3.2.1 while being difficult targets for tensor contraction techniques (and their exponential scaling with entanglement).

The main figure of merit of Google's experiment was the so-called linear cross-entropy benchmarking fidelity,

$$\mathrm{XEB} = \frac{2^n}{\mathcal{N}} \sum_{i=1}^{\mathcal{N}} P(x_i) - 1, \tag{39}$$

where $P(x) = |\langle x|\Psi\rangle|^2 = |\langle x|U|0\rangle|^2$ is the probability of measuring bitstring x after the execution of the random quantum circuit described by unitary operator U, and $\{x_i\}_{i=1\ldots\mathcal{N}}$ are $\mathcal{N}$ samples measured in the experiment (note that $P(x)$ is exponentially costly to compute, a fact that led us to propose an alternative benchmark of quantum computers, see Section 4.3.3 below). Under certain assumptions, this number can be proven to equal the state fidelity $\mathcal{F} = \langle\Psi|\rho|\Psi\rangle$ of the (noisy) final state ρ obtained in the experiment.

Google's claim essentially was that the run time needed to reach a given XEB in the experiment (0.2% in the 2019 experiment, down from 100% for a perfect quantum computer) was orders of magnitude smaller (200 s compared to 10,000 years) than that of performing a classical computation to sample from the final distribution with a similar quality. The figure of 0.2% can be obtained directly (albeit only roughly) from the product formula Equation (34) with the experiment's error rates and number of operations. The figure of 10,000 years was obtained by resorting to a particular way of contracting the tensor network corresponding to the random quantum circuit at hand. We sketch here the rough strategy: instead of performing the full tensor network contraction, Google considered the computation of $P(x)$ as the square modulus of the summation over the internal indices of the tensor network:

$$P(x) = \left| \sum_{a_1,a_2,a_3,b_1,b_2,b_3,c_1,c_2} \left[\psi_0\right]_{a_1 b_1 c_1} [u_1]_{a_1 a_2} [u_2]_{a_2 b_1 a_3 b_2} [u_3]_{b_2 c_1 b_3 c_2} [x_1]_{a_3} [x_2]_{b_3} [x_3]_{c_2} \right|^2$$

(here spelled out in the simple case illustrated in Figure 7). They evaluated cost $\mathscr{C}$ of evaluating one term (one fixed assignment of $a_1, a_2, a_3, b_1, b_2, b_3, c_1, c_2$) on a classical computer. To get the total (naive) contraction cost, one then in principle needs to multiply $\mathscr{C}$ by the number $\mathscr{A}$ of possible assignments or "paths" (in analogy to the Feynman path integral technique, [116]). In practice, however, the finite XEB or fidelity needs to be taken into account; it can be shown that summing only a fraction $\mathscr{F}$ of the paths leads to a state of fidelity $\mathscr{F}$ with respect to the perfect state (that corresponds to all paths). Therefore, the back-of-the-envelope total computational cost is $\mathscr{F} \cdot \mathscr{A} \cdot \mathscr{C}$. The 10,000 years are obtained from a similar formula.

Many followup works challenged Google's estimation of the best classical run time. Most relied on a combination of the general tensor network contraction strategy outlined above and the Feynman-path approach: the so-called tensor-slicing method selects only a subset of the internal variables for a Feynman-like handling, and contracts, using usual tensor-contraction strategies, the rest of the internal variables. These methods [117–120] led to greatly reduced estimations, down to a few seconds. However, due to the aforementioned inherent exponential scaling of tensor network contraction methods, these methods are hardly scalable, since they are exponential in the number of qubits or depth of the circuit.

A tensor network compression strategy. A different strategy to tackle the problem of matching experimental fidelities with a classical technique was proposed in [121]. It consists in performing the tensor network contraction in a "Schrödinger"-style, as illustrated in Figure 7(b): the tensors are contracted from the left to the right, i.e. following the time arrow, like the evolution of a Schrödinger equation. However, to avoid the computational cost of storing the full wave function and of performing the matrix-vector multiplications corresponding to the contractions, the wavefunction is stored not as a dense vector of 2^n complex amplitudes, but as a matrix product state (see [122] for a first use of MPS to emulate quantum circuits). Each entangling gate is followed by a compression step via a singular value decomposition (SVD), see Figure 7(c). The singular values (also known as Schmidt coefficients in this context) s_i are truncated when their index is larger than the bond-dimension χ of the MPS.

The fidelity of such a compression is given by the overlap between the original MPS state $|\psi\rangle$ (with Schmidt coefficients s_i) and the compressed state $|\tilde{\psi}\rangle$ (the truncated singular values $\tilde{s}_i$ are also renormalized to preserve the unity of the norm):

$$f = \left|\langle\psi|\tilde{\psi}\rangle\right|^2 = \left(\sum_i s_i \tilde{s}_i\right)^2 = \sum_{i\le\chi} s_i^2. \tag{40}$$

For random quantum circuits, the product formula Equation (34) can be shown to hold ([121]), so that the final fidelity can be estimated as

$$\mathscr{F}_{\text{MPS}} = |\langle\Psi|\Psi_{\text{MPS}}\rangle|^2 \approx \prod_k f_k, \tag{41}$$

where f_k is the fidelity of the kth compression step. Such an algorithm works, but the final fidelity $\mathscr{F}$ obtained for 2D circuits of the type used by Google's team is far below the aimed-for 0.2%. This failure comes from the numerous contraction steps and from the fact that the singular values of the random quantum states generated by the circuit have a very long tail, which means that neglecting even high-index singular values causes a sizable error $1 - f = \sum_{i>\chi} s_i^2$.

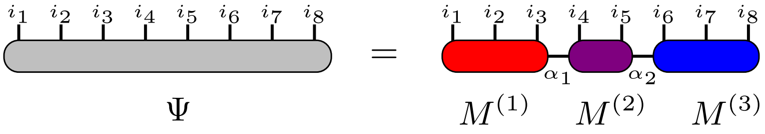
i_1 i_2 i_3 i_4 i_5 i_6 i_7 i_8
Ψ
=
i_1 i_2 i_3 i_4 i_5 i_6 i_7 i_8
α_1
α_2
$M^{(1)}$
$M^{(2)}$
$M^{(3)}$

Figure 8. "Grouped" MPS representation of a 8-qubit quantum state $|\Psi\rangle$: the eight qubits are partitioned into three groups of 3, 2 and 3 qubits respectively, corresponding to three tensors $M^{(1)}$, $M^{(2)}$ and $M^{(3)}$. A standard MPS would use n tensors for n qubits. Adapted from [123]. Copyright (2023) by the American Physical Society.

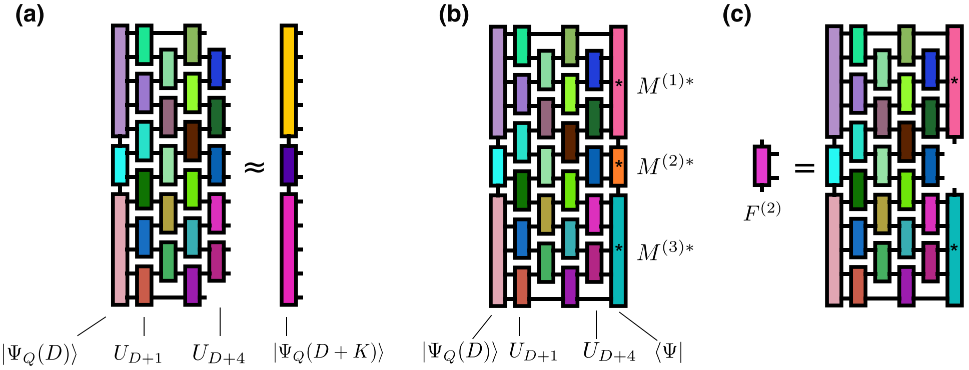

Figure 9. Compression step in the DMRG algorithm. (a) General schematic of the compression step: one adds K (N_l in the text) layers of the quantum circuit, then approximates the resulting state with a MPS. (b) Tensor network representation of the scalar product to be optimized. (c) Computation of the $F^{(\tau)}$ tensor. From [123]. Copyright (2023) by the American Physical Society.

This is true even when using so-called "grouped MPS" [121], where some of the qubits are grouped into one tensor (instead of one tensor per qubit for regular MPS), as illustrated in Figure 8. Grouped MPS interpolate between a dense representation of $|\Psi\rangle$ (of storage cost 2^n) and a plain-vanilla MPS representation (of storage cost $\leq 2\chi^2 n$).

A subsequent work of ours [56] addresses this issue by switching from a SVD compression strategy to a strategy inspired by the density-matrix renormalization group (DMRG) method. Instead of compressing the (grouped) MPS after each entangling gate via a SVD, we optimize the MPS only after a certain number of layers of entangling gates using the DMRG algorithm, namely by finding the MPS of fixed bond dimension with largest overlap with the state obtained by applying a number N_l of entangling layers on the previous MPS, as illustrated in Figure 9(a) and (b). To perform this optimization technique, we need to perform the contraction of a network containing the previous MPS and N_l new layers, with a cost of $O(e^{N_l})$ provided $N_l \leq n$ (using the treewidth estimation of the cost): indeed, the optimal tensor $M^{(\tau)*}$ is given, up to a normalization constant, by

$$M^{(\tau)*} \propto F^{(\tau)},$$

with $F^{(\tau)}$ graphically defined in Figure 9(c), namely it is essentially a $N_l \times n$ grid of tensors to be contracted.

This new optimization strategy leads to larger compression fidelities, as illustrated in Figure 11, which compares the technique used in [121] (dubbed time-evolving block decimation, TEBD, in reference to a similar method for time-evolving matrix product states) with two variants of our technique, the "open" and the "closed" simulation. The open strategy contracts the

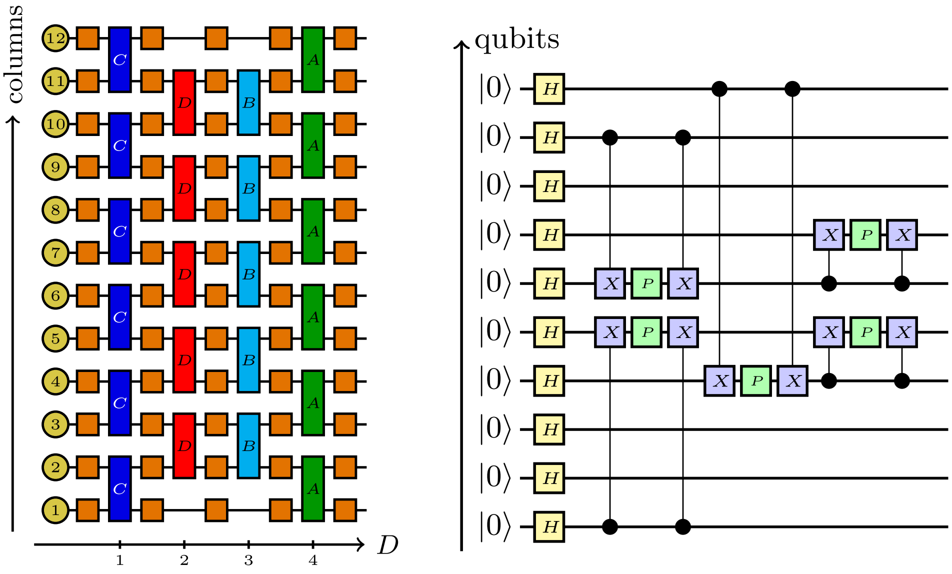

Figure 10. Circuits used in the MPS simulations. Left: Google (modified) supremacy sequence. Right: QAOA circuit for a MaxCut problem. From [123]. Copyright (2023) by the American Physical Society.

circuit on the initial MPS from left to right, yielding an "open" MPS, namely with arbitrary external indices. The closed strategy aims at computing $\langle x|\Psi\rangle$ for a fixed bitstring $|x\rangle$: one fixes the external indices to a set bitstring x and contracts the circuit both from the left and from the right, thus halving the effective depth of the circuit and thus improving the compression error. The fact that our method (in its two variants) outperforms the TEBD variant can be ascribed to the lesser frequency of compression steps and, more importantly, to the larger optimization freedom entailed by DMRG compared to simple SVD compression, a well-known feature in MPS [23].

Concretely, for Google's random circuits on 54 qubits (a sketch of which is shown in Figure 10(left)), we obtain fidelities above the experimental fidelities for bond dimensions of $\chi \sim 50$, corresponding to a few hours of computations. More importantly, the scaling of our simulation is not exponential in the circuit depth D nor in the number of qubits $n = n_c \times n_r$ (with n_c the number of columns and n_r the number of rows of the 2D grid of qubits). A rough estimate of the run time assuming a grouping of the qubits by columns is

$$O(\chi^3 2^{n_c} n_r N_{\mathrm{sweeps}} \mathrm{e}^{N_l} D/N_l). \tag{42}$$

The exponential in n_c (which, in the worst case, is $\sqrt{n}$) comes from the storage of dense vectors for a given column, while the exponential in N_l comes from the contraction over N_l layers. N_{sweeps} is the number of DMRG sweeps. The χ^3 scaling comes from the QR decompositions needed to make the MPS canonical, a crucial requirement of DMRG. This makes the computation scalable to quite large numbers of qubits compared to techniques scaling in 2^n (instead of $2^{n_c} \sim 2^{\sqrt{n}}$). We note in passing that this $2^{\sqrt{n}}$ scaling would in principle disappear when using 2D tensor networks instead of (grouped) MPS to represent the state—with, however, the issues associated with 2D tensor networks such as PEPS, that do not come with a canonical form and therefore a polynomial contraction strategy (an exception being the recently developed isometric tensor network states (isoTNS, [27]), which however come with a steeper $O(\chi^7)$ scaling).

Another interesting feature of the quality of the approximate state obtained through this method is that is strongly depends on the circuit to be simulated: for random quantum circuits,

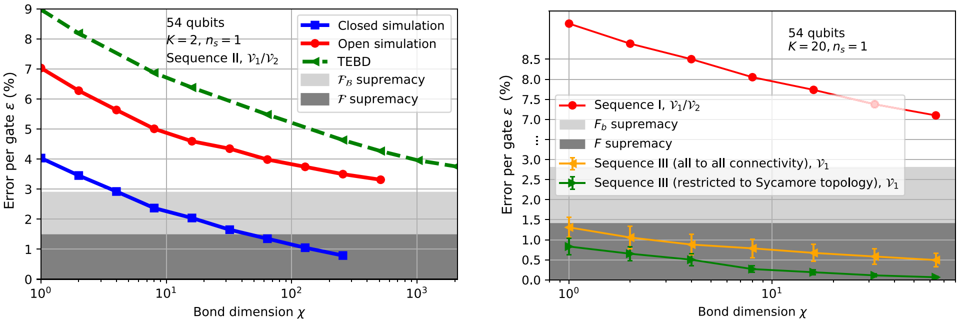

Figure 11. Error per gate as a function of the bond dimension. Left: Google supremacy random circuit. Right: QAOA circuit for a MaxCut problem with 54 qubits. From [123]. Copyright (2023) by the American Physical Society.

reaching low enough error rates requires quite large bond dimensions, while circuits used in the QAOA algorithm [124], a variational quantum algorithm for solving combinatorial optimization problems, illustrated in Figure 10(right), yield very low error rates even for small bond dimensions, as illustrated in Figure 11(right).

The reason for this quality variability among circuits is linked to the entanglement generated by the circuit at hand: the fundamental bottleneck of the method comes from the requisite bond dimension χ. It is related to the (bipartite) von Neumann entanglement entropy S of the state to be represented. S is defined as the Shannon entropy of the distribution of the squared singular values (or Schmidt coefficients) of state $|\Psi\rangle$:

$$S = -\sum_{i=1}^{\chi} s_i^2 \log_2 s_i^2. \tag{43}$$

Since the entropy is convex, it is maximized by constant singular values $s_i = \text{const} = 1/\sqrt{\chi}$ (the value of the constant is chosen to ensure normalization of the state), leading so $S \leq \log_2 \chi$. Thus, a necessary condition for a MPS to be able to exactly represent a state with an entanglement entropy S is

$$\chi \geq 2^S. \tag{44}$$

Usually, this statement is colloquially turned into a scaling of $\chi \sim O(2^S)$, which we will use in our analysis. It means that the exponential wall of a MPS representation comes, as already mentioned in Section 2.1.4, from the degree of entanglement of the state.

We can now come back to the difference between random quantum circuits and QAOA circuits: random quantum circuits are known to generate a fast (ballistic) increase of entanglement and quickly reach a volume law entanglement, namely the maximum reachable entropy. Consequently, any other circuit, like a QAOA circuit, will produce less entangled states and thus be more easy to simulate with this method. More details about this MPS simulation of Google's circuit can be found in the original article [123].

Noisy simulations with tensor networks. In the previous subsection, we described a tensor-network method that can mimic a quantum computation with a comparable level of fidelity. However, the origin of the errors in the MPS algorithm we just outlined, namely compression via the DMRG technique, is very different from the origin of errors in actual quantum computers, that is, decoherence.

In fact, tensor networks can also be used to simulate the evolution of noisy quantum states. In principle, this task looks much (exponentially) harder than simulating a perfect quantum computer, since the cost of storing a density matrix ρ on n qubits is 4^n, compared to 2^n for a

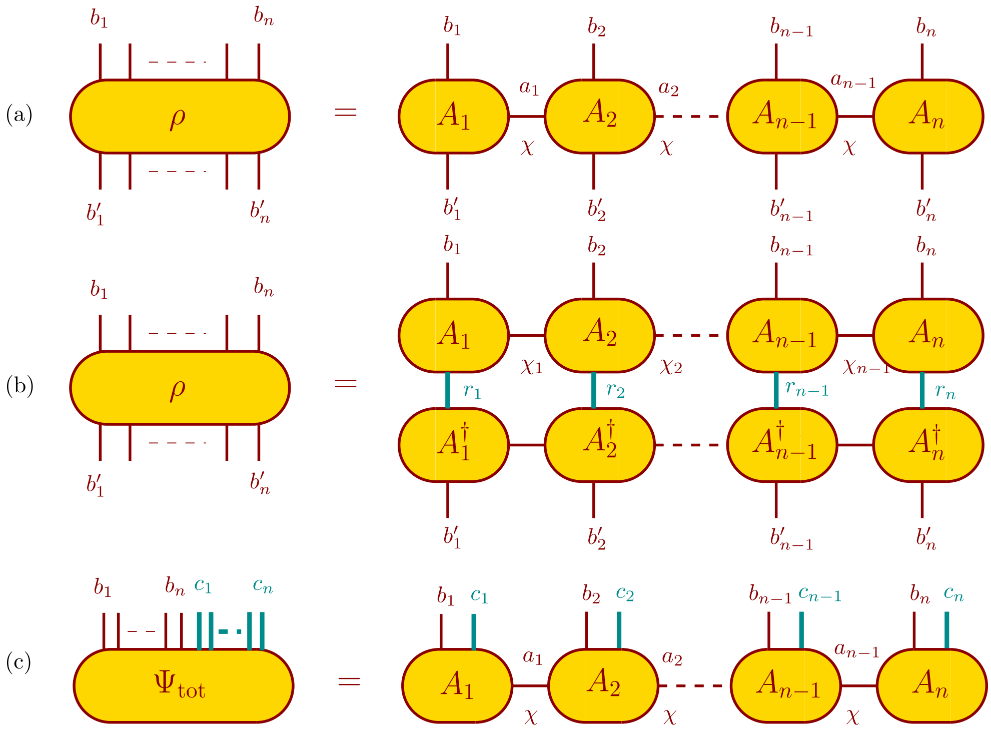
(a)
b_1
b_n
ρ
b'_1
b'_n
=
b_1 b_2 b_{n-1} b_n
A_1 A_2 A_{n-1} A_n
a_1 a_2 a_{n-1}
χ χ χ
b'_1 b'_2 b'_{n-1} b'_n
(b)
b_1
b_n
ρ
b'_1
b'_n
=
b_1 b_2 b_{n-1} b_n
A_1 A_2 A_{n-1} A_n
χ_1 χ_2 χ_{n-1}
r_1 r_2 r_{n-1} r_n
$A_1^\dagger$ $A_2^\dagger$ $A_{n-1}^\dagger$ $A_n^\dagger$
b'_1 b'_2 b'_{n-1} b'_n
(c)
b_1 b_n c_1 c_n
Ψ_{tot}
=
b_1 c_1 b_2 c_2 b_{n-1} c_{n-1} b_n c_n
A_1 A_2 A_{n-1} A_n
a_1 a_2 a_{n-1}
χ χ χ

Figure 12. Two representations of density matrices. (a) Matrix product operator (MPO). (b) Matrix product density operator (MPDO). (c) MPS representation of the purification $|\Psi_{\mathrm{tot}}\rangle$ of ρ.

pure state. But since the cost of tensor network computations is largely driven by the degree of entanglement of the state to be represented, they should come with a natural advantage for simulating noisy evolutions: intuitively, decoherence destroys entanglement, and therefore a smaller bond dimension (as per Equation (44)) should be needed to represent a noisy state.

The most straightforward extension of matrix product states to represent operators such as the density matrix is the matrix product operator (MPO) representation, illustrated in Figure 12(a). It allows to define an entropy for mixed states called the operator-space or matrix-product-operator entanglement entropy (OEE): it is the Shannon entropy of the (renormalized) singular values $s_i^2/\sum_j s_j^2$ of the MPO.

In the presence of depolarizing noise of probability p (see Equation (33)), one observes, in 1D random quantum circuits, that the entanglement entropy increases to a maximum value of [125]:

$$S_{\mathrm{OEE}}^{\max} \sim \frac{1}{p^{1/\alpha}}, \tag{45}$$

with $\alpha \approx 2$, before decreasing once the optimal depth (which is itself shorter and shorter as p increases) is reached. This is illustrated in the top row of Figure 13 by the “MPO” curve: the entropy first increases and then decreases (the increase is almost absent for $p = 10\%$, which is a very strong noise level). This trend in the entropy is reflected in the bond dimension: here, we adjusted the bond dimension to discard all singular values below a certain threshold ϵ. Thus, the bond dimension automatically adjusts to the entropy of the state: we do observe first an increasing bond dimension (entanglement creation through a unitary evolution dominates), followed by a decrease (entanglement destruction by noise dominates). This behavior and the

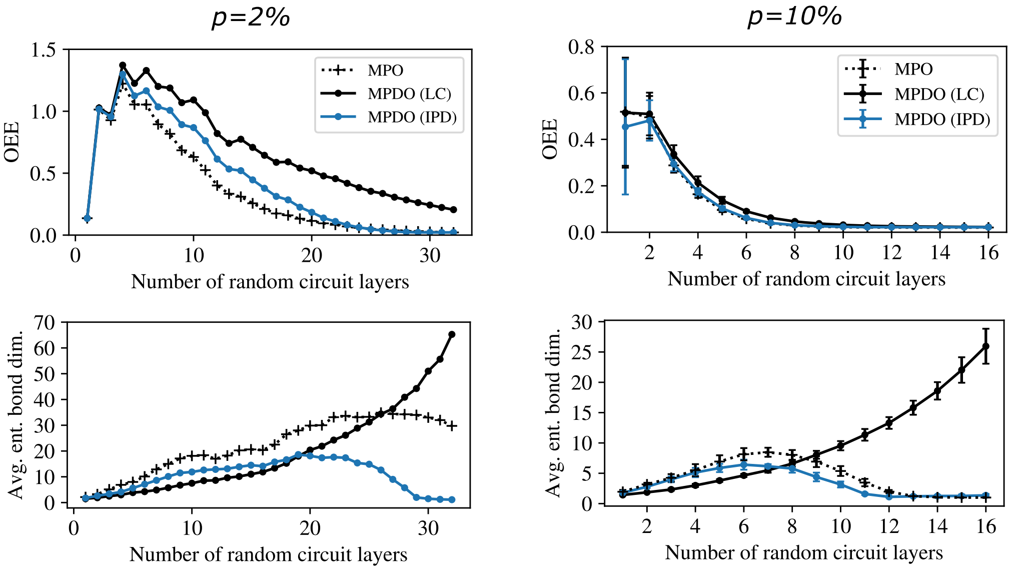

Figure 13. Simulations of random 1D quantum circuits with 8 qubits and depolarizing noise with MPOs and two MPDO truncation strategies. Top row: evolution of the operator entanglement entropy as a function of the number of layers. Bottom row: evolution of the (bond-averaged) bond dimension for truncation thresholds of $\epsilon = 0.01$, 0.015 and 0.001 for MPO, MPDO (LC) and MPDO (IPD). Left column: depolarization level of $p = 2\%$. Right column: $p = 10\%$. Adapted from [126].

scaling law Equation (45) point to the importance of hardware improvements to lower p: this will in turn raise the bar for the bond dimensions needed to reproduce quantum results.

A formal issue with a MPO representation of the density matrix is that is does not guarantee the positive definiteness of the matrix upon truncation of its singular values. This property can be imposed by first decomposing ρ as $\rho = AA^\dagger$ (with A a $2^n \times r$ matrix, with r the rank of ρ) and then using a MPO representation of A. This yields the so-called matrix product density operator (MPDO) representation, illustrated in Figure 12(b). The internal vertical bonds can be interpreted as representing environment sites (as opposed to the external vertical bonds, which are physical system sites). Indeed, a MPDO representation can also be thought of as the MPS representation (see Figure 12(c)) of a given purification Ψ_{tot} of ρ, namely a state Ψ_{tot} defined in an enlarged Hilbert space with environmental degrees of freedom and such that

$$\rho = \mathrm{Tr}_{\mathrm{env}} \left[|\Psi_{\mathrm{tot}}\rangle \langle \Psi_{\mathrm{tot}}| \right], \tag{46}$$

with $\mathrm{Tr}_{\mathrm{env}}$ the partial trace on the environment.

Thus, in addition to the usual “horizontal” bond dimensions χ_i, which essentially capture the degree of entanglement of the purification Ψ_{tot}, a MPDO is also parameterized by “vertical” bond dimensions r_i, which depend on the purity of the state (a pure state should have $r_i = 1$, namely A is $2^n \times 1$ in this case). Evolving a MPDO in time is similar to evolving a MPS or a MPO (see Figure 7(c)), except that compression steps involve both compressing both the horizontal and the vertical bonds.

The issue with prior art in MPDO simulations of noisy random circuits [127] is that such a simple contraction strategy (where both horizontal and vertical singular values are discarded below a certain threshold) leads to ever increasing bond horizontal bond dimensions, as illustrated in Figure 13 (lower panels) on the MPDO “LC” curve. This is due to the fact that the horizontal bond dimension carries not only entanglement within the system (as in regular MPS), but also possibly within the environment: in Figure 12(c), it is clear that two neighboring tensors share a bond that links a system-environment site with another. A major flexibility of MPDO, however, is the freedom in choosing a purification. In other words, one can attempt to reduce the "horizontal" entanglement by using the gauge freedom: $\rho = AA^\dagger = (AU)(U^\dagger A^\dagger)$ for any unitary U, which is equivalent to transforming $|\Psi_{\text{tot}}\rangle$ to $U|\Psi_{\text{tot}}\rangle$. One can optimize U to reduce the entanglement between environmental sites. This is in essence what we propose in [126]. Of course, one cannot optimize over a general $2^n \times 2^n$ unitary U. Rather, in the spirit of e.g. two-site DMRG, we optimize over two-site unitaries acting on two neighboring tensors. This leads to the suppression of the ever increasing horizontal bond dimension, as shown in Figure 13 (bottom row). The requisite bond dimensions for MPDOs is now similar to that of MPOs.

Circuit cutting: a hybrid tensor contraction strategy. Let us conclude this section by mentioning that tensor network contraction is also at the heart of work we have conducted to achieve the practical task of executing a quantum circuit with more qubits than the number of qubits available on the available quantum processor. In these works [128, 129], we examine a hybrid quantum classical scheme where the tensor network corresponding to the circuit is cut into subnetworks. Each subnetwork is then "contracted" by translating it back to a (small) quantum circuit that is executed on a quantum processor. The individual subnetwork results are combined by contracting obtained tensors. The exponential classical complexity of the overall task is then reduced to an exponential in the number of cuts needed to disect the whole tensor network into subnetworks whose corresponding circuits can fit on the available processor. Noteworthy, this technique, also called "circuit knitting", seems to be of importance in roadmaps for large-scale processors [130]. One main challenge is to balance the classical and quantum costs by optimizing the cut locations.

4.1.2. *Matrix product states as an antidote to decoherence: compressing quantum circuits*

In the previous section, we argued that matrix product states can be used to replicate finite-fidelity experiments with random quantum circuits, and to emulate noisy quantum circuits in a realistic fashion. In this section, we aim at showing that they can also be used to "give a leg up" to noisy quantum computers.

The goal of the method, presented in a recent publication ([131]), is to study the dynamics of a 1D transverse-field Ising model (TFIM)

$$H = \sum_{i=1}^{n} Z_i Z_{i+1} + h \sum_{i=1}^{n} X_i \tag{47}$$

after a sudden quench of the transverse field from 0 to a finite value h, starting from the antiferromagnetic spin state. The entanglement entropy S of the state of such a system typically increases linearly with time [30], making it a hard target for MPS methods, due to their $\chi \sim O(2^S)$ scaling. While it seems to be an easy target for quantum computing via a Trotterization of $\mathrm{e}^{-\mathrm{i}Ht}$, the exponential suppression of the fidelity presented in the previous sections poses a serious practical challenge. Indeed, to keep a fixed Trotter error, one needs to increase the number of Trotter steps with time, meaning a circuit depth also increasing with time. Therefore, the final fidelity (given by the product formula (34)) is going to decrease exponentially with time (in the absence of error correction), thus starkly degrading the quality of a purely quantum computation.

In [131], we propose to use a MPS and MPO-based time evolution to produce better (shallower) quantum circuits than those one would have obtained with a simple Trotter evolution. Two main stages, illustrated in Figure 14, have to be distinguished. The first stage consists in performing a standard TEBD time evolution up to the maximal time $t_{\text{max}} < t_{\text{f}}$ reachable by the classical computer. This part of the algorithm is constrained by the growth of entropy with time,

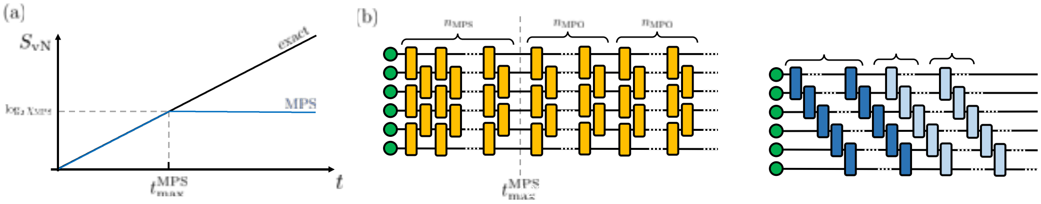

Figure 14. Sketch of the QMPS method. (a) Sketch of the evolution of the von Neumann entanglement entropy as a function of time. (b) TEBD evolution. Right: corresponding staircase layer quantum circuit. Adapted from [131]. Copyright (2024) by the American Physical Society.

which means that a large enough χ should be chosen to avoid large compression errors until, at some point, one reaches the random access memory limits of the classical processor. One then converts the so-obtained MPS into a quantum circuit, using recent techniques to perform this conversion [132]: we optimize a staircase circuit of N_l layers, with maximum bond dimension 2^{N_l} to have the largest overlap with the MPS state obtained with TEBD. Said circuit is much shallower than the circuit that would have been obtained by simple Trotterization. Intuitively, one could interpret this by saying that the TEBD procedure selects the circuit that will be the most efficient in terms of entropy creation (as opposed to a Trotter evolution, where smaller Trotter errors means more layers, each of which generates less entanglement than with larger, yet more discretization-error-prone, steps).

The second stage consists in first representing subsequent Trotter steps with a MPO, and then turning this MPO into a staircase circuit that is appended to the circuit obtained in the first stage.

We first perform noisy simulations of the performance of three different time evolution strategies: MPS-TEBD alone, quantum Trotter evolution alone, and our QMPSO hybrid method. The respective state fidelities are compared in the left and middle panels of Figure 15: for short times, MPS alone is the best method as it is essentially perfect (small times mean small enough bond dimensions), while the quantum computer will suffer from decoherence. For longer times, which method is best depends on the noise level: for large noise levels, as expected, MPS yields better fidelities as the quantum state is degraded by quantum noise. For very small noise levels, the quantum Trotter evolution is essentially perfect (up to Trotter errors, which can be made small because longer circuits will not suffer from very weak decoherence), while the MPS-TEBD evolution is plagued with large truncation errors (the maximum affordable bond dimension is smaller than what the entropy increase with time would dictate). Thus, the quantum Trotter evolution is the best method. In between, namely in a range between about 0.01% and 1% depolarizing error (supposing the maximum bond dimension on the classical computer is $\chi = 32$, an artificially small budget), our hybrid QMPSO method wins. This error range contains most NISQ hardware.

We test this prediction through experimental runs on a transmon processor of the IBM company. We compare a MPS-only algorithm (with a large compression error due to a limited bond dimension), a quantum-only algorithm (with a large error coming from decoherence and Trotter discretization), and our hybrid method, for a TFIM simulation on 10 spins (qubits). We observe, on the right panel of Figure 15, that the hybrid method indeed outperforms the other two methods.

Of course, this work calls for future improvements: in practice, advanced MPS implementations can reach much larger bond dimensions, so that the window for advantage using our hybrid method versus a purely quantum Trotter evolution is much smaller, namely it requires very low quantum error rates. A more realistic setting for a quantum advantage of quantum-enhanced

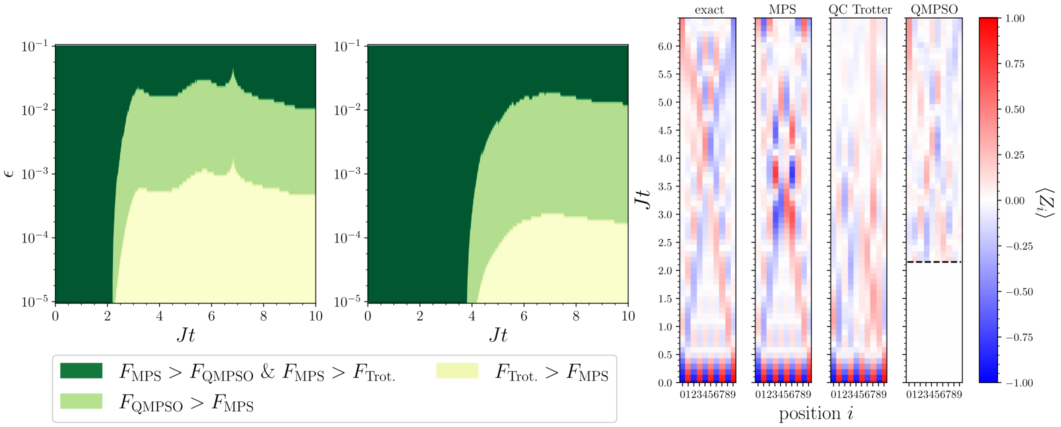

Figure 15. Results of the QMPS method. Left: theoretical diagram of the respective advantage zones of MPS (purely classical), QMPSO (hybrid classical + quantum) and Trotterization (purely quantum) as a function of depolarizing error rate ϵ and (dimensionless) evolution time Jt for $\chi = 8$ (left) and $\chi = 32$. Right: Evolution in time of the local magnetization for a $n = 10$ Ising chain. QC Trotter and QMPSO involve experimental runs on ibmq_kolkata hardware. Adapted from [131]. Copyright (2023) by the American Physical Society.

time evolution (whether hybrid or purely quantum) is time evolutions on two-dimensional lattices. There, MPS are quickly limited by their 1D structure, and PEPS face many difficulties, among which the fact that their exact contraction is exponentially costly, as opposed to MPS. Therefore, a hybrid method turning PEPS into quantum circuits [133, 134] appears very promising.

4.1.3. *Perspective: quantum advantage and tensor networks*

In the sections above, we explained how tensor networks could be used to assess and challenge the validity of quantum advantage claims made for a very specific task, that of sampling from the output distribution of random quantum circuits. These classical simulation efforts pushed back the boundary of quantum advantage, as evidenced in subsequent publications by e.g. Google's team like [93], who reassessed the classical simulation time to 6.18 s down from the original 10,000 years. However, hardware improvements were made both on the Sycamore processor and on a Chinese superconducting processor called Zuchongzhi [135]. On the former, the new bar is set to 47.2 years. Classical algorithms could yet again emerge to redefine this boundary, but the actual challenge of quantum processors lies elsewhere. Indeed, if quantum processors are to be deemed useful, they need to outperform classical processors for "useful" applications, be they as specialized as quantum many-body physics.

First claims in this direction were made by [136]. There, the problem at hand was a quench of a 127-spin transverse-field Ising model Equation (47) and the subsequent time evolution. Reference [136] claimed that their superconducting processor, after the appropriate error mitigation, could obtain essentially exact time-evolved observables in regimes where tensor network methods like MPS or isometric tensor networks failed. However, a few weeks after this publication, a series of publications appeared that succeeded in using classical simulation methods to reproduce the results of IBM's teams: [137] used a PEPS representation of the state with an approximate belief propagation algorithm, [138] used a sparse Pauli dynamics algorithm, [139] used a simple numerical mitigation strategy to extrapolate the experimental results from a 30-qubit simulation, while [140] performed a single-site dissipative mean-field computation and got reasonable agreement with the experimental results. The ease with which these various classical methods reproduced the experimental results can be interpreted as follows: the low connectivity of the IBM device (most qubits have 2 neighbors, a few 3 neighbors) make it an easy target for tensor networks, which are designed to perform well in low dimensions (like one dimension, cf. MPS), or graphs that are close to trees. This is the case for IBM's processor, which is locally very close to a tree. There, approximate PEPS contraction methods work very well. The level of noise in the device likely also plays an important role: it leads to smaller correlation lengths, which (i) justify why the large loops on IBM's device "look" like trees (the correlation length is not long enough that interferences within a loop can happen), (ii) explains why methods which rely on a small causal cone (like [139]) work well. Also, noise (in particular depolarizing noise) kills "Pauli paths" with high weight, making Pauli sparse dynamics techniques more efficient [138].

In the light of these observations, one can conclude that one of the main challenges for quantum processors today is their ability to generate states with a large correlation length despite decoherence.

In the next section, we turn to hybrid techniques that could help in the quest for this goal.

4.2. *Hybrid algorithms for fermionic problems*

In the previous subsection, we saw how tensor-network techniques could be used either to assess the potential advantage of quantum computers compared to classical methods, or as a way to build more noise-robust circuits to study many-spin problems such as the transverse-field Ising model.

In this section, we review our recent works on the use of quantum processors to tackle many-electron problems, whether on gate-based processors (Section 4.2.1) or on analog processors (Section 4.2.2). Our guiding principle will be to generate correlated states with the shortest possible circuits or time evolutions. This goal is all the more challenging with fermions as they come with complications due to their antisymmetry properties, which typically, through fermionic encodings, lead to more complex time evolutions.

4.2.1. *Optimizing the single-orbital basis*

As already discussed, one main limitation of current processors is the limited depth of the circuits that can be executed. In this subsection, we explore a way to reduce the depth required for performing many-fermion computations with quantum processors.

In near-term quantum algorithms like the variational quantum eigensolver (VQE, see discussion in Section 3.2.2), one main challenge is the choice of ansätze that are both expressive enough (so that they stand a chance of approximating the sought-after ground state) and short enough (so as to reduce the quality degradation induced by decoherence). In two recent works [141, 142], we present a method to keep the same expressivity while reducing the circuit depth. This gain is obtained by changing the orbital basis in which the problem is represented.

The link between fermionic orbitals and qubits is routinely made by fermion-to-spin transformations. There is a wide variety of them, which differ in the ratio of qubit count to orbital count and locality properties of the qubit Hamiltonian obtained after the transformation. Here, we will focus on the most straightforward transform, called the Jordan–Wigner transform, that we already mentioned previously. It maps fermionic orbital occupations (0 or 1 due to the Pauli principle) to qubit states ($|0\rangle$ or $|1\rangle$). The ith qubit is thus directly related to the ith orbital (for instance a site orbital $\phi_i(r)$ localized around the ith site in the Hubbard model, Equation (2)). In such a representation, a Fock state $\prod_{k=1}^{N_{\mathrm{el}}} c_{i_k}^\dagger |0\rangle$ (orbitals $i_1, \ldots, i_k$ are filled) is represented by a computational state with ones at locations $i_1, \ldots, i_k$ and zeros elsewhere. Such states are very easy to generate: a circuit with X gates on qubits $i_1, \ldots, i_k$ will create them (see Figure 16(b)). In other words, in the right basis (the Fock basis), a Fock state is a trivial computational basis state.

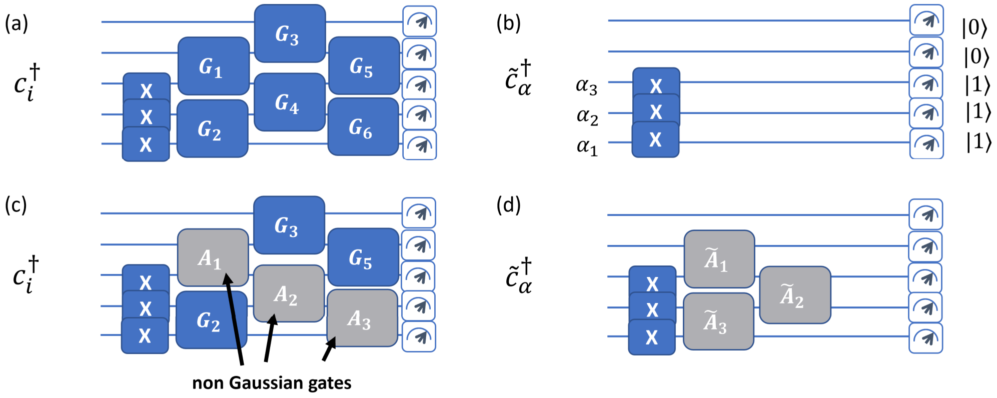

Figure 16. The importance of the orbital basis. (a,c) Original basis. (b,d) Optimized basis (natural orbitals). (a,b) Noninteracting case. (c,d) Interacting case. G_k denote Givens rotations (Gaussian gates).

This statement is not true in any other basis. For instance, our Fock state, $|\psi\rangle = \prod_{k=1}^{N_{\mathrm{el}}} c_{i_k}^\dagger|0\rangle$, relative to orbitals $\{\phi_i(r)\}_{i=1\ldots n}$, becomes much more complicated if expressed in another (rotated) basis $\varphi_\alpha(r) = \sum_i V_{\alpha i}\phi_i(r)$. Using $c_i^\dagger = \sum_\alpha V_{i\alpha}^\dagger \tilde{c}_\alpha^\dagger$, we see that $|\psi\rangle = \sum_{\alpha_1\ldots\alpha_n} A_{\alpha_1\ldots\alpha_n}\prod_{k=1}^{N_{\mathrm{el}}} \tilde{c}_{\alpha_k}^\dagger|0\rangle$ (with A a function of the V coefficients). If the qubits are mapped, via the Jordan–Wigner transformation, to the φ_α orbitals instead of ϕ_i orbitals, the circuit to build $|\psi\rangle$ is thus much more complex than the simple circuit with only X gates that we outlined above. This remark leads to the notion that the orbital basis can be optimized to reduce circuit depth (to construct one and the same state).

This statement easily extends to “mean-field states” or Hartree–Fock states, namely single Slater determinants: they can always be written in the form $\prod_{k=1}^{N_{\mathrm{el}}} \tilde{c}_{\alpha_k}^\dagger|0\rangle$, with $\tilde{c}_\alpha^\dagger$ creating an electron in the $\varphi_\alpha(r)$ orbital. In this basis, the corresponding circuit will consist only of single-qubit rotations (Figure 16(b)). In other bases, a circuit made up of so-called Givens rotations [143], which are entangling gates, will have to be used (see Figure 16(a)). This class of gates essentially implements the orbital rotation V in the many-body Fock space. In other words, to prepare a Slater determinant, either one does not use the right orbital basis and uses a potentially long and thus error-prone circuit with Givens rotations, or one changes the orbital basis of the qubits so that the corresponding circuit becomes trivial.

For strongly-correlated states (or multireference states in a quantum chemistry context), no orbital rotation can lead to a single Slater determinant and hence to a trivial circuit: one expects that they will require more complex circuits, since they are linear combinations of (usually a large number of) uncorrelated (product) states, and are thus highly entangled states. However, one also expects the orbital basis to play a role in the complexity of the corresponding state preparation circuit.

For these generic states (as opposed to single Slater determinants), no clear mathematical statement as to the optimal single-particle basis has been made, to our best knowledge. However, quantum chemists have long used a particular basis called the natural-orbital (NO) basis [33]. It is defined relatively to a given state $|\psi\rangle$ as the basis that diagonalizes the 1-RDM (which we already encountered, see Equation (23)):

$$D_{ij} = \langle\psi|c_i^\dagger c_j|\psi\rangle.$$

This basis is usually loosely claimed to be the basis that minimizes the number of Slater determinants required to represent a given state $|\psi\rangle$. While this is strictly true for Slater determinants

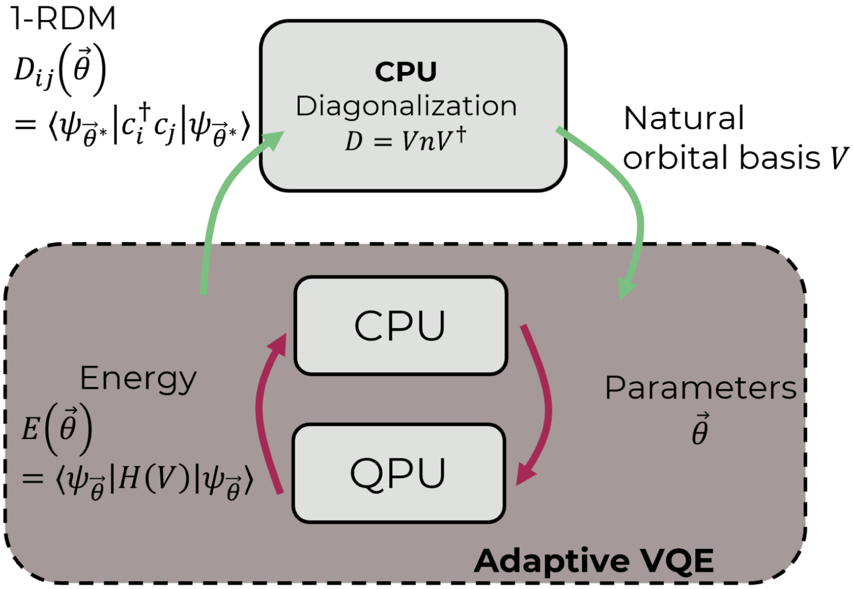
1-RDM
$D_{ij}(\vec{\theta}) = \langle \psi_{\vec{\theta}^*} | c_i^\dagger c_j | \psi_{\vec{\theta}^*} \rangle$
CPU
Diagonalization
$D = VnV^\dagger$
Natural orbital basis V
CPU
QPU
Energy
$E(\vec{\theta}) = \langle \psi_{\vec{\theta}} | H(V) | \psi_{\vec{\theta}} \rangle$
Parameters
$\vec{\theta}$
Adaptive VQE

Figure 17. Sketch of the (adaptive) NOization procedure.

as discussed in the previous paragraph, it is unclear that it is true for generic states (in fact it is proven for two-electron states but probably wrong for more-than-two-electron states).

Despite these caveats, we expect that this basis will provide the simplest possible circuit implementation for the state preparation. In a way, rotating to the NO basis should strip the "unnecessary" Givens rotations from the circuit, as illustrated in Figure 16(c,d).

In practice, to perform the computation in the NO basis, one needs to compute the 1-RDM D, which requires the knowledge of $|\psi\rangle$, and then diagonalize $D = VnV^\dagger$. The basis $\chi_\alpha = \sum_i V_{\alpha i}\phi_i$ is then the NO basis. Yet, in chemistry (and in VQE), the state of interest, $|\psi\rangle$, is unknown since it is precisely the ground state that one is looking for. We thus propose the following iterative procedure (inspired by e.g. [144, 145]) to find an approximate NO basis, as illustrated in Figure 17:

(1) Start from the original (e.g. site) basis ϕ_i.
(2) Perform a VQE step to obtain an approximate ground state $|\psi(\theta^*)\rangle$ in this basis
(3) Compute the 1-RDM corresponding to this state. Diagonalize it to find the orbital-rotation matrix V.
(4) Transform the qubits (i.e., in practice, the Hamiltonian H on which the VQE is applied) to the basis V and go back to step 2 until convergence.

We applied this method to the Anderson impurity Hamiltonian (21) in [141] and the Hubbard model in [142], and find that (i) the iterative procedure (dubbed NOization) usually converges to the true NO basis, and (ii) one can afford to use shorter VQE ansatz circuits when using this basis than when using the original basis (which is uneconomical in terms of the number of Slater determinants to represent the ground state, and thus the complexity of the corresponding circuit). This shortening of the circuits in turn leads to a better robustness to noise.

In [142], we refined the method by (i) studying its robustness to shot noise and (ii) combining it to the recently proposed adaptive-VQE schemes [103, 104].

The importance of point (i) is linked to the fact that changing the orbital basis comes at a cost: in general, it increases the number of Pauli terms in the Pauli representation of the Hamiltonian at hand. For instance, while the Hubbard model (Equation (2)) has $O(n)$ Pauli terms

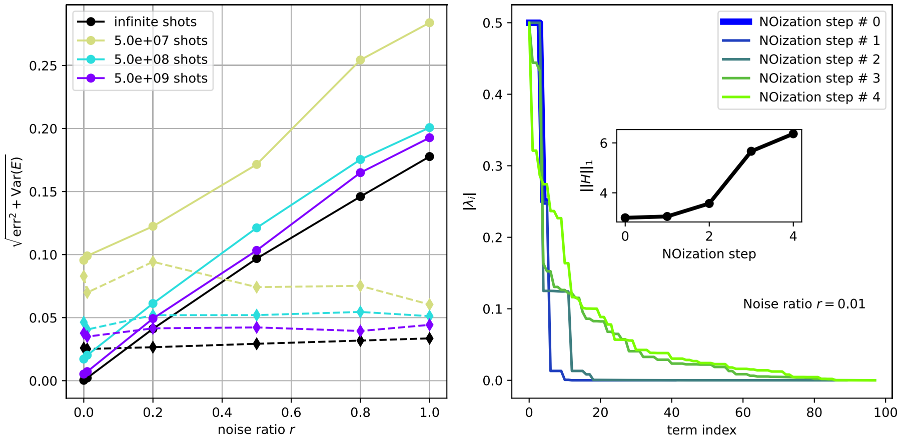

Figure 18. Interplay of shot noise and natural orbitalization. Left: relative VQE error (including bias and variance) as a function of the noise ratio to typical NISQ levels of noise for a standard VQE run with a LDCA ansatz (solid lines) compared to a natural-orbitalization strategy with the fSim circuit (dashed lines), half-filled Hubbard dimer, $U/t = 1$. Right: distribution of the Pauli coefficients $|\lambda_i|$ for successive NOization steps at $r = 0.01$. Inset: one-norm as a function of the NOization step. Reproduced from [142].

after a Jordan–Wigner transformation, after a generic orbital transformation, it can have up to $O(n^4)$ terms. This should play a role when computing the variational energy $E(\boldsymbol{\theta})$ using a finite number of shots. In particular, we need to assess whether the statistical precision for a given shot budget does not suffer too much from a rotation to the (approximate) natural orbital basis. We observe (Figure 18(left)) that while NOization does reduce the sensitivity to decoherence noise (one observes a linear bias with increasing noise ratio r for standard VQE, while with NOization, which can afford a much shallower circuit, noise has virtually no impact), it also does not increase sensitivity to shot noise, at least in this example. This can be ascribed to the quite sharply peaked distribution of the Pauli coefficients despite their increased number (Figure 18(right)). This leads to a one-norm (shown in the inset) that does not significantly increase upon orbital rotation, and thus a similar shot noise (since the one-norm upper bounds shot noise, see Equation (38)).

Let us now focus on the use of adaptive techniques (ii). The goal here is to take advantage of the fact that rotating the basis in principle allows for shorter circuits. However, with standard VQE, this means changing the circuit "manually" after each rotation. Adaptive techniques should solve this issue by automatically constructing a circuit that is well suited to the new orbital basis. Adaptive methods iteratively construct the variational circuit by adding new gates based on their potential to decrease the variational energy at the next step, which is measured by the corresponding energy gradient. In particular, they come with a natural stopping criterion: the ansatz construction stops when the gradients vanish, i.e. when a minimum (hopefully a global one) of the energy landscape is reached. In [142], we observe that the size (number of gates) of the iteratively constructed ansatz decreases when gradually rotating to the approximate natural-orbital basis, and that the energy reached by the method gets closer to the ground state energy at the same time (Figure 19).

This investigation of the gradual transformation of the orbital basis to the natural orbitals shows the promising potential of the method. Much remains to be done: the true target application is the Anderson impurity model, with the goal of successfully preparing a good ground state and then computing the corresponding 1-RDM (for DMET or RISB) or Green's function (for DMFT). This orbital rotation could be combined with a recent proposal by [146]

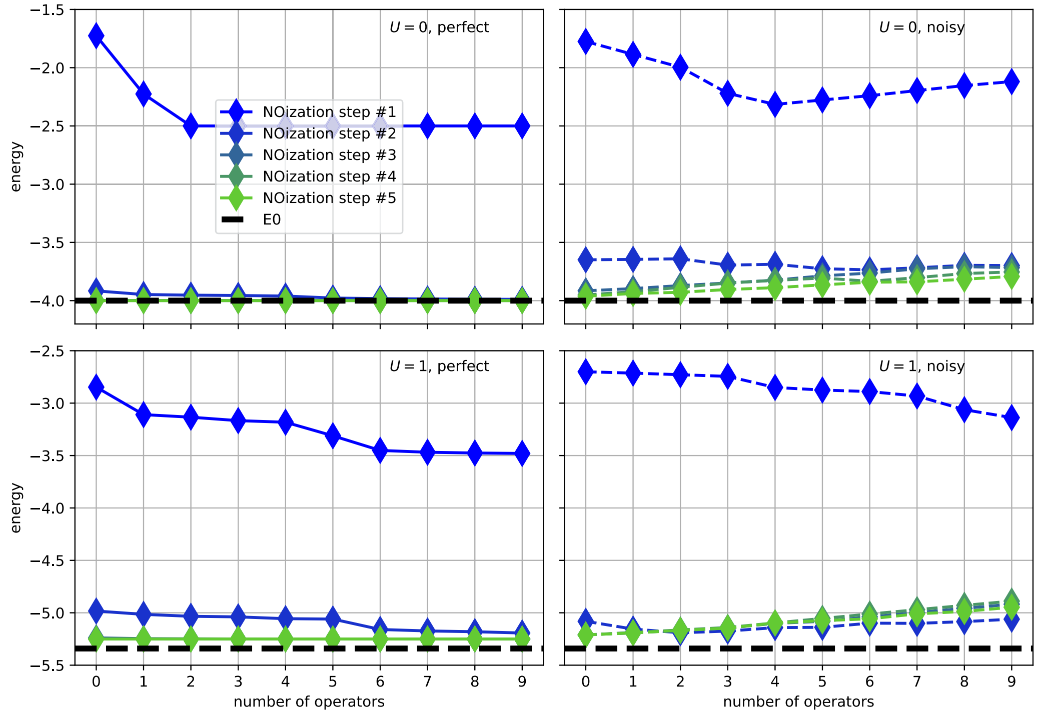

Figure 19. Results of the NOization procedure with an adaptive ansatz construction: converged VQE energy as a function of the ADAPT-VQE step (one operator in the ADAPT-VQE pool corresponds to a few additional gates in the ansatz). Reproduced from [142].

(similar to the QMPS method of Section 4.1.2) to use a tensor network method to compute an approximate AIM ground state and then load it on the quantum processor to compute the Green's function.

4.2.2. *Rydberg atoms for Mott physics through a slave-spin mapping*

Most quantum algorithms for studying fermionic many-body problems use (i) fermion-to-qubit encodings like the Jordan–Wigner transformation, and (ii) are for designed gate-based quantum processors. In this section, we explore a method that avoids the overheads of the encodings and that is particularly well-suited (but by no means restricted to) analog Rydberg processors.

Rydberg processors. These processors are described by the following Hamiltonian [62]:

$$\hat{H}_{\text{Rydberg}} = \sum_{i \neq j} \frac{C_6}{|\mathbf{r}_i - \mathbf{r}_j|^6} \hat{n}_i \hat{n}_j - \hbar\delta(t) \sum_i \hat{n}_i + \frac{\hbar\Omega(t)}{2} \sum_i \hat{S}_i^x, \tag{48}$$

where $C_6/|\mathbf{r}_i - \mathbf{r}_j|^6$ is the van der Waals interaction between atoms located at positions $\mathbf{r}_i$ and $\mathbf{r}_j$, $\delta(t)$ and $\Omega(t)$ are the detuning and Rabi drives, respectively. The atoms are described as an assembly of two-level systems where level $|0\rangle$ is the atom at rest and $|1\rangle$ is a highly-excited Rydberg state of the atom.

"Programming" such a processor consists in finding ways to manipulate the quantum state of the machine using this Hamiltonian, with three main knobs: the positions $\{\mathbf{r}_i\}$ of the atoms, the Rabi drive $\Omega(t)$ and the detuning drive $\delta(t)$. The Hamiltonian (48) realized experimentally in Rydberg processors—essentially an antiferromagnetic transverse-field Ising model (TFIM)—represents a difficult many-body problem to simulate for classical computers, especially in out-of-equilibrium situations. This difficulty results from to the combination of a large number of particles (196), long correlation lengths (7 lattice spacings) and true two-dimensional geometry reported in recent experiments [64]. These properties make their classical simulation by the most advanced tensor-network techniques (like MPS or PEPS) a very tall order (as opposed to the TFIM of [136], which is defined on quasi-1D heavy-hex geometry).

As we will discuss below in Section 4.3.2, this Hamiltonian has "straightforward" candidate use cases such as the unit-disk maximum independent set problem. However, for fermionic many-body problems that are the focus of this section, one must reconcile two seemingly very different problems: an interacting fermion problem and an interacting spin problem. Although there exists straightforward transformations from fermionic variables to spin variables (as briefly explained in Section 3.1.4), they all come with an overhead in terms of circuit depth either because they give up on locality or because they require additional qubits: On the one hand, the most widespread transformations like the Jordan–Wigner transformation, parity or Bravyi–Kitaev transformations (see e.g. [147] for a unified view) lead to a more-or-less drastic loss of locality of the Hamiltonian. For instance, as we already saw before, a $c_i^\dagger c_j$ term leads to terms like $X_i Z_{i+1} \cdots Z_{j-1} X_j$ terms in the corresponding spin Hamiltonian. This in turn leads to long circuits to implement e.g. time evolutions with $\mathrm{e}^{-\mathrm{i} X_i Z_{i+1} \cdots Z_{j-1} X_j t}$ operators, which require a $O(n)$ depth ($O(\log(n))$ for the Bravyi–Kitaev transformation). On the other hand, other techniques like the Verstraete–Cirac transformation [148] require auxiliary qubits, and thus ultimately longer circuits since deeper circuits are needed to entangle more qubits.

What is more, the analog Rydberg platform we are considering comes with a controllable but fixed Hamiltonian that differs from the spin Hamiltonian resulting from the aforementioned fermion-qubit transformations.

A slave-particle method to disentangle fermionic and spin degrees of freedom. In a recent work [149], we propose a way to circumvent the fermion-to-spin conversion overhead by resorting to an existing technique called the slave-spin method [150–152], a simplification of the slave-boson method we briefly mentioned in Section 2. In its $\mathbb{Z}_2$ flavor [151], it consists in rewriting the original fermionic creation operators $c_{i\sigma}^\dagger$ as

$$c_{i\sigma}^\dagger = S_i^z f_{i\sigma}^\dagger, \tag{49}$$

where S_i^z is the Pauli-z spin matrix, and $f_{i\sigma}^\dagger$ creates a (pseudo) fermion at site i. Since the corresponding Hilbert space is larger than the original one, one imposes contraints to project states back onto a so-called physical Hilbert space in one-to-one correspondence to the original. The constraints read

$$S_i^x + 1 = 2(n_i^f - 1)^2 \tag{50}$$

at each site (with $n_i^f = \sum_\sigma f_{i\sigma}^\dagger f_{i\sigma}$), so that only 4 states among the 8 possible local states are allowed, as in the original model. These four states (left column below) as well as the corresponding original states (right column) are:

$$\begin{aligned} |0\rangle|+\rangle_x &\leftrightarrow |0\rangle \\ |\uparrow\rangle|-\rangle_x &\leftrightarrow |\uparrow\rangle \\ |\downarrow\rangle|-\rangle_x &\leftrightarrow |\downarrow\rangle \\ |\uparrow\downarrow\rangle|+\rangle_x &\leftrightarrow |\uparrow\downarrow\rangle. \end{aligned}$$

The method thus a priori consists in tackling this larger model and then projecting it back onto the subspace satisfying (50). Approximate strategies to handle the model in this larger space will be needed (as for the original Hubbard model), but the usual conjecture is that approximations

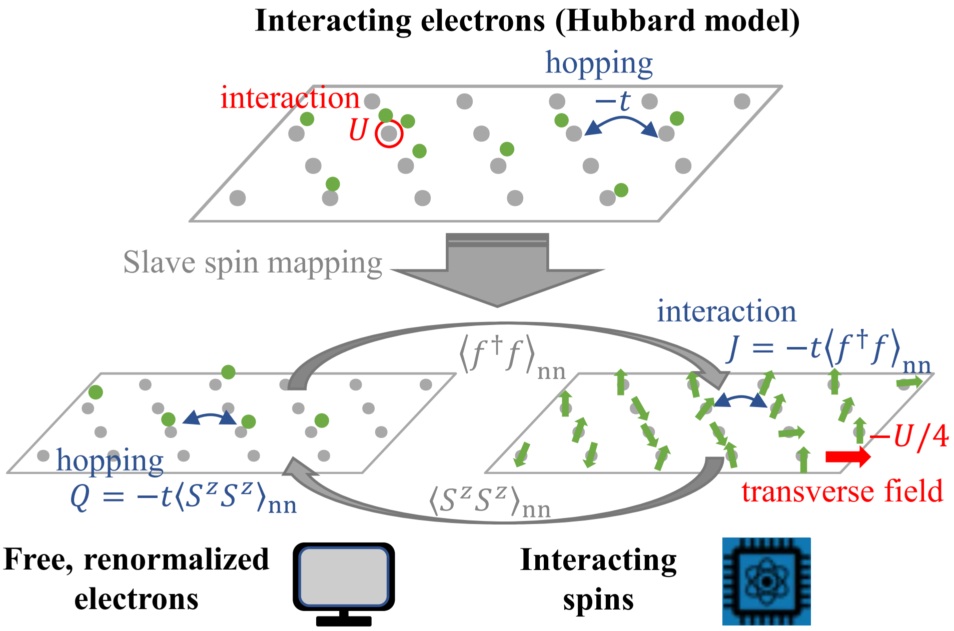
Interacting electrons (Hubbard model)
hopping
interaction
$-t$
U
Slave spin mapping
interaction
$J = -t\langle f^\dagger f\rangle_{\mathrm{nn}}$
$\langle f^\dagger f\rangle_{\mathrm{nn}}$
$-U/4$
transverse field
hopping
$Q = -t\langle S^z S^z\rangle_{\mathrm{nn}}$
$\langle S^z S^z\rangle_{\mathrm{nn}}$
Free, renormalized electrons
Interacting spins

Figure 20. Sketch of the slave-spin method with a quantum processor. Reproduced from [149]. Copyright (2024) by the American Physical Society.

in the larger space will lead to better results than a similar level of approximation in the original model.

Plugging Equation (49) into the original Hubbard model (Equation (2)) and using the constraint (50) leads to

$$H = \sum_{ij\sigma} t_{ij} S_i^z S_j^z f_{i\sigma}^\dagger f_{j\sigma} + \frac{U}{4}\sum_i S_i^x. \tag{51}$$

We now perform a mean-field approximation to decouple the fermion-spin coupling terms:

$$S_i^z S_j^z f_{i\sigma}^\dagger f_{j\sigma} \approx \langle S_i^z S_j^z\rangle f_{i\sigma}^\dagger f_{j\sigma} + S_i^z S_j^z \langle f_{i\sigma}^\dagger f_{j\sigma}\rangle + \mathrm{const.}$$

This yields

$$H \approx \left\{\sum_{ij\sigma} Q_{ij} f_{i\sigma}^\dagger f_{j\sigma}\right\} + \left\{\sum_{ij} J_{ij} S_i^z S_j^z + \frac{U}{4}\sum_i S_i^x\right\}, \tag{52}$$

namely the Hamiltonian turns into the sum of a free-electron Hamiltonian with renormalized hopping $Q_{ij} = t_{ij}\langle S_i^z S_j^z\rangle$ and a transverse-field Ising model with Ising coupling $J_{ij} = \sum_\sigma t_{ij}\langle f_{i\sigma}^\dagger f_{j\sigma}\rangle$. The ground state of H reads $|\Psi\rangle = |\psi_{\mathrm{f}}\rangle \otimes |\psi_{\mathrm{s}}\rangle$ where $|\psi_{\mathrm{f}}\rangle$ and $|\psi_{\mathrm{s}}\rangle$ are the respective ground states of the fermionic and spin Hamiltonians. They need to be computed in a self-consistent fashion because the hopping and Ising interaction matrices depend on the solution through $\langle S_i^z S_j^z\rangle$ and $\langle f_{i\sigma}^\dagger f_{j\sigma}\rangle$. Thus, the slave-spin decoupling translates (at the cost of the mean-field decoupling of the fermion and spin variables) the Hubbard model into a free fermionic model (which carries the fermionic nature of the original problem) self-consistently coupled to an interacting spin model (which carries the interacting nature of the original problem), as illustrated in Figure 20.

While former (purely classical) approaches [151, 153, 154] introduced a further single-site (or small cluster, [155]) mean-field approximation to solve the spin model, we propose to tackle it instead with a quantum processor: indeed, the spin model

$$H_\mathrm{s} = \sum_{ij} J_{ij} S_i^z S_j^z + \frac{U}{4} \sum_i S_i^x \tag{53}$$

is very similar to the Hamiltonian realized by Rydberg atoms, Equation (48). The main deviation lies (i) in the finite number of atoms of Equation (48), as opposed to the thermodynamic size of Hubbard's Hamiltonian and hence H_s, and (ii) in the specific form $C_6/|\mathbf{r}_i - \mathbf{r}_j|^6$ of the spin interaction in the Rydberg platform.

We propose to handle the first issue by resorting to a cluster mean-field approach to H_s (like in [155]), which gives rise to

$$H_\mathrm{s}^{\mathscr{C}} = \sum_{i,j\in\mathscr{C}} J_{ij} S_i^z S_j^z + \frac{U}{4} \sum_{i\in\mathscr{C}} S_i^x + \sum_{i\in\mathscr{C}} h_i S_i^z, \tag{54}$$

where $h_i = 2z_i \overline{J}\,\overline{m}$, $\overline{J} = 1/N_p \sum_{\langle i,j\rangle\in\mathscr{C}} J_{i,j}$ and $\overline{m} = 1/N \sum_{i\in\mathscr{C}} \langle S_i^z\rangle$. $\mathscr{C}$ denotes the cluster of sites, whose size is fixed by the number of available atoms. $H_\mathrm{s}^{\mathscr{C}}$ thus has the same size as H_Rydberg.

The second issue is solved in an approximate fashion by optimizing the locations $\{\boldsymbol{r}_i\}_{i\in\mathscr{C}}$ of the atoms to have $C_6/|\mathbf{r}_i - \mathbf{r}_j|^6$ match J_{ij}. For the small sizes we tackled, the optimization converges well, although there will probably always be some residual mismatch due to the $1/r^6$ tail of the van der Waals interaction (as opposed to the nearest-neighbor-only character of J_{ij}).

Quasiparticle dependence on interaction at equilibrium. At equilibrium (and $T = 0$), the main computational step consists in computing the correlation function $\langle S_i^z S_j^z\rangle$ (needed in the self-consistent loop) in the ground state of $H_\mathrm{s}^{\mathscr{C}}$ using H_Rydberg as a proxy. This can be done by resorting to an annealing procedure on the Rydberg platform, namely we slowly ramp up the value of the interaction (therefore of Ω) to its final value to prepare the (approximate) ground state of $H_\mathrm{s}^{\mathscr{C}}$. As for any annealing technique, longer annealing times will lead to a larger success probability, namely a larger overlap of the state generated by this ramp with the sought-after ground state. In practice, because the hardware suffers from decoherence, long anneal times will lead to more errors. It is thus vital that one check that annealing can succeed in realistic times given the hardware noise levels. To answer this question, we model noise in the Rydberg platform using a Lindblad master equation

$$\mathrm{i}\hbar \frac{\mathrm{d}}{\mathrm{d}t}\rho = \left[H_\mathrm{Rydberg}(t), \rho\right] - \frac{\mathrm{i}}{2} \sum_k \left\{L_k^\dagger L_k, \rho\right\} - 2L_k \rho L_k^\dagger, \tag{55}$$

where the L_k are called jump operators (k has a priori no physical meaning yet, it just labels the operators). In the specific case of Rydberg platforms, dephasing noise is an important source of noise. It can be modelled using $L_k = \sqrt{\gamma} Z_k$ (where k now labels a Rydberg atom, and γ is the dephasing strength, which can be fitted to experiments (see e.g. [156])). Here, we take $\gamma = 0.02$ MHz. We also include false positive and negative readout errors (with a 3% rate corresponding to today's experimental situation).

Performing this time evolution to get the $\langle S_i^z S_j^z\rangle$ correlation function and performing the whole cycle self-consistently yields the left panel of Figure 21 for different mean-field cluster sizes. We observe that noise does cause a deviation from exact diagonalization results, but that we can predict a reasonable estimate for the Mott transition's critical interaction (within the slave-spin approximation) by computing the U-dependence of the quasiparticle weight Z. The latter, in the slave-spin mapping, can be accessed via the squared magnetization $Z \approx \overline{m}^2$ of the spin model. For $n = 12$, the computations are not converged with respect to cluster sizes, larger sizes are thus required. As for classical techniques, exact diagonalization can probably reach up to 50 atoms, and MPS techniques can be pushed to 100 atoms [64]. On the quantum side,

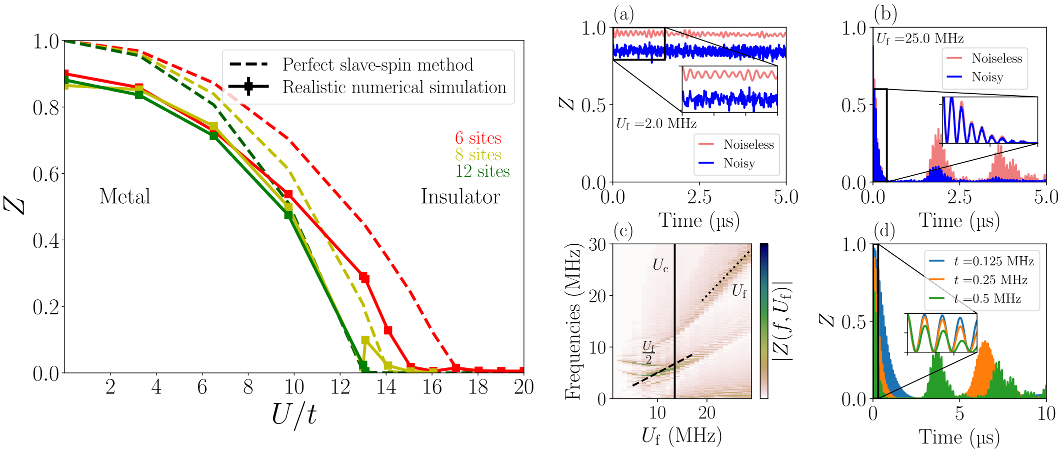

Figure 21. Hubbard physics with Rydberg platforms. Left: evolution of the quasiparticle weight as a function of the interaction strength for various cluster sizes (6, 8 and 12) with an exact diagonalization of the spin Hamiltonian (dashed lines) and a noisy annealing algorithm with a Rydberg Hamiltonian (solid lines). Right: Dynamics of the quasiparticle weight after an interaction quench with $U_f < U_c$ (top left) and $U_f > U_c$ (top right) on a noiseless and a noisy platform. Fourier transform of $Z(t)$ for various interaction strengths (bottom left) (noiseless case) and influence of the hopping (noiseless case). Reproduced from [149]. Copyright (2024) by the American Physical Society.

sizes of up to 200 atoms are now feasible [64]. Whether these atoms can be manipulated in a useful way depends on the interaction regime and the quality of the hardware: intuitively, the target is regimes where the true spin–spin correlation length is large, so that classical methods cannot capture it, and hardware platforms that are good enough that they can sustain these long correlation lengths. To precisely delineate this target, a better understanding of the dependence of the correlation length on hardware noise is needed. (As for finding regimes with a large correlation length, this is an easier task as the Mott transition corresponds, through the slave-spin mapping, to the ferromagnetic transition of the TFIM, where the correlation length is expected to diverge, see e.g. [157] for a PEPS investigation of this transition).

Dynamics of the quasiparticle weight after an interaction quench. A similar procedure can be applied in the out-of-equilibrium case [153]. Through the slave-spin mapping, an interaction quench in the Hubbard model translates to a tranverse-field quench in the spin model. One can thus quench the Rabi field to $\Omega = U/4$ in H_{Rydberg} to study the quench-induced dynamics of the quasiparticle weight $Z(t)$ in the Hubbard model.

The right panel of Figure 21 displays the resulting $Z(t)$ dynamics for a perfect Rydberg platform and in the presence of imperfections, simulated by the aforementioned Lindblad equation. One can see that the main phenomenology of such quenches can be resolved even in the presence of noise: U oscillations with hopping-enhanced damping in quenches to the Mott phase, $U/2$ and other oscillation frequencies in quenches to the Fermi-liquid phase.

This first study is limited to the emulation of a 12-atom Rydberg platform. In the future, experimental runs with more atoms should lead to the observation of temporal and spatial dynamics of the quasiparticle weight. Such dynamics are arguably hard to simulate with classical computers given the number of atoms and correlation lengths attainable by current platforms [64], so that the introduced mapping could lead to the investigation of dynamical properties of the

Hubbard model previously unreachable by classical means. The method is not limited to analog quantum computers. Yet, implementing it on an analog quantum computer like Rydberg atoms allows to avoid the discretization errors inherent to gate-based computers, like Trotterization (see Section 3.1.4).

Of course, several aspects of the work can be improved: away from the particle-hole symmetric case studied here (by working at half-filling on a bipartite lattice), the fulfilment of the constraint needs to be taken into account explicitly. Multi-orbital models (see [150]) will also require some adaptations as the slave-spin model H_s is no longer directly that of the Rydberg atoms, Equation (48).

4.2.3. *Quantitative criteria for fermionic problems*

In the previous subsections of this section on fermionic problems, we have introduced algorithms and tested them on small-size problems. A natural question to ask is whether these methods scale up to large sizes. In [158], we try to answer this question for two major methods that we already introduced, namely the variational quantum eigensolver (VQE) and the quantum phase estimation (QPE) algorithms.

The scale of noise in VQE... As already argued in Section 3.2.2, VQE intrinsically suffers from optimization issues and statistical shot noise. Here, we will neglect these issues and instead focus on the influence of decoherence. We will thus suppose we have found parameters $\boldsymbol{\theta}^\star$ such that $|\Psi(\boldsymbol{\theta}^\star)\rangle = |\Psi_0\rangle$, where $|\Psi_0\rangle$ is the ground state of the Hamiltonian at hand. For a parametric circuit with N_g gates, noise generically turns our perfect state $|\Psi_0\rangle$ into a mixed state ρ with a fidelity

$$F = \langle\Psi_0|\rho|\Psi_0\rangle \ll 1$$

with the ground state. We can write, generically,

$$\rho = F|\Psi_0\rangle\langle\Psi_0| + (1-F)\rho_{\text{noise}},$$

where ρ_{noise} is the (unknown) state generated by the noise. Thus, the energy we can compute is (in the absence of shot noise) $E = FE_0 + (1-F)E_{\text{noise}}$ with $E_{\text{noise}} = \mathrm{Tr}(\rho_{\text{noise}}H)$. In other words, noise introduces a bias

$$\Delta E = E - E_0 = (1-F)(E_{\text{noise}} - E_0). \tag{56}$$

We can relate the final fidelity to the individual gate error rates ϵ using the product formula we already encountered (Equation (34)), which we rewrite, for our current purpose, as $F = \mathrm{e}^{-\epsilon N_g}$. We now need to assume that $\epsilon N_g \ll 1$(otherwise, our final state is essentially noise and we have no chance of getting anywhere close to the solution), so that $1 - F \approx \epsilon N_g$.

The success of VQE is quantified by the targetted accuracy on the energy. If we target $\Delta E \leq \eta$, it means that our hardware error rate should satisfy

$$\epsilon \leq \frac{\eta}{(E_{\text{noise}} - E_0)\, N_g}. \tag{57}$$

This gives us a quantitative estimate of the hardware quality ϵ needed to reach a given accuracy η on the energy. The dependence on the size of the system (number of electrons or orbitals n) comes from E_{noise}, E_0 and N_g. The ground state energy E_0 is an extensive quantity, $E_0 \propto n$. The number of gates in the ansatz should probably increase with n. We now argue that the most severe effect comes from the n-dependence of E_{noise}.

To this aim, we need to focus on a specific case. We assume the noise to be a global depolarizing channel (Equation (33)). This choice may look arbitrary, but any long enough circuit (that it forms a so-called 2-design) will produce global depolarizing noise , so that this

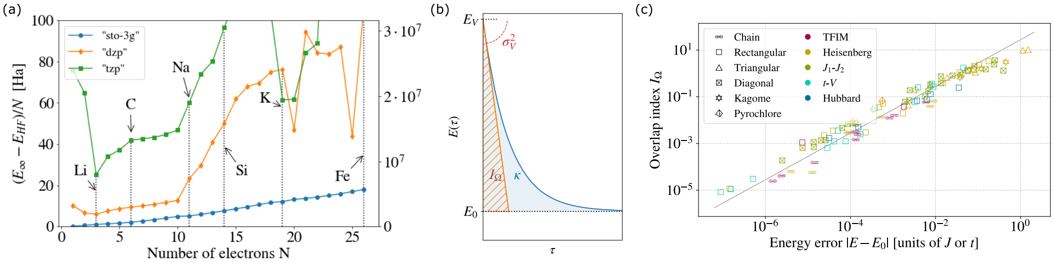

Figure 22. Quantitative criteria for VQE and QPE. (a) $(E_\infty - E_{\mathrm{HF}})/N$ as a function the number of electrons N for various molecules and basis sets. (b) Sketch of $E(\tau)$ as a function of imaginary time τ and definition of κ (area under the curve) and I_Ω (triangle defined by the slope at $\tau = 0$). (c) Overlap index I_Ω as a function of the energy error for various classical computations for various models. Adapted from [158].

choice is quite representative of many current ansatz circuits. For this noise, $\rho_{\mathrm{noise}} = I/2^n$, and therefore

$$E_{\mathrm{noise}} = \mathrm{Tr}(H)/2^n = \mathrm{Tr}(\rho_\infty H),$$

where ρ_∞ is the thermal state $\rho_T = \mathrm{e}^{-H/k_{\mathrm{B}}T}/Z$ at infinite temperature. In other words, global depolarizing noise populates very high energy states.

Let us now turn to the scaling of E_{noise}. In chemical systems, in the absence of screening of the charge of electrons, the Coulomb energy scales as n^2 (each electron interacts with all other electrons). In the ground state, screening reduces this scaling to $\propto n$. This means that noise brings in excited states whose scaling with size is no longer $\propto n$ but $\propto n^2$, a dramatic change of scaling. To make this statement more quantitative, we computed $(E_\infty - E_{\mathrm{HF}})/n$ (with E_{HF} a replacement for exact energy but, given the energy scales, $E_{\mathrm{HF}} - E_0$ is negligible) as a function of n for the first few elements of the periodic table. The results, shown in Figure 22(a), do display a behavior $E_\infty - E_{\mathrm{HF}} \propto n^2$. In addition, the scale of $E_\infty - E_{\mathrm{HF}}$ itself is very large, namely tens of Hartrees, to be compared to the chemical accuracy, $\eta = 1$ mHa. Assuming 100 gates (a very conservative number) and $E_{\mathrm{noise}} - E_0 = 10$ Ha (also a conservative estimate), this gives

$$\epsilon \leq 10^{-6}\text{–}10^{-4}\%,$$

at least three orders of magnitude from current noise rates (superconducting processors achieve $\epsilon \approx 0.1\%$ for two-qubit gates [93]).

A criterion to estimate overlaps for QPE. We now turn to a central question pertaining to the quantum phase estimation algorithm (QPE), which is often hailed as the go-to replacement of VQE when fault-tolerant quantum computers will be available: how large is the overlap Ω (Equation (58)) of the state that is input to QPE? Since QPE's complexity scales as $O(1/\Omega)$, knowing how Ω scales with system size is a crucial issue. While, for weakly correlated molecules, it seems that large enough overlaps can be obtained using classical methods, for Hubbard-like models in a regime of large interactions, the overlaps obtained by classical methods seem to be much smaller [86], calling for a quantitative examination of this question [87].

Here, we provide a way of computing this overlap with the only knowledge of the energy and variance of the input state, as well as some estimate of the exact ground state energy. The former two are typical outputs of many classical methods (like variational Monte-Carlo methods, see Section 2.1.3) and can also be estimated in VQE, if VQE is used to prepare inputs to QPE. The latter can usually be estimated by extrapolating e.g. coupled cluster computations with increasing excitation order (from CCS to CCSD to CCSDT …).

Let $|\Phi_0\rangle$ denote a variational state obtained e.g. by VQE (or by a classical variational method), and $|\Psi_0\rangle$ denote the exact ground state wavefunction. We want to estimate the overlap Ω. Reference [159] established the following identity:

$$\Omega = \mathrm{e}^{-\int_0^\infty \mathrm{d}\tau [E(\tau)-E_0]}, \tag{58}$$

with $E(\tau)$ the energy of imaginary-time evolved state $|\Psi(\tau)\rangle \propto \mathrm{e}^{-\tau H/2}|\Phi_0\rangle$, namely:

$$E(\tau) = \frac{\langle\Psi(\tau)|H|\Psi(\tau)\rangle}{\langle\Psi(\tau)|\Psi(\tau)\rangle} = \frac{\langle\Phi_0|H\mathrm{e}^{-\tau H}|\Phi_0\rangle}{\langle\Phi_0|\mathrm{e}^{-\tau H}|\Phi_0\rangle}.$$

The area under the $E(\tau) - E_0$ curve is hard to compute, but one can approximate it with the area of a triangle, as illustrated in Figure 22(b):

$$\int_0^\infty \mathrm{d}\tau\, [E(\tau) - E_0] \approx \frac{1}{2}(E_V - E_0)\frac{E_V - E_0}{|\partial_\tau E(\tau = 0)|}.$$

Let us compute $\partial_\tau E(\tau) = (-\langle\Phi_0|H^2\mathrm{e}^{-\tau H}|\Phi_0\rangle\langle\Phi_0|\mathrm{e}^{-\tau H}|\Phi_0\rangle + \langle\Phi_0|H\mathrm{e}^{-\tau H}|\Phi_0\rangle\langle\Phi_0|H\mathrm{e}^{-\tau H}|\Phi_0\rangle)/\langle\Phi_0|\mathrm{e}^{-\tau H}|\Phi_0\rangle^2$, and so

$$\partial_\tau E(\tau = 0) = \frac{-\langle\Phi_0|H^2|\Phi_0\rangle\langle\Phi_0|\Phi_0\rangle + \langle\Phi_0|H|\Phi_0\rangle^2}{\langle\Phi_0|\Phi_0\rangle^2} = -\langle H^2\rangle + \langle H\rangle^2$$

Therefore,

$$\Omega \approx \exp\left(-\frac{(E_V - E_0)^2}{2\sigma_V^2}\right), \tag{59}$$

namely we can compute an approximation to the overlap using the energy E_V and variance σ_V^2 of the input state, as well as the exact energy E_0 (or an estimate thereof), as claimed.

With this formula, we estimated the overlap of classical states obtained by a large sample of state-of-the-art methods applied on a variety of models [160]. The results are shown in Figure 22(c), where we plot the overlap index

$$I_\Omega = \frac{(E_V - E_0)^2}{2\sigma_V^2}. \tag{60}$$

We see that $I_\Omega \propto |E - E_0|$. Besides, we know that E, E_0 and σ_V^2 are extensive, therefore $I_\Omega \propto N$, and therefore $\Omega = \mathrm{e}^{-I_\Omega}$ decreases exponentially with system size for large variety of models and methods. This is not a surprising result: it is the infamous orthogonality catastrophe of condensed-matter physics. This catastrophe a priori renders QPE exponentially costly in the system size. Modifications of the algorithm, or other algorithmic advances, are thus probably needed.

4.3. *Beyond many-fermion problems: solving hard optimization problems with quantum processors*

We have focused so far on the solution of fermionic quantum many-body problems with quantum processors. In this section, we shed light on a seemingly less straightforward application of quantum processors, that is to solve *classical* optimization problems. More precisely, we will focus on combinatorial optimization problems. As we have seen in Section 2.3, they are nothing but classical many-body problems since they can be mapped to classical interacting spin problems. (We refer the reader to [161] for a review on general optimization problems and quantum computing).

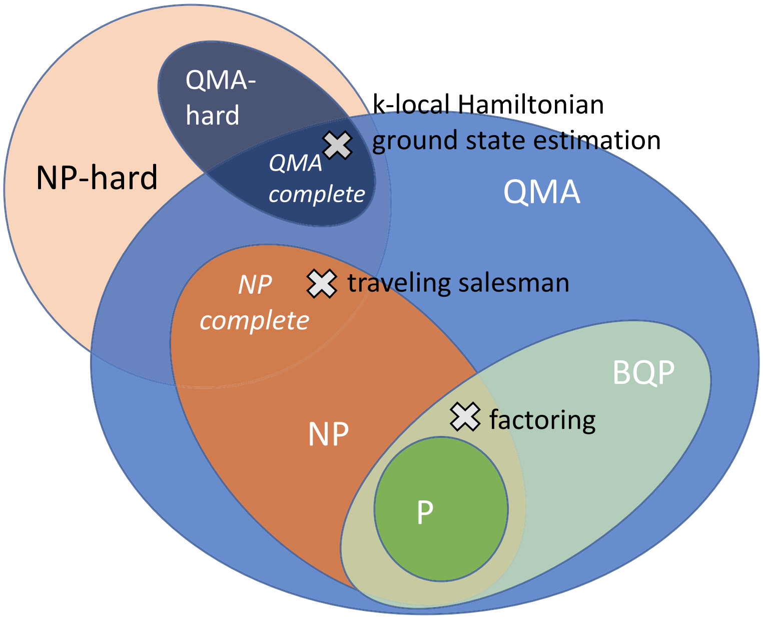

Figure 23. Complexity classes.

4.3.1. *A reminder on complexity theory… and what accelerations quantum computers can (or cannot) bring*

Before turning to concrete examples, let us first elaborate—although in a quite informal way—on the formal guarantees (or lack thereof) that quantum computers bring in terms of computational acceleration.

It is customary to classify computational problems—in fact, decision problems, corresponding to yes/no questions—using the notion of complexity classes. They—informally speaking—indicate the scaling of the time to solution of a given problem as a function of the problem size. On classical computers (classical deterministic Turing machines), the P class corresponds to the problems that can be solved in polynomial time. NP (for "nondeterministic polynomial" time) denotes problems that can be solved in polynomial time on *nondeterministic* Turing machines (which in practice captures problems that cannot be solved polynomially on deterministic machines… at least this is a very likely fact, known as the P ≠ NP conjecture in computer science). NP-hard problems are the problems that are at least as hard to solve as the problems in the NP class. Finally, the NP-complete class is the intersection of NP and NP-hard problems. They are very likely (that is, under the P ≠ NP conjecture) to have an exponential run time. These complexity classes are illustrated in Figure 23.

The same classification can be sketched for quantum computers. The polynomial quantum counterpart of P is called BQP (for bounded-error quantum polynomial time), while that of NP is QMA (for quantum Merlin–Arthur).

Let us take a look at a few of the famous problems that have been considered as promising for quantum computers:

- factoring is a NP problem (but not NP-hard), which is why it has a subexponential time classical algorithm (being subexponential does not prevent it from being hard to solve in practice, which guarantees the safety of encryption methods based on the RSA algorithm). Shor's algorithm is polynomial, which means that factoring is in BQP. It achieves an exponential speedup compared to the best known classical algorithm.

- combinatorial optimization problems like the traveling salesperson problem, the maximum independent set (MIS) problem (see Section 4.3.2) or the maximum cut problem (see Section 4.3.3) are NP-complete problems. It is conjectured that the NP-complete and the BQP class are disjoint, and therefore that there exist no polynomial quantum algorithm for these problems.
- the problem of finding the ground state energy of a k-local Hamiltonian (namely with Pauli terms, as in (29), acting on at most k qubits) is NP-hard [162] and QMA-complete [163], that is, informally, exponentially hard both for classical and quantum computers. One consequence of QMA-complete being conjectured to be disjoint from BQP is that there will *not* be a polynomial quantum algorithm to solve it.
- time-dynamics simulation, the original idea of Feynman [60], is an example where quantum computers are efficient (this problem is in BQP, see [76] and [164] for two polynomial implementations for k-local and sparse Hamiltonians, respectively) while classical computers are not.

Should one shy away from NP-complete or QMA-complete problems on the account that they will necessarily lead to exponential algorithms? A reasonable answer is no.

A first reason is that even if both classical computers and quantum computers scale exponentially for a given problem, (i) in concrete examples, the specific regime (symmetries, temperature, filling etc) of the problem at hand may make it easier to solve than the complexity theoretic prediction, and it could well be that a quantum algorithm outperforms a classical one in such regime (although it could also go in the other direction, namely the classical algorithm better exploits the specificity of the problem at hand), and (ii) even in a true exponential regime, the scaling of the exponent could be different and therefore lead to a speedup (the definition of speedup itself is worth discussing [44]: one could either define it as the ratio of the complexities, or express $C_{\text{classical}} = f(C_{\text{quantum}})$ and call the speedup polynomial or exponential depending on the scaling of the ratio or on the polynomial or exponential scaling of f. The first choice implies that two exponentially scaling algorithms could still scale exponentially with respect to each other; the second implies a polynomial scaling between two such algorithms).

A second more practical reason is that most application fields are not looking for the exact solution of a problem but for approximate solutions. The gain in *approximability* of a problem when switching from classical to quantum is all that matters in many cases. Indeed, in practice, most hard problems are solved only in an approximate fashion: obtaining an approximate solution in a reasonable time is considered to be a satisfactory goal for exponential problems. Strikingly, not all NP-complete problems are equal when it comes to their "approximability", namely the possibility to find approximate solutions in "reasonable" time. A classification in terms of approximability is therefore a more useful notion than the complexity classes we sketched above. There, the main quantity of interest is the "approximation ratio", defined as

$$\alpha(S) = \frac{C(S)}{C(S^\star)}, \tag{61}$$

where $C(S)$ is the cost of a solution (the quantity one wants to maximize, see Equation (24) for MaxCut or Equation (62) below for the unit-disk MIS, UDMIS) and $S^\star$ the optimal solution (which, for a NP-complete problem, can only be obtained in an exponential run time). The closest α is to 1, the better the solution. The approximability classification works in terms of how close to 1 the approximation ratio is:

(1) the easiest-to-approximate problems are those that, for any $\epsilon > 0$, have an approximation algorithm with $\alpha > 1-\epsilon$ that runs in time poly$(1/\epsilon, n)$. Such algorithms are called FPTAS

(for fully polynomial time approximation scheme). This is the case for the "backpack" problem, a NP-complete problem that consists in putting as many valuable objects in a backpack as possible but with a cap on the maximum weight. Belonging to FPTAS essentially means that the problem is a very easy problem in practice.

(2) then, there exists algorithms such that for any $\epsilon > 0$, the approximate solution with $\alpha > 1-\epsilon$ is obtained in time poly(n) (but this run time could scale as $\mathrm{e}^{1/\epsilon}!$). UDMIS is among those problems, called PTAS (for polynomial time approximation scheme).

(3) the class of problems called APX has algorithms that achieve $\alpha > f(n)$ in time 1/poly(n):

- (a) MaxCut and bounded-degree MIS (MIS on graphs with a bounded degree) are such that $f(n) = \text{const} = C$: there are algorithms that find an approximate solution with a guarantee that its approximation ratio will be at most C.
- (b) the general MIS problem has $f(n) = 1/\text{poly}(n)$ (where poly(n) denotes a polynomial in the size n) of the problem.

This classification suggests that the approximation ratio of a given algorithm (whether classical or quantum) be placed at the center of the stage when considering a new method. In particular, how α scales with system size is a key property to be monitored, with the hope that the scaling offered by quantum computers could be better than that of classical computers. Finally, it also helps identify those problems that a worth tackling with quantum computers: problems that are easy to approximate using a classical algorithm, like knapsack, are probably not a promising target. Conversely, problems that are hard to approximate by classical algorithms probably offer more space for quantum advantage (although nothing guarantees that there will be one).

4.3.2. *Rydberg processors: a spin-1/2 machine dedicated to unit-disk problems: the effect of decoherence*

The Hamiltonian (48) of Rydberg atoms turns out to be similar to the Hamiltonian translation of a well-known combinatorial optimization problem called the unit-disk maximum independent set (UDMIS) problem, as first pointed out by [165]. This problem consists in finding, given a graph $G = (V, E)$ (with vertices V and edges E), the largest set of vertices that do not share an edge (a set called the maximum independent set). This is illustrated in Figure 24(a). The "unit-disk" qualifier restricts the class of graphs to "unit-disk graphs", which correspond to planar graphs such that edges are drawn only between vertices that are closer than a certain distance (one by convention, hence the name "unit").

A candidate solution to the problem is characterized by a bitstring $S = (b_1, \ldots, b_n)$ (where n is the number of vertices), with $b_i = 1$ if the ith vertex belongs to the set and $b_i = 0$ otherwise. Maximizing the size of the set corresponds to maximizing $\sum_{i=1}^{n} b_i$, while excluding vertices that share an edge corresponds to imposing $b_i b_j = 0$ if $i, j \in E$. This constrained optimization problem can be formulated in an unconstrained way by using a Lagrange multiplier U to uphold the constraint, yielding the cost function

$$C(S) = \sum_{i=1}^{n} b_i - U \sum_{ij \in E} b_i b_j. \tag{62}$$

This classical cost function can be "quantized" by turning the binary variables b_i into operators $\hat{n}_i$. This yields the Hamiltonian

$$H = U \sum_{ij \in E} \hat{n}_i \hat{n}_j - \sum_{i=1}^{n} \hat{n}_i, \tag{63}$$

where we also added an overall minus sign to turn the problem into a minimization rather than a maximization problem. Hamiltonian (63) is very similar to (48), provided one approximates $C_6/|\mathbf{r}_i - \mathbf{r}_j|^6 > 0$ for $|\mathbf{r}_i - \mathbf{r}_j| < r_R$ and $= 0$ for larger distances. Thus, finding the solution to

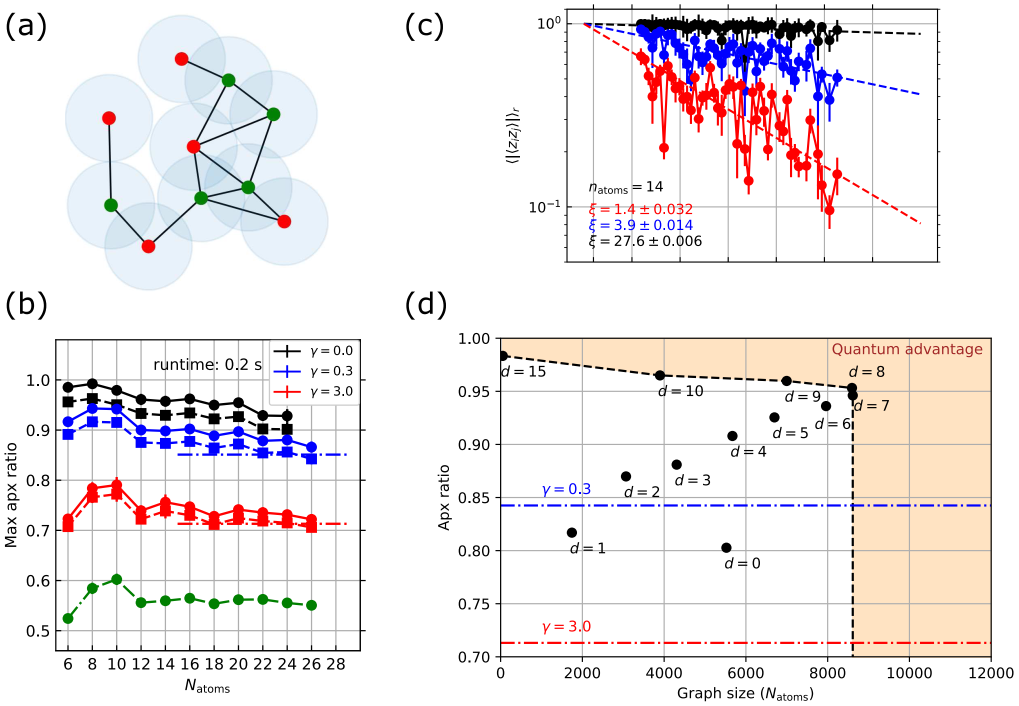

Figure 24. Solving the UDMIS problem with a Rydberg platform. (a) Sketch of the UDMIS problem: dots are vertices V while lines are edges E of the graph $G = (V, E)$ at hand. The red dots stand for the sought-after solution, the MIS. The circles are unit disks that determine the existence of edges. (b) Approximation ratio as a function of graph size for various dephasing noise levels γ (green curve: results obtained with uniform random sampling algorithm). (c) Absolute value of the spin–spin correlation as a function of distance (y-log scale) for the three noise levels of panel (b), and exponential fit. (d) Break-even diagram: black dots represent the approximation ratio and computational time achieved by a state-of-the-art classical heuristic algorithm to solve UDMIS. Reproduced from [156]. Copyright (2020, 2024) by the American Physical Society.

the UDMIS problem can be translated to finding the ground state of the Rydberg Hamiltonian (with $\Omega = 0$). This ground state is a priori not easy to find as the rest state of the processor is the $|00,\dots,0\rangle$ state (all atoms at rest). To reach the ground state with high probability, one can resort to a quantum annealing procedure (see Section 3.1.2), where one slowly turns the instantaneous Hamiltonian H_{Rydberg} from a Hamiltonian H_0 whose ground state is $|00,\dots,0\rangle$ to a final Hamiltonian as close as possible to H (Equation (63)). The adiabatic theorem guarantees that, provided this time evolution is long enough compared to the inverse instantaneous squared gap, and neglecting the difference between the final Rydberg Hamiltonian and the target H, the final state will be the ground state of H.

In [156], we analyzed the performance of a simple quantum annealing procedure in the presence of decoherence. We quantitatively verified the intuitive tradeoff between long annealing times, which increase the success probability of the annealing procedure, and short times, which limit the influence of noise. This competition leads to an optimal, noise-dependent annealing time. Choosing this time as our annealing time, we studied approximation ratio (Equation (61)) of our quantum algorithm, with $S^\star$ denoting the exact solution of the UD-MIS problem.

We investigated the dependence of the approximation ratio on the size of the graph for different noise strengths (with a Lindblad noise model similar to the one we presented in Section 4.2.2). As expected, more noise leads to a smaller approximation ratio (see Figure 24(b)). The approximation ratio slightly decreases with increasing size, with a plateau that comes at smaller and smaller sizes as the noise level grows. We use the value of this plateau to extrapolate the results to large graphs.

We compared these approximations ratios to those of a state-of-the-art classical approximation algorithm for UD-MIS. In this classical algorithm, the graph at hand is divided into subgraphs of small size d, whose MIS is determined exactly. These subsolutions are combined to build an approximate solution. The size d is tuned to reach a given approximation ratio (the larger d, the better quality) within a given time budget (the run time is exponential in d since UD-MIS is a NP-complete, hence exponential, problem). This leads to Figure 24(d), where the black dots are the approximation ratios and graph sizes that can be reached within a 2-second time budget. It allows to draw a “break-even” diagram, where the region of quantum advantage is drawn in orange. The quantum approximation ratios corresponding to two noise levels ($\gamma = 3$ corresponding to [75] and $\gamma = 0.3$ close to 2024 levels of noise) are represented by the red and blue line, assuming the decoherence levels are kept constant with the number of atoms. We predict that a number of more than 8000 atoms will be needed to beat the classical algorithm (to reach the right-hand “advantage area”), or much better noise levels (to reach the top “advantage area”). Increasing the number of atoms is an undergoing effort in experimental labs and companies. One major element to factor in, though, is the repetition rate of the experiment. In Rydberg platforms, the repetition rate about a Herz [62], with possible improvements up to a few tens of Herz. This is many orders of magnitude away from superconducting processors (MHz), not to mention classical processors (GHz).

Improving the quality is another key aspect. Beyond the final approximation ratio estimates that we were able to extract from our simulations, our simulations also indicate one of the causes of the degradation in solution quality. Indeed, a maximum independent set can be regarded as the generalization to any graph of a (classical) antiferromagnetic (AF) state: in a 1D chain a MIS is nothing but an antiferromagnetic spin chain 0101...10. Errors will cause disruptions in the perfect AF order by flipping some of the spins, resulting in AF domains separated by boundaries corresponding to defects. The size of the corresponding IS will be that of the maximum IS (the AF state) minus the number of boundary sites. To count the number of defects, the relevant quantity to look at is the (AF) correlation length ξ of the chain: it gives the typical size of the domains. For a fixed ξ, the number of boundary sites be approximately n/ξ. Thus, the approximation ratio in 1D will be

$$\alpha = \frac{|\mathrm{MIS}| - n/\xi}{|\mathrm{MIS}|} = \frac{\kappa n - n/\xi}{\kappa n} = 1 - \frac{1}{\kappa\xi}, \tag{64}$$

where we used the fact that the size of the MIS is proportional to the graph size. The same reasoning can be extended to two dimensions: there, the number of domains is $\sim n/\xi^2$ and the size of their boundary is $\sim\xi$, so that the size of the IS will be $|\mathrm{MIS}| - n/\xi^2 \times \xi$. Equation (64) is consistent with the numerical data: one can extract the correlation length from the noisy data (Figure 24(c)). One does see that noise shortens ξ, leading to a lower approximation ratio. The relationship between α and ξ we briefly derived is confirmed by our data. In other words, this classical optimization example emphasizes the importance for quantum processors to be able to generate states with very long correlations lengths. Classical heuristics, in a way, also come with a finite correlation length or short-sightedness in that they solve subproblems with a given size d exactly (at an exponential price) and they then patch the results together. The challenge of quantum computers is to “solve” subproblems that are larger, and/or to solve them faster, than classical algorithms.

There are other promising avenues besides our work. One is to find better algorithms than quantum annealing to find the ground state of the problem at hand. Variational approaches, in the same vein as the variational quantum eigensolver, have been explored. The quantum approximate optimization algorithm (QAOA, [124]) is, like the Hamiltonian variational ansatz we encountered previously, directly inspired by quantum annealing, and suffers from the same issues as VQE.

One issue with the UDMIS problem is that it is, in the approximability hierarchy, a quite easy problem. It thus looks promising to try and tackle the MIS problem on more general graphs than unit-disk graphs. Ways to achieve this for general 2D graphs with Rydberg atoms have been laid out in [166] (with the help of additional atoms adding up to $O(n^2)$ atoms to handle a graph of size n) and in [167] (by using 3D arrangements of atoms).

4.3.3. *Q-score: a benchmark for quantum computers*

The above study of the UDMIS problem introduced the approximation ratio α as a quantitative measure of the solution quality of a given (quantum as well as classical) algorithm. This measure of success is also, as we explained in Section 4.3.1, a formal measure of the *approximability* of hard computational problems.

Starting from the importance of this metric, we defined, in [168], a benchmark for quantum computing architectures based on the assessment of the approximation ratio reachable by quantum processors when solving another combinatorial optimization problem, namely the maximum cut (MaxCut) problem. As introduced in Section 2.3, this problem consists in finding the bipartition of vertices in a graph that maximizes the number of edges between the vertices of each partition, and finds applications in many fields.

The design of this benchmark obeyed three criteria: to be (i) application-oriented, as opposed to low-level benchmarks like randomized benchmarking [169, 170], (ii) scalable, namely efficiently computable for large problem instances, as opposed to benchmarks like the quantum volume or cross-entropy benchmarking, which require the exponential computation of circuit outcome probabilities, and (iii) hardware-agnostic, namely not restricted to, let alone favoring any technology, whether gate-based or analog.

While points (i) and (iii) are easily met by the choice of the MaxCut problem, (ii) poses an important challenge: to compute the approximation ratio α, one in principle needs to compute the exact solution $S^\star$ to compute the denominator of Equation (61), thus incurring an exponential run time. This issue is circumvented in our proposal by computing the ratio of the *averages* over graph classes. We chose the Erdos–Renyi class to do this average because of the possibility to reach hard graph instances by tuning the edge probability p of the Erdos–Renyi graphs. The advantage of performing averages is that asymptotic scalings of average optimal costs $\langle C(S^\star)\rangle_{\mathscr{G}}$ can be computed efficiently, as opposed to costs of individual solutions $S^\star$.

Generically, the so-obtained approximation ratio (in fact, a slight modification thereof that we called β instead of α to subtract a trivial contribution) decreases with graph size due to the increasing effect of decoherence. We call “Q-score” the graph size above which β falls below an arbitrary threshold of 20%.

Figure 25 illustrates a Q-score computation by using a variational algorithm, QAOA, to solve the MaxCut problem: while noiseless simulations show more or less constant approximation ratios as the graph size (hence number of qubits) grows (with improved ratios as the number p of layers, and hence ansatz expressivity, increase), noise leads to a gradual degradation of the ratio with size, even more so when the connectivity is limited, which leads to deeper, and hence more error-prone, circuits. Thus, the Q-score reflects both the quality of hardware and that of the software (a better compilation or a better algorithm can lead to higher Q-scores for the same hardware).

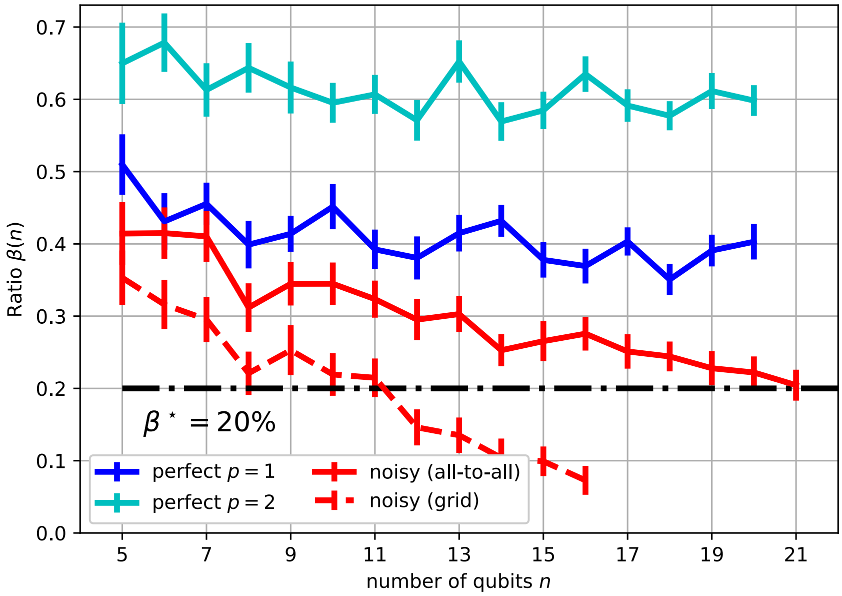

Figure 25. Simulation of the Q-score: modified approximation ratio β as a function of the number of qubits (graph size) for a QAOA algorithm with $p = 1$ and $p = 2$ layers, for a perfect (noiseless, blue and cyan lines) and a noisy (red lines) quantum computer (with two qubit topologies in the latter case: all-to-all (solid) and nearest-neighbor on a grid (dashed)). The noise levels (2% and 0.4% average depolarizing error rates for two and one-qubit gates, respectively) are representative of NISQ devices in 2021. Reproduced from [168]. Copyright (2021) by IEEE.

Since the publication of the Q-score proposal, the Q-score was computed for three architectures: (i) a d-wave computer architecture, with reported an experimental Q-score of 140 (and even 12,500 for a hybrid quantum-classical approach, [171]), (ii) a Rydberg processor, with numerical simulations predicting Q-scores above 18 [172], and (iii) a transmon quantum computer by the IQM company, with a reported experimental Q-score of 11 (unpublished, https://www.meetiqm.com/newsroom/press-releases/iqm-achieves-20-qubit-benchmark-result), with error mitigation techniques.

5. Conclusion

The field of quantum computing for many-body physics stands at a crossroads. By now, many proof-of-concept computations have shown that one could indeed run many-body computations on the emerging hardware platforms, but also that these demonstrations are still largely outperformed by classical algorithms. This state of affairs is both due to hardware imperfections and to more fundamental limitations of variational quantum algorithms, which have been the main fuel of the NISQ era so far. Resource estimates and a better knowledge of the generic cost landscape unambiguously indicate that sophisticated approaches will be needed to avoid the measurement variance problem and the barren plateau problem. Whether these hurdles will prove insurmountable or not is an open question. In any case, the design of resource-efficient algorithms to generate complex many-body states will be a central ingredient.

Given these difficulties on the one hand, and the recent demonstrations of small-scale error correction on the other hand, one may be tempted to directly aim for fault-tolerant quantum algorithms. It is at the same time an optimistic move, and a pessimistic one. It is optimistic because there are strings attached to this program: first of all, the scale of the experimental effort required to obtain fault-tolerant processors is such that this technology will probably not be available before long years [130]. Second, even perfectly corrected quantum computers come with open challenges, like the orthogonality catastrophe we discussed in Section 22.

It is also a pessimistic move, because the scale and quality of current processors, with hundreds of particles with large spatial correlations (7 lattice spacings in 2021, [64]), makes it very likely that hitherto unreachable physical regimes are already accessible with current or near-term hardware. Indeed, although the most advanced classical methods can still rival with quantum advantage contentions, as we showed in Section 4.1.1, they also come with hard limitations, especially when it comes to dynamics. Of course, the precise way to squeeze acceleration out of these NISQ processors is still elusive. Our view is that it will probably result from a subtle interplay between classical and quantum algorithms. The hybrid algorithms we presented, that exchange classical and quantum information to optimize the qubit orbital basis (Section 4.2.1), to compress state preparation circuits (Section 4.1.2), or to sidestep the issues related to fermionic antisymmetry (Section 4.2.2), are first steps in this direction.

Very recently, much work has been devoted to going beyond the limitations of simple VQEs without directly turning to fault-tolerant algorithms like quantum phase estimation. For instance, [173] introduces a quantum version of the projective variant of coupled cluster equations (see Section 2.1.6), with promising convergence properties compared to VQE. Reference [174] do not carry any optimization of the variational energy, but instead (i) start from a classical CCSD computation to build their quantum circuit and (ii) do not attempt to estimate energies but sample bitstrings from the obtained state, to then construct restrictions of the many-body Hamiltonian to the subspace of the most probable bistrings, before classically diagonalizing this Hamiltonian using exact diagonalization techniques (similar to those described in Section 2.1.2). Another promising strand of methods [22] uses a quantum processing step within an otherwise classical quantum Monte Carlo code like AFQMC [175] or FCI-QMC [176]. There again, a careful investigation of what advantage the quantum processor brings will be crucial [21].

Declaration of interests

The authors do not work for, advise, own shares in, or receive funds from any organization that could benefit from this article, and have declared no affiliations other than their research organizations.

Acknowledgments

This manuscript was submitted for my Habilitation à Diriger des Recherches (HDR). I would like to thank the reviewers Grégoire Misguich, Laurent Sanchez-Palencia and Eric Cancès, as well as the other jury members Yuri Alexeev, Antoine Browaeys, Denis Lacroix and Daniel Estève, for a careful reading of the manuscript.

References

[1] J. Hubbard, "Electron correlations in narrow energy bands", *Proc. R. Soc. A: Math. Phys. Eng. Sci.* **276** (1963), no. 1365, pp. 238–257.

[2] F. Aryasetiawan, M. Imada, A. Georges, G. Kotliar, S. Biermann and A. Lichtenstein, "Frequency-dependent local interactions and low-energy effective models from electronic structure calculations", *Phys. Rev. B* **70** (2004), no. 19, article no. 195104.

[3] S. Jiang, D. J. Scalapino and S. R. White, "Density matrix renormalization group based downfolding of the three-band Hubbard model: Importance of density-assisted hopping", *Phys. Rev. B* **108** (2023), no. 16, article no. L161111.

[4] J. G. Bednorz and K. A. Muller, "Possible high Tc superconductivity in the Ba–La–Cu–O system", *Z. Phys. B: Condens. Matter* **64** (1986), pp. 189–193.

[5] N. F. Mott and R. Peierls, "Discussion of the paper by de Boer and Verwey", *Proc. Phys. Soc.* **49** (1937), no. 4S, pp. 72–73.

[6] W. Kohn, "Nobel lecture: electronic structure of matter-wave functions and density functionals", *Rev. Mod. Phys.* **71** (1999), no. 5, pp. 1253–1266.

[7] W. Kohn and L. J. Sham, "Self-consistent equations including exchange and correlation effects", *Phys. Rev.* **385** (1965), no. 1951, pp. 1133–1138.

[8] F. A. Evangelista, "Perspective: multireference coupled cluster theories of dynamical electron correlation", *J. Chem. Phys.* **149** (2018), no. 3, article no. 030901.

[9] E. Koch, "The Lanczos method", in *The LDA+DMFT Approach to Strongly Correlated Materials* (E. Pavarini, E. Koch, D. Vollhardt and A. Lichtenstein, eds.), Verlag des Forschungszentrum Jülich, Jülich, 2011. Online at https://www.cond-mat.de/events/correl11/manuscripts/koch.pdf.

[10] A. Wietek and A. M. Läuchli, "Sublattice coding algorithm and distributed memory parallelization for large-scale exact diagonalizations of quantum many-body systems", *Phys. Rev. E* **98** (2018), no. 3, article no. 033309.

[11] W. Krauth, "Introduction to Monte Carlo algorithms", in *Advances in Computer Simulation* (J. Kertesz and I. Kondor, eds.), Lecture Notes in Physics, Springer Verlag, 1998.

[12] M. Troyer and U. J. Wiese, "Computational complexity and fundamental limitations to fermionic quantum Monte Carlo simulations", *Phys. Rev. Lett.* **94** (2005), no. 17, pp. 1–4.

[13] N. Prokof'ev, "Diagrammatic Monte Carlo", in *Many-Body Methods for Real Materials Modeling and Simulation* (E. Pavarini, E. Koch and S. Zhang, eds.), Forschungszentrum Jülich, Jülich, 2019.

[14] R. Rossi, "Determinant diagrammatic Monte Carlo algorithm in the thermodynamic limit", *Phys. Rev. Lett.* **119** (2017), no. 4, article no. 045701.

[15] R. Rossi, N. Prokof'ev, B. Svistunov, K. Van Houcke and F. Werner, "Polynomial complexity despite the fermionic sign", *Europhys. Lett.* **118** (2017), no. 1, article no. 10004.

[16] E. Pavarini, E. Koch and U. Schollwöck, *Emergent Phenomena in Correlated Matter*, Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag Jülich, Jülich, 2013, p. 562. Online at http://hdl.handle.net/2128/5389.

[17] R. C. Grimm and R. G. Storer, "Monte-Carlo solution of Schrödinger's equation", *J. Comput. Phys.* **7** (1971), no. 1, pp. 134–156.

[18] S. Zhang, J. Carlson and J. E. Gubernatis, "Constrained path Monte Carlo method for fermion ground states", *Phys. Rev. B* **55** (1997), no. 12, pp. 7464–7477.

[19] G. H. Booth, A. J. W. Thom and A. Alavi, "Fermion Monte Carlo without fixed nodes: a game of life, death, and annihilation in Slater determinant space", *J. Chem. Phys.* **131** (2009), no. 5, article no. 054106.

[20] G. Carleo and M. Troyer, "Solving the quantum many-body problem with artificial neural networks", *Science* **355** (2017), no. 6325, pp. 602–606.

[21] G. Mazzola, "Quantum computing for chemistry and physics applications from a Monte Carlo perspective", *J. Chem. Phys.* **160** (2024), article no. 010901.

[22] T. Jiang, J. Zhang, M. K. A. Baumgarten, et al., *Walking through Hilbert space with quantum computers*, preprint, 2024, 2407.11672.

[23] U. Schollwöck, "The density-matrix renormalization group in the age of matrix product states", *Ann. Phys.* **326** (2011), no. 1, pp. 96–192.

[24] M. B. Hastings, "An area law for one-dimensional quantum systems", *J. Stat. Mech.: Theory Exp.* **2007** (2007), no. 08, P08024–P08024.

[25] F. Verstraete and J. I. Cirac, *Renormalization algorithms for Quantum-Many Body Systems in two and higher dimensions*, preprint, 2004, cond-mat/0407066. p. 1–5.

[26] I. L. Markov and Y. Shi, "Simulating quantum computation by contracting tensor networks", *SIAM J. Comput.* **38** (2008), no. 3, pp. 963–981.

[27] M. P. Zaletel and F. Pollmann, "Isometric tensor network states in two dimensions", *Phys. Rev. Lett.* **124** (2020), no. 3, article no. 037201.

[28] R. Alkabetz and I. Arad, "Tensor networks contraction and the belief propagation algorithm", *Phys. Rev. Res.* **3** (2021), no. 2, article no. 023073.

[29] S. Sahu and B. Swingle, *Efficient tensor network simulation of quantum many-body physics on sparse graphs*, preprint, 2022, 2206.04701.

[30] P. Calabrese and J. Cardy, "Evolution of entanglement entropy in one-dimensional systems", *J. Stat. Mech.: Theory Exp.* **2005** (2005), no. 04, P04010.

[31] J. P. F. LeBlanc, A. E. Antipov, F. Becca, et al., "Solutions of the two-dimensional Hubbard model: benchmarks and results from a wide range of numerical algorithms", *Phys. Rev. X* **5** (2015), no. 4, article no. 041041.

[32] H. Xu, C.-M. Chung, M. Qin, U. Schollwöck, S. R. White and S. Zhang, *Coexistence of superconductivity with partially filled stripes in the Hubbard model*, preprint, 2023, 2303.08376.

[33] P. O. Löwdin, "Quantum theory of many-particle systems. I. Physical interpretations by means of density matrices, natural spin-orbitals, and convergence problems in the method of configurational interaction", *Phys. Rev.* **97** (1955), no. 6, pp. 1474–1489.

[34] F. Coester and H. Kümmel, "Short-range correlations in nuclear wave functions", *Nucl. Phys.* **17** (1960), no. C, pp. 477–485.

[35] J. B. Robinson and P. J. Knowles, "Approximate variational coupled cluster theory", *J. Chem. Phys.* **135** (2011), no. 4, article no. 044113.

[36] G. Harsha, T. Shiozaki and G. E. Scuseria, "On the difference between variational and unitary coupled cluster theories", *J. Chem. Phys.* **148** (2018), no. 4, article no. 044107.

[37] W. Kutzelnigg, "Pair correlation theories", in *Methods of Electronic Structure Theory* (H. F. Schaefer, ed.), Springer US, 1977, pp. 129–182.

[38] W. Kutzelnigg, "Error analysis and improvements of coupled-cluster theory", *Theoret. Chim. Acta* **80** (1991), no. 4–5, pp. 349–386.

[39] P. G. Szalay, M. Nooijen and R. J. Bartlett, "Alternative ansätze in single reference coupled-cluster theory. III. A critical analysis of different methods", *J. Chem. Phys.* **103** (1995), no. 1, pp. 281–298.

[40] R. F. Bishop, "The coupled cluster method", in *Microscopic Quantum Many-Body Theories and Their Applications*, Springer, Berlin, Heidelberg, 2008, pp. 1–70.

[41] B. O. Roos, P. R. Taylor and P. E. M. Sigbahn, "A complete active space SCF method (CASSCF) using a density matrix formulated super-CI approach", *Chem. Phys.* **48** (1980), no. 2, pp. 157–173.

[42] B. O. Roos, "The complete active space self-consistent field method and its applications in electronic structure calculations", *Adv. Chem. Phys.* **69** (1987), pp. 399–445.

[43] B. Huron, J. P. Malrieu and P. Rancurel, "Iterative perturbation calculations of ground and excited state energies from multiconfigurational zeroth-order wavefunctions", *J. Chem. Phys.* **58** (1973), no. 12, pp. 5745–5759.

[44] G. K.-L. Chan, "Quantum chemistry, classical heuristics, and quantum advantage", *Faraday Discuss.* **254** (2024), pp. 11–52.

[45] A. Georges, G. Kotliar, W. Krauth and M. J. Rozenberg, "Dynamical mean-field theory of strongly correlated fermion systems and the limit of infinite dimensions", *Rev. Mod. Phys.* **68** (1996), no. 1, pp. 13–125.

[46] T. Ayral, P. Besserve, D. Lacroix and E. A. Ruiz Guzman, "Quantum computing with and for many-body physics", *Eur. Phys. J. A* **59** (2023), no. 10, article no. 227.

[47] A. Georges, "Strongly correlated electron materials: dynamical mean-field theory and electronic structure", *AIP Conf. Proc.* **715** (2004), no. 1, pp. 3–74.

[48] E. Gull, A. J. Millis, A. I. Lichtenstein, A. N. Rubtsov, M. Troyer and P. Werner, "Continuous-time Monte Carlo methods for quantum impurity models", *Rev. Mod. Phys.* **83** (2011), no. 2, pp. 349–404.

[49] H. Aoki, N. Tsuji, M. Eckstein, M. Kollar, T. Oka and P. Werner, "Nonequilibrium dynamical mean-field theory and its applications", *Rev. Mod. Phys.* **86** (2014), no. 2, pp. 779–837.

[50] F. Lechermann, A. Georges, G. Kotliar and O. Parcollet, "Rotationally invariant slave-boson formalism and momentum dependence of the quasiparticle weight", *Phys. Rev. B* **76** (2007), no. 15, article no. 155102.

[51] J. Bünemann and F. Gebhard, "Equivalence of Gutzwiller and slave-boson mean-field theories for multiband Hubbard models", *Phys. Rev. B* **76** (2007), no. 19, article no. 193104.

[52] N. Lanatà, Y. Yao, C.-Z. Wang, K.-M. Ho and G. Kotliar, "Phase diagram and electronic structure of praseodymium and plutonium", *Phys. Rev. X* **5** (2015), no. 1, article no. 011008.

[53] T. Ayral, T.-H. Lee and G. Kotliar, "Dynamical mean-field theory, density-matrix embedding theory, and rotationally invariant slave bosons: A unified perspective", *Phys. Rev. B* **96** (2017), no. 23, article no. 235139.

[54] M. Ferrero, P. Cornaglia, L. De Leo, O. Parcollet, G. Kotliar and A. Georges, "Pseudogap opening and formation of Fermi arcs as an orbital-selective Mott transition in momentum space", *Phys. Rev. B* **80** (2009), no. 6, article no. 064501.

[55] G. Knizia and G. K.-L. Chan, "Density matrix embedding: a simple alternative to dynamical mean-field theory", *Phys. Rev. Lett.* **109** (2012), no. 18, article no. 186404.

[56] T. Ayral, P. Besserve, D. Lacroix and E. A. R. Guzman, *Quantum Computing with and for Many-Body Physics*, Springer, Berlin, Heidelberg, 2023, pp. 1–46. Online at http://arxiv.org/abs/2303.04850.

[57] S. Bravyi, "Monte Carlo simulation of stoquastic Hamiltonians", *Quantum Inf. Comput.* **15** (2015), no. 13–14, pp. 1122–1140.

[58] B. Heim, T. F. Rønnow, S. V. Isakov and M. Troyer, "Quantum versus classical annealing of Ising spin glasses", *Science* **348** (2015), no. 6231, pp. 215–217.

[59] V. S. Denchev, S. Boixo, S. V. Isakov, N. Ding, R. Babbush, V. Smelyanskiy, J. Martinis and H. Neven, “What is the computational value of finite-range tunneling?”, *Phys. Rev. X* **6** (2016), article no. 031015.

[60] R. P. Feynman, “Simulating physics with computers”, *Int. J. Theor. Phys.* **21** (1982), no. 6–7, pp. 467–488.

[61] I. Bloch, J. Dalibard and W. Zwerger, “Many-body physics with ultracold gases”, *Rev. Mod. Phys.* **80** (2008), no. September, article no. 885.

[62] A. Browaeys and T. Lahaye, “Many-body physics with individually controlled Rydberg atoms”, *Nat. Phys.* **16** (2020), no. 2, pp. 132–142.

[63] R. Jördens, N. Strohmaier, K. Günter, M. Henning and T. Esslinger, “A Mott insulator of fermionic atoms in an optical lattice”, *Nature* **455** (2008), pp. 204–207.

[64] P. Scholl, M. Schuler, H. J. Williams, et al., “Quantum simulation of 2D antiferromagnets with hundreds of Rydberg atoms”, *Nature* **595** (2021), no. 7866, pp. 233–238.

[65] M. Xu, L. H. Kendrick, A. Kale, Y. Gang, G. Ji, R. T. Scalettar, M. Lebrat and M. Greiner, “Frustration- and doping-induced magnetism in a Fermi–Hubbard simulator”, *Nature* **620** (2023), no. 7976, pp. 971–976.

[66] P. W. Shor, “Algorithms for quantum computation: discrete logarithms and factoring”, in *Proceedings 35th Annual Symposium on Foundations of Computer Science*, IEEE Computer Society Press, 1995, pp. 124–134. Online at http://ieeexplore.ieee.org/document/365700.

[67] L. K. Grover, “A fast quantum mechanical algorithm for database search”, in *Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing - STOC ’96*, ACM Press, 1996, pp. 212–219.

[68] P. W. Shor, “Scheme for reducing decoherence in quantum computer memory”, *Phys. Rev. A* **52** (1995), no. 4, pp. 2493–2496.

[69] P. W. Shor, “Fault-tolerant quantum computation”, in *Proceedings of 37th Conference on Foundations of Computer Science*, IEEE Computer Society Press, 1996, pp. 56–65.

[70] Z. Chen, K. J. Satzinger, J. Atalaya, et al., “Exponential suppression of bit or phase errors with cyclic error correction”, *Nature* **595** (2021), no. 7867, pp. 383–387.

[71] S. Krinner, N. Lacroix, A. Remm, et al., “Realizing repeated quantum error correction in a distance-three surface code”, *Nature* **605** (2022), no. 7911, pp. 669–674.

[72] D. Bluvstein, S. J. Evered, A. A. Geim, et al., “Logical quantum processor based on reconfigurable atom arrays”, *Nature* **626** (2024), no. 7997, pp. 58–65.

[73] M. P. da Silva, C. Ryan-Anderson, J. M. Bello-Rivas, et al., *Demonstration of logical qubits and repeated error correction with better-than-physical error rates*, preprint, 2024, 2404.02280. p. 1–13.

[74] T. Albash and D. A. Lidar, “Adiabatic quantum computation”, *Rev. Mod. Phys.* **90** (2018), no. 1, article no. 015002.

[75] V. Lienhard, S. de Léséleuc, D. Barredo, T. Lahaye, A. Browaeys, M. Schuler, L.-P. Henry and A. M. Läuchli, “Observing the space- and time-dependent growth of correlations in dynamically tuned synthetic ising models with antiferromagnetic interactions”, *Phys. Rev. X* **8** (2018), no. 2, article no. 021070.

[76] S. Lloyd, “Universal quantum simulators”, *Science* **273** (1996), no. 5278, pp. 1073–1078.

[77] A. M. Childs, Y. Su, M. C. Tran, N. Wiebe and S. Zhu, “Theory of trotter error with commutator scaling”, *Phys. Rev. X* **11** (2021), no. 1, article no. 011020.

[78] L. Lin and Y. Tong, “Heisenberg-limited ground state energy estimation for early fault-tolerant quantum computers”, *Phys. Rev. X Quantum* **3** (2022), article no. 010318.

[79] A. M. Childs and N. Wiebe, “Hamiltonian simulation using linear combinations of unitary operations”, *Quantum Inform. Comput.* **12** (2012), pp. 901–924.

[80] G. H. Low and I. L. Chuang, “Optimal Hamiltonian simulation by quantum signal processing”, *Phys. Rev. Lett.* **118** (2017), no. 1, article no. 010501.

[81] G. H. Low and I. L. Chuang, “Hamiltonian simulation by qubitization”, *Quantum* **3** (2019), p. 163.

[82] J. Haah, M. B. Hastings, R. Kothari and G. H. Low, “Quantum algorithm for simulating real time evolution of lattice Hamiltonians”, *SIAM J. Comput.* **52** (2023), no. 6, pp. 250–284.

[83] A. Y. Kitaev, *Quantum measurements and the Abelian Stabilizer Problem*, preprint, 1995, quant-ph/9511026. p. 1–22.

[84] R. Cleve, A. Ekert, C. Macchiavello and M. Mosca, “Quantum algorithms revisited”, *Proc. R. Soc. A: Math. Phys. Eng. Sci.* **454** (1998), no. 1969, pp. 339–354.

[85] A. Aspuru-Guzik, A. D. Dutoi, P. J. Love and M. Head-Gordon, “Chemistry: simulated quantum computation of molecular energies”, *Science* **309** (2005), no. 5741, pp. 1704–1707.

[86] N. M. Tubman, C. Mejuto-Zaera, J. M. Epstein, et al., *Postponing the orthogonality catastrophe: efficient state preparation for electronic structure simulations on quantum devices*, preprint, 2018, 1809.05523. p. 1–13.

[87] S. Lee, J. Lee, H. Zhai, et al., “Evaluating the evidence for exponential quantum advantage in ground-state quantum chemistry”, *Nat. Commun.* **14** (2023), no. 1, article no. 1952.

[88] B. Bauer, D. Wecker, A. J. Millis, M. B. Hastings and M. Troyer, “Hybrid quantum-classical approach to correlated materials”, *Phys. Rev. X* **6** (2016), no. 3, article no. 031045.

[89] J. M. Kreula, S. R. Clark and D. Jaksch, “Non-linear quantum-classical scheme to simulate non-equilibrium strongly correlated fermionic many-body dynamics”, *Sci. Rep.* **6** (2016), no. 1, article no. 32940.

[90] J. M. Kreula, L. García-Álvarez, L. Lamata, S. R. Clark, E. Solano and D. Jaksch, “Few-qubit quantum-classical simulation of strongly correlated lattice fermions”, *EPJ Quantum Technol.* **3** (2016), no. 1, article no. 11.

[91] I. D. Kivlichan, C. Gidney, D. W. Berry, et al., “Improved fault-tolerant quantum simulation of condensed-phase correlated electrons via trotterization”, *Quantum* **4** (2020), article no. 296.

[92] A. G. Fowler, M. Mariantoni, J. M. Martinis and A. N. Cleland, “Surface codes: towards practical large-scale quantum computation”, *Phys. Rev. A* **86** (2012), no. 3, article no. 032324.

[93] A. Morvan, B. Villalonga, X. Mi, et al., “Phase transition in random circuit sampling”, *Nature* **634** (2024), pp. 328–333.

[94] F. Arute, K. Arya, R. Babbush, et al., “Quantum supremacy using a programmable superconducting processor”, *Nature* **574** (2019), no. 7779, pp. 505–510.

[95] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P. J. Love, A. Aspuru-Guzik and J. L. O’Brien, “A variational eigenvalue solver on a quantum processor”, *Nat. Commun.* **5** (2013), no. 1, article no. 4213.

[96] J. Tilly, H. Chen, S. Cao, et al., “The variational quantum eigensolver: a review of methods and best practices”, *Phys. Rep.* **986** (2022), pp. 1–128.

[97] C. Kokail, C. Maier, R. van Bijnen, et al., “Self-verifying variational quantum simulation of lattice models”, *Nature* **569** (2019), no. 7756, pp. 355–360.

[98] J. R. McClean, S. Boixo, V. N. Smelyanskiy, R. Babbush and H. Neven, “Barren plateaus in quantum neural network training landscapes”, *Nat. Commun.* **9** (2018), article no. 4812.

[99] M. Larocca, S. Thanasilp, S. Wang, et al., *A review of barren plateaus in variational quantum computing*, preprint, 2024, 2405.00781. p. 1–21.

[100] R. Mao, G. Tian and X. Sun, *Barren plateaus of alternated disentangled UCC ansatzs*, preprint, 2023, 2312.08105. p. 18–21.

[101] M. Ragone, B. N. Bakalov, F. Sauvage, A. F. Kemper, C. O. Marrero, M. Larocca and M. Cerezo, “A unified theory of barren plateaus for deep parametrized quantum circuits”, *Nat. Commun.* **15** (2024), article no. 7172.

[102] A. Arrasmith, Z. Holmes, M. Cerezo and P. J. Coles, “Equivalence of quantum barren plateaus to cost concentration and narrow gorges”, *Quantum Sci. Technol.* **7** (2022), no. 4, article no. 045015.

[103] H. R. Grimsley, S. E. Economou, E. Barnes and N. J. Mayhall, “An adaptive variational algorithm for exact molecular simulations on a quantum computer”, *Nat. Commun.* **10** (2019), no. 1, article no. 3007.

[104] H. L. Tang, V. O. Shkolnikov, G. S. Barron, H. R. Grimsley, N. J. Mayhall, E. Barnes and S. E. Economou, “qubit-ADAPT-VQE: an adaptive algorithm for constructing hardware-efficient ansatze on a quantum processor”, *Phys. Rev. X Quantum* **2** (2021), article no. 020310.

[105] A. Kandala, A. Mezzacapo, K. Temme, M. Takita, M. Brink, J. M. Chow and J. M. Gambetta, “Hardware-efficient variational quantum eigensolver for small molecules and quantum magnets”, *Nature* **549** (2017), no. 7671, pp. 242–246.

[106] D. Wecker, M. B. Hastings and M. Troyer, “Progress towards practical quantum variational algorithms”, *Phys. Rev. A* **92** (2015), no. 4, article no. 042303.

[107] P. J. J. O’Malley, R. Babbush, I. D. Kivlichan, et al., “Scalable quantum simulation of molecular energies”, *Phys. Rev. X* **6** (2015), no. 3, article no. 031007.

[108] N. C. Rubin, R. Babbush and J. McClean, “Application of fermionic marginal constraints to hybrid quantum algorithms”, *J. Phys.* **20** (2018), no. 5, article no. 053020.

[109] A. Arrasmith, L. Cincio, R. D. Somma and P. J. Coles, *Operator sampling for shot-frugal optimization in variational algorithms*, preprint, 2020, 2004.06252. p. 1–11.

[110] M. Potthoff, “Two-site dynamical mean-field theory”, *Phys. Rev. B* **64** (2001), no. 16, article no. 165114.

[111] I. Rungger, N. Fitzpatrick, H. Chen, et al., *Dynamical mean field theory algorithm and experiment on quantum computers*, preprint, 2019, 1910.04735. p. 1-10.

[112] B. Jaderberg, A. Agarwal, K. Leonhardt, M. Kiffner and D. Jaksch, “Minimum hardware requirements for hybrid quantum–classical DMFT”, *Quantum Sci. Technol.* **5** (2020), no. 3, article no. 034015.

[113] Y. Yao, F. Zhang, C.-Z. Wang, K.-M. Ho and P. P. Orth, “Gutzwiller hybrid quantum-classical computing approach for correlated materials”, *Phys. Rev. Res.* **3** (2021), no. 1, article no. 013184.

[114] J. Selisko, M. Amsler, C. Wever, et al., *Dynamical mean field theory for real materials on a quantum computer*, preprint, 2024, 2404.09527. p. 1–25.

[115] S. Boixo, S. V. Isakov, V. N. Smelyanskiy, et al., “Characterizing quantum supremacy in near-term devices”, *Nat. Phys.* **14** (2016), no. 6, pp. 595–600.

[116] B. Rudiak-Gould, *The sum-over-histories formulation of quantum computing*, preprint, 2006, quant-ph/0607151.

[117] F. Pan and P. Zhang, *Simulating the Sycamore quantum supremacy circuits*, preprint, 2021, 2103.03074. p. 1–9.

[118] F. Pan, K. Chen and P. Zhang, "Solving the sampling problem of the Sycamore quantum supremacy circuits", *Phys. Rev. Lett.* **129** (2022), article no. 090502.

[119] J. Gray and S. Kourtis, "Hyper-optimized tensor network contraction", *Quantum* **5** (2021), pp. 1–22.

[120] C. Huang, F. Zhang, M. Newman, et al., "Efficient parallelization of tensor network contraction for simulating quantum computation", *Nat. Comput. Sci.* **1** (2021), no. 9, pp. 578–587.

[121] Y. Zhou, E. M. Stoudenmire and X. Waintal, "What limits the simulation of quantum computers?", *Phys. Rev. X* **10** (2020), no. 4, article no. 041038.

[122] G. Vidal, "Efficient classical simulation of slightly entangled quantum computations", *Phys. Rev. Lett.* **91** (2003), no. 14, article no. 147902.

[123] T. Ayral, T. Louvet, Y. Zhou, C. Lambert, E. M. Stoudenmire and X. Waintal, "Density-matrix renormalization group algorithm for simulating quantum circuits with a finite fidelity", *PRX Quantum* **4** (2023), no. 2, article no. 020304.

[124] E. Farhi, J. Goldstone and S. Gutmann, *A quantum approximate optimization algorithm*, preprint, 2014, 1411.4028.

[125] K. Noh, L. Jiang and B. Fefferman, "Efficient classical simulation of noisy random quantum circuits in one dimension", *Quantum* **4** (2020), p. 318.

[126] A. Müller, T. Ayral and C. Bertrand, *Enabling large-depth simulation of noisy quantum circuits with positive tensor networks*, preprint, 2024, 2403.00152. p. 1–17.

[127] S. Cheng, C. Cao, C. Zhang, Y. Liu, S.-Y. Hou, P. Xu and B. Zeng, "Simulating noisy quantum circuits with matrix product density operators", *Phys. Rev. Res.* **3** (2021), no. 2, article no. 023005.

[128] T. Ayral, F. M. Le Regent, Z. Saleem, Y. Alexeev and M. Suchara, "Quantum divide and compute: Hardware demonstrations and noisy simulations", in *Proceedings of IEEE Computer Society Annual Symposium on VLSI, ISVLSI*, 2020, pp. 138–140.

[129] T. Ayral, F.-m. L. Régent, Z. Saleem, Y. Alexeev and M. Suchara, "Quantum divide and compute: exploring the effect of different noise sources", *SN Comput. Sci.* **2** (2021), no. 3, article no. 132.

[130] M. Mohseni, A. Scherer, K. G. Johnson, et al., *How to build a quantum supercomputer: scaling challenges and opportunities*, preprint, 2024, 2411.10406.

[131] B. Anselme Martin, T. Ayral, F. Jamet, M. J. Ranvcić and P. Simon, "Combining matrix product states and noisy quantum computers for quantum simulation", *Phys. Rev. A* **109** (2024), no. 6, article no. 062437.

[132] S.-H. Lin, R. Dilip, A. G. Green, A. Smith and F. Pollmann, "Real- and imaginary-time evolution with compressed quantum circuits", *PRX Quantum* **2** (2021), no. 1, article no. 010342.

[133] M. Schwarz, K. Temme and F. Verstraete, "Preparing projected entangled pair states on a quantum computer", *Phys. Rev. Lett.* **108** (2012), no. 11, article no. 110502.

[134] M. Schwarz, K. Temme, F. Verstraete, D. Perez-Garcia and T. S. Cubitt, "Preparing topological projected entangled pair states on a quantum computer", *Phys. Rev. A* **88** (2013), no. 3, article no. 032321.

[135] Y. Wu, W.-S. Bao, S. Cao, et al., "Strong quantum computational advantage using a superconducting quantum processor", *Phys. Rev. Lett.* **127** (2021), no. 18, article no. 180501.

[136] Y. Kim, A. Eddins, S. Anand, et al., "Evidence for the utility of quantum computing before fault tolerance", *Nature* **618** (2023), no. 7965, pp. 500–505.

[137] J. Tindall, M. Fishman, E. M. Stoudenmire and D. Sels, "Efficient tensor network simulation of IBM's eagle kicked ising experiment", *PRX Quantum* **5** (2024), no. 1, article no. 010308.

[138] T. Begušić, J. Gray and G. K.-L. Chan, "Fast and converged classical simulations of evidence for the utility of quantum computing before fault tolerance", *Sci. Adv.* **10** (2024), no. 3, pp. 1–4.

[139] K. Kechedzhi, S. V. Isakov, S. Mandrà, B. Villalonga, X. Mi, S. Boixo and V. Smelyanskiy, "Effective quantum volume, fidelity and computational cost of noisy quantum processing experiments", *Future Gener. Comput. Syst.* **153** (2024), pp. 431–441.

[140] E. G. D. Torre and M. M. Roses, "Dissipative mean-field theory of IBM utility experiment", **1** (2023), pp. 1–4. Online at http://arxiv.org/abs/2308.01339.

[141] P. Besserve and T. Ayral, "Unraveling correlated material properties with noisy quantum computers: Natural orbitalized variational quantum eigensolving of extended impurity models within a slave-boson approach", *Phys. Rev. B* **105** (2022), no. 11, article no. 115108.

[142] P. Besserve, M. Ferrero and T. Ayral, *Compact fermionic quantum state preparation with a natural-orbitalizing variational quantum eigensolving scheme*, preprint, 2024, 2406.14170. p. 1–15.

[143] I. D. Kivlichan, J. McClean, N. Wiebe, C. Gidney, A. Aspuru-Guzik, G. K.-L. Chan and R. Babbush, "Quantum simulation of electronic structure with linear depth and connectivity", *Phys. Rev. Lett.* **120** (2018), no. 11, article no. 110501.

[144] Y. Lu, M. Höppner, O. Gunnarsson and M. W. Haverkort, "Efficient real-frequency solver for dynamical mean-field theory", *Phys. Rev. B* **90** (2014), no. 8, article no. 085102.

[145] Y. Lu, X. Cao, P. Hansmann and M. W. Haverkort, “Natural-orbital impurity solver and projection approach for Green’s functions”, *Phys. Rev. B* **100** (2019), no. 11, article no. 115134.

[146] F. Jamet, C. Lenihan, L. P. Lindoy, A. Agarwal, E. Fontana, B. A. Martin and I. Rungger, *Anderson impurity solver integrating tensor network methods with quantum computing*, preprint, 2023, 2304.06587. p. 1–8.

[147] V. Havlívcek, M. Troyer and J. D. Whitfield, “Operator locality in the quantum simulation of fermionic models”, *Phys. Rev. A* **95** (2017), no. 3, article no. 032332.

[148] F. Verstraete and J. I. Cirac, “Mapping local Hamiltonians of fermions to local Hamiltonians of spins”, *J. Stat. Mech.: Theory Exp.* **2005** (2005), no. 09, P09012–P09012.

[149] A. Michel, L. Henriet, C. Domain, A. Browaeys and T. Ayral, “Hubbard physics with Rydberg atoms: Using a quantum spin simulator to simulate strong fermionic correlations”, *Phys. Rev. B* **109** (2024), no. 17, article no. 174409.

[150] L. De’Medici, A. Georges and S. Biermann, “Orbital-selective Mott transition in multiband systems: Slave-spin representation and dynamical mean-field theory”, *Phys. Rev. B* **72** (2005), no. 20, article no. 205124.

[151] A. Rüegg, S. D. Huber and M. Sigrist, “Z2-slave-spin theory for strongly correlated fermions”, *Phys. Rev. B* **81** (2010), no. 15, article no. 155118.

[152] L. de’Medici and M. Capone, “Modeling many-body physics with Slave-Spin mean-field: Mott and Hund’s physics in Fe-superconductors”, in *The Iron Pnictide Superconductors*, Springer Series in Solid-State Sciences, Springer, 2017, pp. 115–185.

[153] M. Schiró and M. Fabrizio, “Quantum quenches in the Hubbard model: time-dependent mean-field theory and the role of quantum fluctuations”, *Phys. Rev. B* **83** (2011), no. 16, pp. 1–19.

[154] M. Sandri, M. Schiro and M. Fabrizio, “Linear ramps of interaction in the fermionic hubbard model”, *Phys. Rev. B* **86** (2012), article no. 075122.

[155] S. R. Hassan and L. de’Medici, “Slave spins away from half filling: cluster mean-field theory of the Hubbard and extended Hubbard models”, *Phys. Rev. B* **81** (2010), no. 3, article no. 035106.

[156] M. F. Serret, B. Marchand and T. Ayral, “Solving optimization problems with Rydberg analog quantum computers: realistic requirements for quantum advantage using noisy simulation and classical benchmarks”, *Phys. Rev. A* **102** (2020), no. 5, article no. 052617.

[157] M. Rader and A. M. Läuchli, “Finite correlation length scaling in lorentz-invariant gapless iPEPS wave functions”, *Phys. Rev. X* **8** (2018), no. 3, article no. 31030.

[158] T. Louvet, T. Ayral and X. Waintal, *On the feasibility of performing quantum chemistry calculations on quantum computers*, preprint, 2023, 2306.02620.

[159] C. Mora and X. Waintal, “Variational wave functions and their overlap with the ground state”, *Phys. Rev. Lett.* **99** (2007), no. 3, article no. 030403.

[160] D. Wu, R. Rossi, F. Vicentini, et al., “Variational benchmarks for quantum many-body problems”, *Science* **386** (2024), pp. 296–301.

[161] A. Abbas, A. Ambainis, B. Augustino, et al., “Quantum optimization: potential, challenges, and the path forward”, *Nat. Rev. Phys.* **6** (2024), article no. 718.

[162] F. Barahona, “On the computational complexity of ising spin glass models”, *J. Phys. A: Math. General* **15** (1982), no. 10, pp. 3241–3253.

[163] A. Y. Kitaev, A. H. Shen and M. N. Vyalyi, *Classical and Quantum Computation*, American Mathematical Society, Providence, RI, 2002.

[164] D. Aharonov and A. Ta-Shma, “Adiabatic quantum state generation and statistical zero knowledge”, in *Proceedings of the Thirty-Fifth Annual ACM Symposium on Theory of Computing*, ACM, 2003, pp. 20–29.

[165] H. Pichler, S.-T. Wang, L. Zhou, S. Choi and M. D. Lukin, *Quantum optimization for maximum independent set using Rydberg atom arrays*, preprint, 2018, 1808.10816.

[166] M. T. Nguyen, J. G. Liu, J. Wurtz, M. D. Lukin, S. T. Wang and H. Pichler, “Quantum optimization with arbitrary connectivity using Rydberg atom arrays”, *PRX Quantum* **4** (2023), no. 1, pp. 1–19.

[167] C. Dalyac, L.-P. Henry, M. Kim, J. Ahn and L. Henriet, “Exploring the impact of graph locality for the resolution of the maximum-independent-set problem with neutral atom devices”, *Phys. Rev. A* **108** (2023), no. 5, article no. 052423.

[168] S. Martiel, T. Ayral and C. Allouche, “Benchmarking quantum coprocessors in an application-centric, hardware-agnostic, and scalable way”, *IEEE Trans. Quantum Eng.* **2** (2021), pp. 1–11.

[169] E. Magesan, J. M. Gambetta and J. Emerson, “Characterizing quantum gates via randomized benchmarking”, *Phys. Rev. A* **85** (2012), no. 4, article no. 042311.

[170] E. Magesan, J. M. Gambetta and J. Emerson, “Scalable and robust randomized benchmarking of quantum processes”, *Phys. Rev. Lett.* **106** (2011), no. 18, article no. 180504.

[171] W. van der Schoot, D. Leermakers, R. Wezeman, N. Neumann and F. Phillipson, “Evaluating the Q-score of quantum annealers”, in *2022 IEEE International Conference on Quantum Software (QSW)*, IEEE, 2022, pp. 9–16.

[172] W. d. S. Coelho, M. D'Arcangelo and L.-P. Henry, *Efficient protocol for solving combinatorial graph problems on neutral-atom quantum processors*, preprint, 2022, 2207.13030.

[173] N. H. Stair and F. A. Evangelista, "Simulating many-body systems with a projective quantum eigensolver", *PRX Quantum* **2** (2021), no. 3, article no. 030301.

[174] J. Robledo-Moreno, M. Motta, H. Haas, et al., *Chemistry beyond exact solutions on a quantum-centric supercomputer*, preprint, 2024, 2405.05068.

[175] W. J. Huggins, B. A. O'Gorman, N. C. Rubin, D. R. Reichman, R. Babbush and J. Lee, "Unbiasing fermionic quantum Monte Carlo with a quantum computer", *Nature* **603** (2022), no. 7901, pp. 416–420.

[176] Y. Zhang, Y. Huang, J. Sun, D. Lv and X. Yuan, *Quantum computing quantum Monte Carlo*, preprint, 2022, 2206.10431. p. 1–7.