Quantum computing for nuclear reactions

Ionel Stetcu[1,*]

[1]Theoretical Division, Los Alamos National Laboratory, Los Alamos, New Mexico, 87545, USA

Abstract. Description of quantum many-body dynamics is extremely challenging on classical computers, as it requires taking into account many degrees of freedom. In nuclear physics, this translates on a large number of break up channels that have to be taken into account depending on available energy. Even using classical computing exascale capabilities will not allow a full description of dynamical processes starting from internucleon interactions beyond drastic approximations. On the other hand, the dynamics can be naturally implemented on quantum hardware that is designed to easily handle unitary transformations, and hence the hope that in the future we will be able to obtain reaction observables from first principles. In this contribution, I describe the main steps involved in quantum algorithms and review our efforts to apply quantum computing to the nuclear many-body problem using realistic inter-nucleon interactions.

1 Introduction

Quantum computing (QC) aims at revolutionizing modern approaches to solving many-body quantum dynamics. Little by little, Feynman's vision [1] of getting to a point where "the computer will do *exactly* the same as nature" could become a reality in the near future. In the same way we are living now in a period where artificial intelligence is promoting a new technological revolution, in the next years we could get to a point when quantum computers will be ubiquitous and solving the many-body quantum dynamics will become a lot easier, in principle. Technological breakthroughs over the last few years, combined with efforts in algorithm design and error corrections, are important steps towards delivering the goal of solving problems that are beyond classical computing capabilities, and which could lead to important advancements in our theoretical understanding of nature.

Nuclear reactions are perfect examples of complicated quantum many-body dynamics. Depending on the energies involved in the process, multiple reaction channels could be in competition. This makes not only the theoretical description challenging, but also experimental measurements. A very simple example is the case of ^{235}U, where the first excited state lies at 76 eV above the ground state. Thus, this makes it extremely difficult for experimentalists to separate the elastic from inelastic process; hence, one cannot expect a precise measurement of the inelastic scattering cross sections at the threshold. (This is not to say that with QC algorithms separating such a low-lying state would be easy.) The (n,2n) channel is also plagued by similar obstacles as more channels are usually open when the (n,2n) reaction becomes energetically allowed. For actinides, the situation is even worse, as fission competes with all the other channels even at low energies, and in fission one emits neutrons and gamma rays from excited fission fragments, which makes it very difficult to distinguish it from other channels. As an example, see the recent measurement [2] of the ^{233}U ratio of the capture-to-fission cross sections for fast neutrons, which is hard to reconcile with a previous experiment [3] and phenomenological Hauser-Feshbach calculations with parameters tuned in the resonance region [2].

Modern theoretical approaches to describing fission observables range from the R-matrix [4] to the statistical model of Hauser and Feshbach [5]. Such models make use of drastic approximations to solve the nuclear many-body problem. They are successful as long as data are available to tune the (many) parameters. Where no data are available, the confidence of the predictions is low and can vary wildly from practitioner to practitioner, even if they are using the same underlying model, due to differences in numerical implementations and phenomenological parameters. Fully microscopic models that solve the nuclear many-body dynamics using realistic nucleon-nucleon interactions have a small range of applicability, and are restricted to a few light nuclei targets [6–10], and even in those cases where they exist, additional approximations are often required.

In this contribution, I paint with thick strokes the landscape of QC applications to nuclear dynamics. The picture will be very blurry because the current state of the hardware does not allow us to be more precise. Quantum gates that are used to implement the true quantum correlations in physical states have somewhat large errors, while the relatively short decoherence time does not allow the application of a large number of gates necessary to describe nuclear processes. Fast progress in quantum technologies allows us to hope that in the longer term applications to nuclear reactions will be possible, and hence the time to develop new algorithms is now.

*e-mail: stetcu@lanl.gov

2 Quantum computing and nuclear dynamics

Simulating quantum processes using classical computers has the advantage that the user has direct access to the wave function, which can be manipulated as necessary and even stored indefinitely after the computation. From the beginning one has to observe that this is not the case on quantum hardware that imitates nature, in Feynman's words. Like nature, one can only access the wave function by measurements, and each measurement destroys the wave function, or more accurately projects into one of the allowed configurations. The measurement needs to be performed a sufficiently large number of times to extract the desired expectation value. Obtaining different observables will require the preparation of the state again and taking again other measurements. Furthermore, measurements can be used to prepare the initial state.

Solving for the dynamics of a quantum many-body system requires three main steps: (i) preparation of the initial state, (ii) time evolution with the system with the Hamiltonian, $\exp(-iHt)$, and (iii) measurements of observables (and extraction of desired quantities like response functions or cross sections). Before discussing these steps in more detail, one also needs to mention that there are different ways to map the problem to hardware.

There are several encodings available, all with their own advantages and disadvantages. In the second quantization mapping, the number of qubits required to represent the problem is equal to the number of states. In the case of a reaction calculation, where the problem is represented on a lattice, the number of qubits is proportional to the number of lattice points, in other words, the number of points scales with the volume. This becomes prohibitive very quickly, as description of nuclear reactions requires the description of asymptotic states, and hence large lattices. A complementary choice is to use the first quantization encoding, in which each single-particle state is encoded into a unique superposition of qubits in states $|0\rangle$ or $|1\rangle$. This requires $\log_2(N_s)$ qubits to represent all the single-particle states, where N_s is the number of states for each type of particle (protons or neutrons), in total $N_p \log_2(N_s)$ qubits for N_p particles. While economical in terms of qubits, this mapping suffers from the fact that for a system of identical fermions, the states need to be antisymmetrized, which requires considerable resources for more than 2-3 particles. An intermediate mapping was proposed in Ref. [11], where the *antisymmetric* basis states are encoded in superpositions of qubits in states $|0\rangle$ or $|1\rangle$. The advantage of this encoding is that one can take into account any additional constraints when generating the basis states, like preserving the J_z projection for M-scheme shell model-like calculations or zero total linear momentum in the case when the problem is represented in a lattice. In recent work [12], we have used this encoding to describe the deuteron in a 8^3 lattice, using contact interactions. The proton-neutron bound state was prepared and then the response to electron scattering was calculated using QC algorithms. The number of qubits is $\log_2$ of the number of lattice points, and hence the resources scale logaritmically with volume. The mapping of any operator is performed considering that $\hat{O} = \sum_{\alpha,\beta}\langle\alpha|O|\beta\rangle|\alpha\rangle\langle\beta|$, where $\langle\alpha|O|\beta\rangle$ are classically computed matrix elements, and $|\beta\rangle\langle\alpha|$ are encoded in Pauli operators by taking the well-known identities $|0\rangle\langle 0| = (I+Z)/2$, $|1\rangle\langle 1| = (I-Z)/2$, $|1\rangle\langle 0| = (X-iY)/2$, and $|0\rangle\langle 1| = (X+iY)/2$. Here, I is the identity 2×2 matrix, and X, Y, and Z are the Pauli matrices. Since we work in momentum space, the kinetic energy operator is diagonal, and hence it maps into combinations of I and Z only, while the interaction is a matrix with constant elements C_0, where C_0 is the strength of the contact interaction. This corresponds to $(I+X)^{N_q}$, where N_q is the number of qubits used to represent the physical system. As expected, the kinetic energy and interaction do not commute, so the time evolution with this Hamiltonian will require the use of the Trotter approximation.

State preparation. In order to start any algorithm, it is necessary to prepare an initial state. For example, to describe neutron-induced reactions one would most likely need to prepare the target nucleus in the ground state, while the neutron could be prepared as a Gaussian package, far away from the target.

On quantum hardware, preparing the initial state is a very difficult problem. Unitary coupled cluster-like techniques are very popular, as they are unitary transformations that are well suited to be implemented on quantum computers with relatively few resources required. However, when a very accurate state is necessary, one often needs to project it into a desired subspace designed to contain the target state. As a projection is not a unitary transformation, it can be accomplished by measurements. A schematic circuit implementing this task is shown in Fig. 1 [13]. The trial state $|\psi\rangle$ can be easily prepared on the hardware with a minimum number of gates, and ideally has a large overlap with the exact target state $|\tilde{\psi}\rangle$, since the success probability of the algorithm is equal (in the absence of hardware errors) with $|\langle\tilde{\psi}|\psi\rangle|^2$. Each application of the algorithm produces the state $\exp(-iY_a \otimes (t_i O + \delta_i)|\psi\rangle|0\rangle_a = \cos(t_i O + \delta_i)|\psi\rangle|0\rangle + \sin(t_i O + \delta_i)|\psi\rangle|1\rangle$. We choose times t_i and phases δ_i so that the target state would be included in the subspace with ancilla in state $|0\rangle$. Note that the operator O is general. If we want to project a deformed state (Hartree-Fock) into a fixed angular momentum, we can choose $O = J^2 - J_0(J_0+1)$, where J_0 is the target value of the angular momentum operator. Because the eigenvalues of such an operator are known, we choose all $\delta_i = 0$, while the times will depend on the value of J_0. For example if $J_0 = 0$, using $t_1 = \pi/4$ to evolve the trial and measuring the ancilla in state $|0\rangle$ eliminates $J = 1, 2, 5, 6, 9, \ldots$ states, as for these $\cos(J(J+1)\pi/4) = 0$. In a second step, taking $t_2 = \pi/8$ eliminates $J = 3, 4, 11, 12, \ldots$

The same can be done by taking O as the Hamiltonian. In this case, the spectrum is not known in general. However, very little information about the spectrum can be used to filter the target state. The maximum evolution time is given by $t_1 = \pi/2\Delta$, where Δ is the energy difference between the target state and the next closed one. For the ground state, Δ is the usual gap. In our experience, taking $t_i = t_{i-1}/2$, for $i > 1$ already performs very well, but knowledge about the energy spectrum (from experiments

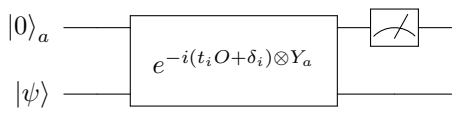

Figure 1. Circuit for state preparation using projection with one ancilla measurement, see Ref. [13] for details. The physical state $|\psi\rangle$ is represented on an appropriate number of qubits. Y_a is the Pauli Y matrix acting on the ancilla qubit a.

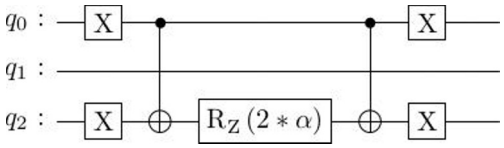

Figure 2. Quantum circuit for performing operation $\exp(-i\alpha XIX)$, using IBM's Qiskit SDK.

or approximate calculations) can be used to obtain optimal times. In addition, one can include in the optimization some small non-zero phases δ_i that can produce an exponential speedup over other methods [13]. After an appropriate number of measurements of the ancilla, the initial state $|\psi\rangle$ is projected to the desired state.

Once the ground state has been prepared, we follow the dynamics under an external probe. In nuclear reactions, this could be another nucleus or a proton/neutron projectile. For a first application, we have considered electron scattering, which is modeled by applying the unitary operator $\exp(-i\vec{q}\vec{r})$ to the ground state and using the new state to start the time evolution. This task was accomplished with few resources compared to the time evolution and ground state preparation, and it consists on mapping the operator $\vec{r}$ (since we work in momentum space) into qubits.

Time evolution. The dynamics simulation proceeds by applying $\exp(-iHt)$ to the perturbed state produced after perturbing the system with $\exp(-i\vec{q}\vec{r})$. There are standard techniques to build a quantum circuit for time evolution, once the Hamiltonian is mapped to a sum of Pauli strings. In addition to the Trotter approximation [14], it is often important to consider the order of the Pauli strings in the sum, so as to cancel as many entangling CNOT gates. A CNOT gate performs a flip on the target qubit if the controlling qubit is in the state $|1\rangle$. Two CNOTs in a row acting on the same qubits cancel each other (if the controlling qubit is in state $|0\rangle$, neither CNOT changes the state of the target qubit, while if the controlling qubit is in state one $|1\rangle$, one performs two flips of the target qubit, so its state does not change again). This can reduce the number of CNOTs required to perform the calculation.

In QC, all operators are mapped into a sum of Pauli strings, as discussed at the beginning of this section. Thus, the Hamiltonian can be written as $H = \sum_k c_k P_k$, where P_k stands for a generic Pauli string that is formed by the tensor product of N_q single-qubit Pauli operators. For example, in a three qubit circuit, let us take $P = XIX$, which means that one acts with a X operation on the first and the last qubit, leaving the middle qubit unchanged. Exponentiation of $-i\alpha P$ produces the result $\exp(-i\alpha XIX) = \cos(\alpha)III - i\sin(\alpha)XIX$, where α is real. After the application on an initial three-qubit state, the result will be a new state which is a superposition of the original one, and another one in which the first and last qubits are flipped. The quantum circuit is illustrated in Fig. 2, using Qiskit [15]. The evolution circuit will be composed from similar (and more complicated) collections of gates, depending on the structure of the Pauli strings present in the expansion. As not all the Pauli strings commute, one usually needs to consider the Trotter approximation implying taking small enough time steps (small α in our example). Other approaches exist, like in the case of classical computing.

As discussed before, in momentum space the kinetic energy maps into a relatively small number of Pauli strings that involve only I and Z matrices acting on up to a few qubits at the time, and are implemented via (controlled) qubit rotations similar to Fig. 2.

The interaction is much more demanding because it contains 2^{N_q} Pauli strings that contain all possible combinations of I and X Pauli matrices [12]. The ordering of these strings is arbitrary since they do commute among themselves. However, Welch *et al.* [16] have shown that ordering Pauli strings based on the Grey Code produces a very efficient circuit if one takes into account all the properties of CNOT gates. Thus, this allows one to obtain a circuit with a single CNOT between two adjacent Z rotations, with a total of $2^{N_q} - 2$ CNOTs and $2^{N_q} - 1$ Z rotations in the exact application of $\exp(-iVdt)$ [12, 16]. We note that in fault-tolerant QC, the CNOT operations are relatively easy to perform, while rotations are more difficult. Hence, we are working on another circuit that will allow us to reduce in this particular case the number of rotations, while adding ancilla qubits to perform the same time evolution.

Measurements. After the system evolves in time, measurements are necessary to extract useful information about the nuclear process. This can be in the form of extracting an expectation value, after the operator has been consistently mapped into Pauli strings. One very distinct difference with respect to classical computing is the fact that one does not have direct access to the wave function. Hence, one needs to design the correct measurements to extract the information. For example, calculating an expectation value of an operator written as a superposition of strings reduces to calculating the expectation value of each string. In other cases it is convenient to couple the physical system with one or more ancilla, like in Fig. 1.

One of the most used algorithms in QC is Quantum Phase Estimate (QPE) [17]. In this case, the qubits corresponding to the physical system are coupled to a set of ancilla qubits that are individually used to perform a controlled time evolution (if their state is $|1\rangle$). After an application of a Quantum Fourier Transform [18] only to the ancilla qubits, QPE produces a decomposition of the initial state $|\psi_0\rangle$ into eigenstates of the Hamiltonian coupled with ancilla states, $|\psi_0\rangle = \sum_k c_k |E_k\rangle|k\rangle$ (here one should interpret $|k\rangle$ as the binary representation of integer k). Per-

forming the algorithm many times and measuring only the ancillas, one obtains the probability $|c_k|^2 = \langle E_k|\psi_0\rangle|^2$.

Roggero and Carlson have adapted QPE to obtain the strength function [19], choosing $|\psi_0\rangle = O|\phi_0\rangle$ as input to QPE, where $|\phi_0\rangle$ is a fixed eigenstate. The result is the strength function $|\langle E_k|O|\phi_0\rangle|^2$. Taking $|\phi_0\rangle$ as the ground state of the deuteron and $O = \exp(-i\vec{q}\vec{r})$, we have used this algorithm in Ref. [12] to calculate the response function of the deuteron to electron scattering. Given that the noise (errors in quantum gate execution) is and will remain a problem for the foreseeable future, we investigated its impact and found that in applications to deuteron a depolarizing noise level of 5×10^{-4} was at the upper limit where we can still obtain reasonable results as compared to classical calculations. The depolarizing noise is due to errors in applications of the CNOT gates, either because the target qubit is flipped even though the controlling qubit is in state $|0\rangle$, or the target qubit is not flipped when the controlling qubit is in state $|1\rangle$. As the CNOT gates are building correlations between states, controlling errors and understanding the maximum level of noise that still allows for quality simulations are essential.

3 Summary

Quantum computing could become the method of choice for solving difficult many-body quantum dynamics. We are now in the developing stages of fault-tolerant hardware that will still need to advance to be useful for nuclear physics. New quantum algorithms that are somewhat different from their classical computing counterparts will need to be developed, in particular for projecting into different channels. Such algorithms will have to be adapted to the fault-tolerant computing capabilities, which requires intimate knowledge about the architecture. Nuclear physics is an area where QC can help us get better predictions in the future, in particular for nuclear reactions and, by extension, nuclear data.

Acknowledgement

I thank A. Baroni, J. Carlson and R. Weiss for many discussions and collaboration on QC applications to nuclear dynamics. Partial funding for this work was provided by the Advanced Simulations and Computing program.

References

[1] R.P. Feynman, Simulating physics with computers, International Journal of Theoretical Physics **21**, 467 (1982). 10.1007/BF02650179

[2] E. Leal-Cidoncha, A. Couture, E.M. Bond, T.A. Bredeweg, C. Fry, T. Kawano, A.E. Lovell, G. Rusev, I. Stetcu, J.L. Ullmann et al., Measurement of the neutron-induced capture-to-fission cross section ratio in ^{233}U at LANSCE, Phys. Rev. C **108**, 014608 (2023). 10.1103/PhysRevC.108.014608

[3] J.C. Hopkins, B.C. Diven, Neutron Capture to Fission Ratios in U233, U235, Pu239, Nucl. Sci. Eng. **12**, 169 (1962). 10.13182/NSE62-A26055

[4] P. Descouvemont, D. Baye, The r-matrix theory, Reports on Progress in Physics **73**, 036301 (2010). 10.1088/0034-4885/73/3/036301

[5] W. Hauser, H. Feshbach, The inelastic scattering of neutrons, Phys. Rev. **87**, 366 (1952). 10.1103/PhysRev.87.366

[6] K.M. Nollett, S.C. Pieper, R.B. Wiringa, J. Carlson, G.M. Hale, Quantum monte carlo calculations of neutron-α scattering, Phys. Rev. Lett. **99**, 022502 (2007). 10.1103/PhysRevLett.99.022502

[7] S. Quaglioni, P. Navrátil, Ab initio many-body calculations of n–^{3}H, n–^{4}He, p–^{3,4}He, and n–^{10}Be scattering, Phys. Rev. Lett. **101**, 092501 (2008). 10.1103/PhysRevLett.101.092501

[8] W. Leidemann, Few-nucleon physics, **168**, 012001 (2009). 10.1088/1742-6596/168/1/012001

[9] M. Atkinson, K. Kravvaris, S. Quaglioni, P. Navrátil, Ab initio calculation of the ^{3}He$(\alpha,\gamma)^7$Be astrophysical S factor with chiral two- and three-nucleon forces, Physics Letters B **860**, 139189 (2025).

[10] A.R. Flores, K.M. Nollett, Variational Monte Carlo calculations of n +^{3} H scattering, Phys. Rev. C **108**, 034001 (2023). 10.1103/PhysRevC.108.034001

[11] Y. Shee, P.K. Tsai, C.L. Hong, H.C. Cheng, H.S. Goan, Qubit-efficient encoding scheme for quantum simulations of electronic structure, Phys. Rev. Res. **4**, 023154 (2022). 10.1103/PhysRevResearch.4.023154

[12] R. Weiss, A. Baroni, J. Carlson, I. Stetcu, Reaction dynamics with qubit-efficient momentum-space mapping (2024), arXiv:2404.00202. 10.48550/arXiv.2404.00202

[13] I. Stetcu, A. Baroni, J. Carlson, Projection algorithm for state preparation on quantum computers, Phys. Rev. C **108**, L031306 (2023). 10.1103/PhysRevC.108.L031306

[14] H.F. Trotter, On the product of semi-groups of operators, Proceedings of the American Mathematical Society **10**, 545 (1959). 10.2307/2033649

[15] A. Javadi-Abhari, M. Treinish, K. Krsulich, C.J. Wood, J. Lishman, J. Gacon, S. Martiel, P.D. Nation, L.S. Bishop, A.W. Cross et al., Quantum computing with Qiskit (2024), 2405.08810

[16] J. Welch, D. Greenbaum, S. Mostame, A. Aspuru-Guzik, Efficient quantum circuits for diagonal unitaries without ancillas, New Journal of Physics **16**, 033040 (2014). 10.1088/1367-2630/16/3/033040

[17] A.Y. Kitaev, Quantum measurements and the Abelian Stabilizer Problem, arXiv e-prints (1995), quant-ph/9511026. 10.48550/arXiv.quant-ph/9511026

[18] D. Coppersmith, An approximate Fourier transform useful in quantum factoring, arXiv e-prints (2002), quant-ph/0201067. 10.48550/arXiv.quant-ph/0201067

[19] A. Roggero, J. Carlson, Dynamic linear response quantum algorithm, Phys. Rev. C **100**, 034610 (2019). 10.1103/PhysRevC.100.034610