Universal algorithm for transforming Hamiltonian eigenvalues

Tatsuki Odake[1] Hlér Kristjánsson[1, 2, 3] Philip Taranto[1] Mio Murao[1]

[1]*Department of Physics, Graduate School of Science, The University of Tokyo, Hongo 7-3-1, Bunkyo-ku, Tokyo, Japan*
[2]*Perimeter Institute for Theoretical Physics, 31 Caroline Street North, Waterloo, Ontario, N2L 2Y5, Canada*
[3]*Institute for Quantum Computing, University of Waterloo, 200 University Avenue West, Waterloo, Ontario, N2L 3G1, Canada*

1. Introduction & summary. Manipulating Hamiltonians governing physical systems is a crucial task in various areas of science, finding applications from quantum chemistry [1–3] to materials design [4–6]. Although *system-dependent* and *empirical* techniques, such as tuning the composition of alloy [7, 8], exist, quantum algorithms can offer *universal* and *theoretically-guaranteed* means of manipulating Hamiltonians, e.g., via quantum singular value transformation [9–11]. That being said, existing quantum algorithms for transforming the eigenvalues of Hamiltonians rely on access to an encoding of the target Hamiltonian H, e.g., block encoding [9, 10], the controlled Hamiltonian dynamics $\texttt{ctrl}e^{\pm iHt}$ [12–17], or a quantum walk oracle [18, 19], which limits their applicability due to the high cost of estimating/reconstructing the unknown Hamiltonian.

In this work [20], we present a quantum algorithm that simulates the Hamiltonian dynamics of $f(H)$ (defined as $f(H) := \sum_n f(\lambda_n) |E_n\rangle\langle E_n|$ for Hamiltonian diagonalized as $\sum_n \lambda_n |E_n\rangle\langle E_n|$), for any (suitably differentiable) function f only using the positive- and negative- time evolution $e^{\pm iHt}$ of a black box seed Hamiltonian H. Our algorithm implements the transformation of Hamiltonian using a novel subroutine *Fourier series subroutine*, that samples random circuit instances according to the distribution of Fourier coefficients associated with the function f. This process requires access to positive- and negative- controlled dynamics $\texttt{ctrl}e^{\pm iHt}$, which is constructed using another subroutine *controlization* [21]. The time-complexity of our algorithm is $\Theta(C_{4,f} t^2 n/\epsilon)$ for an f-dependent constant $C_{4,f}$, time t, qubit number n and precision ϵ. Regarding ϵ scaling, our algorithm performs better than a QSVT-based algorithm. Crucially, our algorithm does not rely on the classical description of the Hamiltonian, which is motivated by the theoretical framework of the higher-order quantum transformation [21–32] (which considers the situation where the possible inputs to an algorithm are quantum channels specified by their input-output description only). This not only helps extend the class of possible eigenvalue transformation tasks but also opens a way to the Hamiltonian simulation performed via the transformation of unknown Hamiltonian dynamics.

Our work opens new avenues for quantum programming by establishing a general framework for quantum algorithm compilation. Our algorithm uses correlated randomness to efficiently combine two subroutines beyond straightforward concatenation. We abstracted key features that empower our algorithm, providing a method to harness correlated randomness to globally optimize functional programs, rather than naïvely concatenating them (which leads to suboptimal performance). This development potentially offers a solution to the inefficiency in quantum functional programming, a paradigm in quantum programming allowing flexible program development through the concatenation of modular quantum transformations [28, 33, 34].

Furthermore, to enhance the realm of applicability of our algorithm, we present an improved method to simulate the negative-time dynamics of an unknown Hamiltonian from the positive time dynamics, circumventing the explicit need for access to the former. Crucially, this procedure does not require an auxiliary system and has a runtime independent of the number of qubits, instead being bound by the interaction degree of the Hamiltonian. This method enables the approximate implementation of our Hamiltonian eigenvalue transformation algorithm by applying a short pulse sequence of gates on top of continuously running Hamiltonian dynamics, assuming the ability to apply these gates on a much shorter timescale.

2. Universal Hamiltonian eignevalue transformation algorithm.

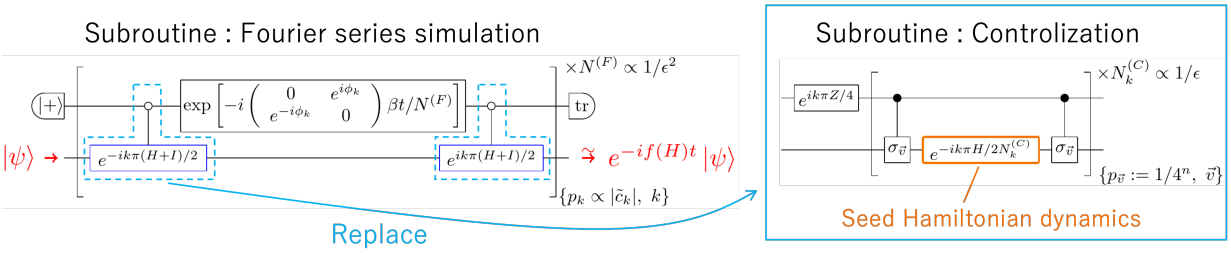


Figure 1: The uncompiled algorithm obtained by concatenating two subroutines can already implement $e^{-if(H)t}$, but is further modified/compiled to improve the ϵ scaling of time complexity.

Algorithm 1 Efficient universal Hamiltonian eigenvalue transformation (Compiled)]

Input:

- A finite number of queries to a black-box Hamiltonian dynamics $e^{\pm iH\tau}$ of a seed Hamiltonian H normalized as $\|H_0\|_{\mathrm{op}} \leq 1$, where H_0 is the traceless part of H, i.e., $H_0 := H - (1/2^n)\mathrm{tr}(H)I$, with $\tau > 0$
- A sufficiently differentiable function $f : [-1, 1] \to \mathbb{R}$
- Input state $|\psi\rangle \in \mathcal{H}$, simulation time $t > 0$, allowed error $\epsilon > 0$

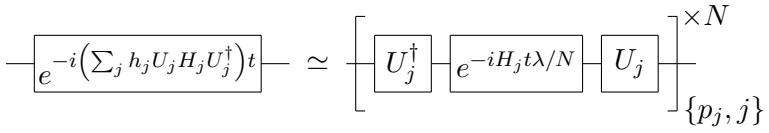
Output: A state approximating $e^{-if(H_0)t}|\psi\rangle$ with an error $\leq \epsilon$ and the mean square of error $\leq 2\epsilon$

Time complexity:

Pre-processing (only once): $\Theta(\grave{K}^3 t^3 n/\epsilon^3)+T_4$, $\grave{K} = O((t/\epsilon)^{1/3})$ ($\grave{K}$: cutoff number of the Fourier series)

Main Process: $\Theta(C_{4,f} t^2 n/\epsilon)$ for an f-dependent constant $C_{4,f}$ which is independent of n, t, and ϵ

Our algorithm is constructed in two steps. The first step is to concatenate two subroutines, Fourier series simulation (or FSS for short) and controlization as shown in Fig. 1. In terms of functionality, the algorithm at this stage, which we refer to as the *uncompiled algorithm*, can already implement $e^{-if(H)t}$ ($\mathrm{tr}(H) = 0$ is assumed for simplicity). Both FSS and controlization are based on the general approximation technique based on Trotterization for simulating Hamiltonian of form $\sum_j h_j U_j H_j U_j^\dagger$, where $\{h_j\}$ is a set of positive number, $\{U_j\}$ is a set of unitaries, and $\{H_j\}$ is a set of Hermitian matrices, thus the uncompiled algorithm consists of two layers of randomization. This technique is represented by the following circuit:



(1)

where $\lambda := \sum_j h_j$ and $p_j := h_j/\lambda$. This circuit repeats the process of applying $U_j e^{-iH_j t\lambda/N} U_j^\dagger = e^{-iU_j H_j U_j^\dagger t\lambda/N}$ N times (which depends on the precision ϵ as $1/\epsilon$), for an index j randomly chosen with probability p_j, similarly to qDRIFT [35]. In particular, the following Hamiltonian is simulated using Circuit. (1) for FSS and controlization:

$$\text{FSS: } \sum_{k=-K}^{K} |\tilde{c}_k| \begin{pmatrix} e^{ik\pi(H+I)/2} & 0 \\ 0 & I \end{pmatrix} \begin{pmatrix} 0 & e^{i\phi_k} I \\ e^{-i\phi_k} I & 0 \end{pmatrix} \begin{pmatrix} e^{-ik\pi(H+I)/2} & 0 \\ 0 & I \end{pmatrix} \simeq \begin{pmatrix} 0 & f(H) \\ f(H) & 0 \end{pmatrix}$$

$$\text{Controlization: } \sum_{\vec{v}\in\{0,1,2,3\}^n} \frac{1}{4^n} \begin{pmatrix} I & 0 \\ 0 & \sigma_{\vec{v}} \end{pmatrix} \begin{pmatrix} H & 0 \\ 0 & H \end{pmatrix} \begin{pmatrix} I & 0 \\ 0 & \sigma_{\vec{v}} \end{pmatrix} = \begin{pmatrix} H & 0 \\ 0 & 0 \end{pmatrix} \tag{2}$$

where $\tilde{c}_k =: |\tilde{c}_k| e^{i\phi_k}$ is the Fourier coefficient for frequency k of a function $\tilde{f}$ (a modification of f such that $f(x) = \tilde{f}((x+1)/2)$, $|\tilde{c}_k|$ converges rapidly), $K > 0$ is a cutoff number, and $\sigma_{\vec{v}} := \sigma_{v_1} \otimes \cdots \otimes \sigma_{v_n}$ with $\sigma_0 = I$, $\sigma_1 = X$, $\sigma_2 = Y$, and $\sigma_3 = Z$. FSS utilizes $f(H) = \tilde{f}((H+I)/2) = \sum_{k=-\infty}^{\infty} \tilde{c}_k e^{-ik\pi(H+I)/2}$. Note that, QSVT can be used instead of FSS for the same task since a block-encoding of $\cos H$ can be constructed using $\texttt{ctrl}e^{\pm iHt}$; indeed, such a QSVT-based algorithm is more efficient than the *uncompiled* algorithm.

The second step is to compile the circuit in Fig. 1 by introducing a correlation between the randomness in the controlization subroutine thereby reducing the ϵ scaling of the time complexity even beyond that of the QSVT-based algorithm. The properties of this *compiled algorithm* is shown in Algorithm 1. The time complexity of the *pre-processing* step, which refers to the processes that only need to be run once for a given set of inputs, scales as $1/\epsilon^4$ at most. Nevertheless, the effect of the large time complexity of the preprocessing step is diluted in the situation where one wants to repetitively implement the transformed Hamiltonian dynamics, which we refer to as the *main process*, for a fixed set of inputs. We intuitively explain two factors that make the main process of Algorithm 1 more efficient in terms of ϵ scaling than that of the QSVT-base algorithm:

(a) Efficient inputting of high-frequency terms: QSVT requires d queries in total to the block-encoding unitary and its inverse when implementing a polynomial function of degree d. On the other hand, the time complexity of the main process of Algorithm 1 has no explicit dependence on the cutoff number $\grave{K}$ of the frequency summation, since the information of Fourier coefficients $\tilde{c}_k$ is input by random sampling according to the magnitude of $\tilde{c}_k$. Such high-frequency terms $\tilde{c}_k$ for larger k have smaller magnitude $|\tilde{c}_k|$ than their low-frequency counterparts and therefore costly unitaries (i.e., for large k) are rarely chosen.

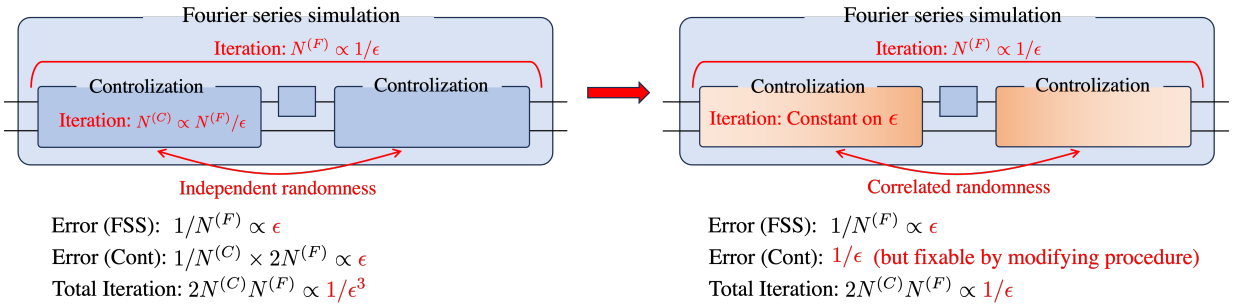


Figure 2: Sketch of compilation. First, the sampling randomnesses for two controlization subroutines in each iteration are made correlated. At the same time, iteration number $N^{(C)}$ of controlization is made independent on ϵ in order to improve ϵ scaling of total iteration.

(b) Iteration number in controlization procedure is independent of ϵ: In order to achieve the precision ϵ, the iteration number $N_k^{(C)}$ of controlization has to be taken $1/\epsilon$ for the QSVT-based algorithm. On the other hand, $N_k^{(C)}$ can be taken independent of ϵ for Algorithm 1 as described in the next section.

3. Technical background of compilation. The uncompiled algorithm has a suboptimal time complexity as a result of concatenating two layers of randomization. To make both the error due to FSS and controlization at most of the order ϵ, the iteration number $N^{(F)}$ for FSS and $N^{(C)}$ for controlization has to scale as $1/\epsilon$ and $N^{(F)}/\epsilon \propto 1/\epsilon^2$, respectively. This makes the overall time complexity scale as $1/\epsilon^2$ (see the left side of Fig. 2). To improve this scaling, the randomness for two controlization subroutines in each iteration is made correlated so that one becomes a Hermitian conjugate of the other. At the same time, the iteration number $N^{(C)}$ for compilation is redefined to be independent on ϵ. This comes with the cost of increasing the error, but because of the more symmetric structure of the circuit, this error can be compensated for by modifying the procedure, which completes the compilation. As a result, the time complexity, which is proportional to $2N^{(C)}N^{(F)}$, only scales as $1/\epsilon$. This framework of compilation performed by introducing correlated randomness, reducing the iteration number, and compensating the error, can be generalized to a wider range of quantum algorithms constructed by concatenation of the (random) Trotterization-based subroutines as discussed in the main paper.

4. Simulation of negative time evolution without auxiliary qubits. We also propose an algorithm to simulate negative time-evolution e^{+iHt} of unknown Hamiltonian H using the positive time-evolution e^{-iHt}. Even in situations where H is unknown, one can often reasonably assume some form of its structure, e.g., that it has at most two-body interactions or it is based on qubit connectivities. Suppose that the connectivity graph of a seven-qubit Hamiltonian H is given by the graph in Fig. 3, where the nodes/edges represent qubits/pairs of qubits for which H might have interactions (see main paper for the general case). Then one can first assign colors to qubits in such a way that any adjacent qubits have different colors as shown in the right side of Fig. 3. Then the negative time evolution can be implemented using Circuit. (1) by simulating the following Hamiltonian using random n-qubit Pauli operators $G_{v_r,v_b,v_g} := \sigma_{v_r}^{(1,3,4)} \otimes \sigma_{v_b}^{(2,5,6)} \otimes \sigma_{v_g}^{(7)}$:

$$\sum_{(v_r,v_b,v_g)\in\{0,1,2,3\}^3\setminus\{(0,0,0)\}} G_{v_r,v_b,v_g} H G_{v_r,v_b,v_g}^\dagger = \left(\sum_{(v_r,v_b,v_g)\in\{0,1,2,3\}^3} G_{v_r,v_b,v_g} H G_{v_r,v_b,v_g}^\dagger \right) - H = -H. \quad (3)$$

With this technique, Algorithm 1 can be implemented even without the negative time-evolution e^{+iHt}.

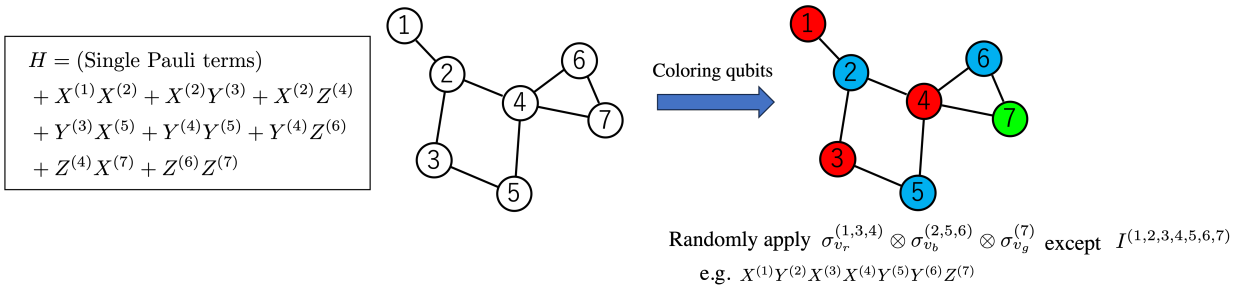


Figure 3: Demonstration of negative time evolution protocol. The graph expresses the connectivity of seven qubits, i.e. Hamiltonian can have interaction terms between connected nodes (qubits). Negative time-evolution can be implemented by applying random Pauli operators depending on the coloring of the graph.

References

[1] R. N. Zare, "Laser control of chemical reactions," Science, vol. 279, no. 5358, pp. 1875–1879, 1998.

[2] R. Hiraoka, E. Minamitani, R. Arafune, N. Tsukahara, S. Watanabe, M. Kawai, and N. Takagi, "Single-molecule quantum dot as a kondo simulator," Nature communications, vol. 8, no. 1, p. 16012, 2017.

[3] H. Teramae, K. Ohtawara, T. Ishimoto, and U. Nagashima, "Molecular structure optimization and molecular dynamics using hamiltonian algorithm: Structure of benzodiazepine minor tranquilizers—towards non-empirical drug design—," Bulletin of the Chemical Society of Japan, vol. 81, no. 9, pp. 1094–1102, 2008.

[4] M. T. Greiner, M. G. Helander, W.-M. Tang, Z.-B. Wang, J. Qiu, and Z.-H. Lu, "Universal energy-level alignment of molecules on metal oxides," Nat. Mater., vol. 11, no. 1, pp. 76–81, 2012.

[5] T. Iadecola, D. Campbell, C. Chamon, C.-Y. Hou, R. Jackiw, S.-Y. Pi, and S. V. Kusminskiy, "Materials design from nonequilibrium steady states: driven graphene as a tunable semiconductor with topological properties," Physical review letters, vol. 110, no. 17, p. 176603, 2013.

[6] A. R. Supka, T. E. Lyons, L. Liyanage, P. D'Amico, R. A. R. Al Orabi, S. Mahatara, P. Gopal, C. Toher, D. Ceresoli, A. Calzolari, et al., "Aflowπ: A minimalist approach to high-throughput ab initio calculations including the generation of tight-binding hamiltonians," Computational Materials Science, vol. 136, pp. 76–84, 2017.

[7] M. Levesque, E. Martínez, C.-C. Fu, M. Nastar, and F. Soisson, "Simple concentration-dependent pair interaction model for large-scale simulations of fe-cr alloys," Physical Review B, vol. 84, no. 18, p. 184205, 2011.

[8] C.-H. Huang and J. Marian, "A generalized ising model for studying alloy evolution under irradiation and its use in kinetic monte carlo simulations," Journal of Physics: Condensed Matter, vol. 28, no. 42, p. 425201, 2016.

[9] G. H. Low and I. L. Chuang, "Hamiltonian simulation by qubitization," Quantum, vol. 3, p. 163, 2019.

[10] A. Gilyén, Y. Su, G. H. Low, and N. Wiebe, "Quantum singular value transformation and beyond: exponential improvements for quantum matrix arithmetics," in Proceedings of the 51st Annual ACM SIGACT Symposium on Theory of Computing, pp. 193–204, 2019.

[11] J. M. Martyn, Z. M. Rossi, A. K. Tan, and I. L. Chuang, "Grand Unification of Quantum Algorithms," PRX Quantum, vol. 2, p. 040203, 2021.

[12] D. Poulin and P. Wocjan, "Sampling from the thermal quantum gibbs state and evaluating partition functions with a quantum computer," Phys. Rev. Lett., vol. 103, no. 22, p. 220502, 2009.

[13] C. Zhang, "Randomized algorithms for Hamiltonian simulation," in Monte Carlo and Quasi-Monte Carlo Methods 2010, pp. 709–719, Springer, 2012.

[14] X. Zhang, Y. Shen, J. Zhang, J. Casanova, L. Lamata, E. Solano, M.-H. Yung, J.-N. Zhang, and K. Kim, "Time reversal and charge conjugation in an embedding quantum simulator," Nat. Commun., vol. 6, no. 1, p. 7917, 2015.

[15] Y. Wang, L. Zhang, Z. Yu, and X. Wang, "Quantum Phase Processing and its Applications in Estimating Phase and Entropies," 2022.

[16] T. Kosugi, Y. Nishiya, H. Nishi, and Y.-I. Matsushita, "Imaginary-time evolution using forward and backward real-time evolution with a single ancilla: First-quantized eigensolver algorithm for quantum chemistry," Phys. Rev. Res., vol. 4, no. 3, p. 033121, 2022.

[17] Y. Dong, L. Lin, and Y. Tong, "Ground-state preparation and energy estimation on early fault-tolerant quantum computers via quantum eigenvalue transformation of unitary matrices," PRX Quantum, vol. 3, no. 4, p. 040305, 2022.

[18] D. W. Berry, A. M. Childs, and R. Kothari, "Hamiltonian simulation with nearly optimal dependence on all parameters," in 2015 IEEE 56th annual symposium on foundations of computer science, pp. 792–809, IEEE, 2015.

[19] A. M. Childs, R. Kothari, and R. D. Somma, "Quantum algorithm for systems of linear equations with exponentially improved dependence on precision," SIAM J. Comput., vol. 46, no. 6, pp. 1920–1950, 2017.

[20] T. Odake, H. Kristjánsson, P. Taranto, and M. Murao, "Universal algorithm for transforming hamiltonian eigenvalues," arXiv preprint arXiv:2312.08848, 2023.

[21] Q. Dong, S. Nakayama, A. Soeda, and M. Murao, "Controlled quantum operations and combs, and their applications to universal controllization of divisible unitary operations," 2019.

[22] G. Chiribella, G. M. D'Ariano, and P. Perinotti, "Quantum circuit architecture," Phys. Rev. Lett., vol. 101, no. 6, p. 060401, 2008.

[23] J. Miyazaki, A. Soeda, and M. Murao, "Complex conjugation supermap of unitary quantum maps and its universal implementation protocol," Phys. Rev. Res., vol. 1, no. 1, p. 013007, 2019.

[24] M. T. Quintino, Q. Dong, A. Shimbo, A. Soeda, and M. Murao, "Probabilistic exact universal quantum circuits for transforming unitary operations," Phys. Rev. A, vol. 100, p. 062339, 2019.

[25] M. T. Quintino, Q. Dong, A. Shimbo, A. Soeda, and M. Murao, "Reversing unknown quantum transformations: Universal quantum circuit for inverting general unitary operations," Phys. Rev. Lett., vol. 123, no. 21, p. 210502, 2019.

[26] S. Yoshida, A. Soeda, and M. Murao, "Reversing Unknown Qubit-Unitary Operation, Deterministically and Exactly," Phys. Rev. Lett., vol. 131, p. 120602, Sep 2023.

[27] G. Chiribella and H. Kristjánsson, "Quantum Shannon theory with superpositions of trajectories," Proc. R. Soc. A, vol. 475, no. 2225, p. 20180903, 2019.

[28] G. Chiribella, G. M. D'Ariano, P. Perinotti, and B. Valiron, "Quantum computations without definite causal structure," Phys. Rev. A, vol. 88, p. 022318, Aug 2013.

[29] F. A. Pollock, C. Rodríguez-Rosario, T. Frauenheim, M. Paternostro, and K. Modi, "Non-Markovian quantum processes: Complete framework and efficient characterization," Phys. Rev. A, vol. 97, no. 1, p. 012127, 2018.

[30] O. Oreshkov, F. Costa, and Č. Brukner, "Quantum correlations with no causal order," Nat. Commun., vol. 3, no. 1, p. 1092, 2012.

[31] G. Bai, Y.-D. Wu, Y. Zhu, M. Hayashi, and G. Chiribella, "Efficient algorithms for causal order discovery in quantum networks," 2020.

[32] T. Odake, H. Kristjánsson, A. Soeda, and M. Murao, "Higher-order quantum transformations of Hamiltonian dynamics," 2023.

[33] P. Selinger, "Towards a Quantum Programming Language," Math. Struct. Comput. Sci., vol. 14, no. 4, p. 527, 2004.

[34] P. Selinger and B. Valiron, "A Lambda Calculus for Quantum Computation with Classical Control," in Typed Lambda Calculi and Applications, pp. 354–368, 2005.

[35] E. Campbell, "Random compiler for fast Hamiltonian simulation," Phys. Rev. Lett., vol. 123, no. 7, p. 070503, 2019.