Global sensitivity analysis for optimization of the Trotter-Suzuki decomposition

Alexey N. Pyrkov,[1, *] Yurii Zotov,[2] Jiangyu Cui,[3] and Manhong Yung[4]

[1]*Institute of problems of chemical physics RAS, Acad. Semenov av. 1, Chernogolovka, Moscow region, Russia, 142432*
[2]*Huawei Russian Research Institute, Huawei Technologies Co. Ltd., Moscow, Russia, 121614*
[3]*Central Research Institute, Huawei Technologies, Shenzhen 518129, China*
[4]*Data Center Technology Laboratory, Huawei Technologies Co Ltd, 115371 Shenzhen, Guangdong, China*

The Trotter-Suzuki decomposition is one of the main approaches for realization of quantum simulations on digital quantum computers. Variance-based global sensitivity analysis (the Sobol method) is a wide used method which allows to decompose output variance of mathematical model into fractions allocated to different sources of uncertainty in inputs or sets of inputs of the model. Here we developed a method for application of the global sensitivity analysis to the optimization of Trotter-Suzuki decomposition. We show with a proof-of-concept example that this approach allows to reduce the number of exponentiations in the decomposition and provides a quantitative method for finding and truncation 'unimportant' terms in the system Hamiltonian.

I. INTRODUCTION

Nowadays complex mathematical and computational models are used in all areas of modern society and affected economy and technologies in many different ways [1]. Contemporary mathematical models and development of computational devices have given impulse to fast digitalization of different areas from medicine to agriculture [2–5]. However, growing of data and complexity of systems under consideration does not allow to process all that exactly and the ab-initio understanding of important features of the complex mathematical models looks like an unrealizable dream [6, 7]. Recently, a lot of methods were developed in order to extract the most important features of the complex models in such a manner that it is possible to make predictions about the outcomes on the basis of the chosen features only [8–12]. One of such methods is the variance-based global sensitivity analysis (the Sobol method) [21–24]. It is a wide used method which allows to decompose output variance of mathematical model into fractions allocated to different sources of uncertainty in inputs or sets of inputs of the model. This allows to simplify model under consideration: we can identify the inputs that have no effect on the output and remove redundant parts of the model from consideration.

On the other hand, recently the fast progress in experimental quantum computing and quantum algorithms was observed [25–36] and now many researchers are trying to learn how quantum computing and the Big Data can benefit from each other. Since the idea of quantum computing appeared, one of the most promising applications of quantum computers has been the simulation of complex quantum systems [37–44]. Nowadays quantum simulations has become one of the most important applications for NISQ devices [45–53]. Quantum programmable devices with two-qubit gate fidelities more 98 percents were developed [54–61] and this raises the exciting opportunities for solving problems that are be-

yond the reach of classical computation [45, 62]. However, the error rates of currently available and near-term quantum machines are severely limited by the total number of gates that can be reliably performed to simulate the dynamics of quantum systems. Usually, the total number of gates is defined by the Trotter-Suzuki (TS) decomposition [63–65, 67]. This decomposition allows to represent the evolutionary operator, in the first order expansion, as a product of matrix exponents of the system Hamiltonian terms. Furthermore, TS decomposition is a key routine in some methods of quantum optimal control [13], in time evolution algorithms for matrix product states [14–16] and so on [20]. Thus reaching the goal of practical quantum supremacy will require not only significant experimental advances, but also careful quantum algorithm design and implementation. Recently, some methods for optimization of TS decomposition were proposed. In particular, the schemes based on evolutionary algorithms [17], random sampling [18] and so on [19] were considered. Nevertheless, all the protocols work with some restrictions and have some disadvantages for practical using on current quantum devices (for example, the protocol on the basis of random sampling [18] has advantages only for the huge unpractical number of gates unachievable on current digital quantum machines). On the other hand, one of the simplest approaches that can be used on the current generation of quantum devices and as preprocessing for other more complex protocols is optimization of TS decomposition by removing unimportant gates from consideration with some increasing in error. However, in many cases it is not clear what terms of Hamiltonian can be removed from consideration when we apply the TS decomposition and what criteria we can use for that. Nowadays, the removing of the irrelevant terms is mostly empirical and deals only with the terms which are a few orders of magnitude smaller in terms of some matrix norm in comparison with others.

Here, we develop an approach based on the Global sensitivity analysis for optimization of the TS decomposition to remove the irrelevant terms. It is shown that the approach allows to find most of the terms that give small contribution in the whole problem variance providing a

---
* pyrkov@icp.ac.ru

method for quantitative analysis of the irrelevant terms in the system Hamiltonian. We show that the removing these terms from consideration allows to reduce the number of exponentiations up to 25 percent with moderate increasing in the error of approximation.

II. THE TROTTER-SUZUKI DECOMPOSITION

Simulations of quantum dynamics of complex systems is the classically unsolvable problem due to the fact that the dimension of the problem grows exponentially with the system size and the matrix exponentiation is required. The Trotter-Suzuki decomposition theoretically allows one to model efficiently evolution of complex quantum systems on a digital quantum computer. This decomposition approximates the matrix exponentiation of the sum of non-commuting operators with the product of exponents, each of which can be represented through single-qubit and two-qubit gates, and thus allows to simulate the matrix exponentiation of quantum systems with the desired accuracy.

The simplest first-order Trotter-Suzuki decomposition can be represented as

$$e^{x(A+B)} \approx e^{xA}e^{xB} + O(x^2), \qquad (1)$$

where x - small parameter and A, B - non-commuting operators $[A, B] \neq 0$.

In order to obtain the second–order expansion, it is possible to represent both sides of the equation (1) in the following form:

$$e^{x(A+B)} = I + x(A + B) + \frac{1}{2}x^2(A + B)^2 + O(x^3) =$$

$$I + x(A + B) + \frac{1}{2}x^2\left(A^2 + AB + BA + B^2\right) + O(x^3),$$

$$e^{xA}e^{xB} = \left(I + xA + \frac{1}{2}x^2A^2 + O(x^3)\right) \times$$

$$\left(I + xB + \frac{1}{2}x^2B^2 + O(x^3)\right) =$$

$$I + x(A + B) + \frac{1}{2}x^2\left(A^2 + 2AB + B^2\right) + O(x^3). \quad (2)$$

Thus, we get the following expression

$$e^{xA}e^{xB} = e^{x(A+B)+\frac{1}{2}x^2[A,B]+O(x^3)}, \qquad (3)$$

which corresponds to the well-known Baker-Hausdorff operator identity

$$e^{(A+B)} = e^A e^B e^{-\frac{1}{2}[A,B]}, \qquad (4)$$

when A and B commute with their commutator.

Usually, using the Trotter-Suzuki decomposition, we split x into n parts and in this case the representation
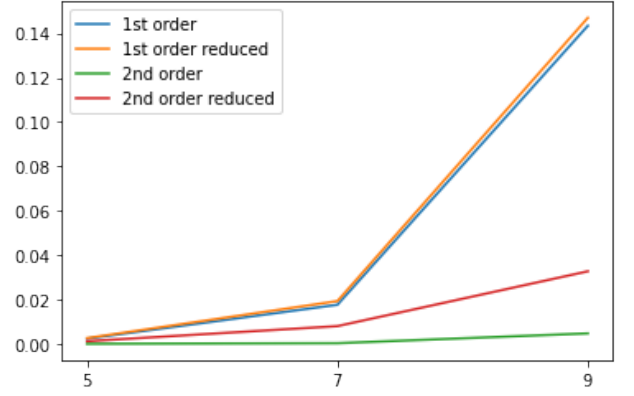

FIG. 1. Error of the TS decomposition versus number of qubits for the dense case. The Hamiltonian has 105 randomly sampled Hermitian terms and about 30 of them are weaken in 50 times. The sensitivity analysis allows to reduce number of exponentiations to 95, 76, 75 terms for 5, 7, 9 qubits respectively with moderate increasing of the error.

can be written as:

$$\left(e^{\frac{x}{n}A}e^{\frac{x}{n}B}\right)^n = \left[e^{\frac{x}{n}(A+B)+\frac{1}{2}\left(\frac{x}{n}\right)^2[A,B]+O\left(\left(\frac{x}{n}\right)^3\right)}\right]^n$$

$$= e^{x(A+B)+\frac{1}{2}\frac{x^2}{n}[A,B]+O\left(\frac{x^3}{n^2}\right)}, \quad (5)$$

that gives the convergence of the formula for $n \to \infty$.

Generalization of the Trotter-Suzuki decomposition to the higher orders reads as

$$e^{x(A+B)} = e^{p_1 xA}e^{p_2 xB}e^{p_3 xA}e^{p_4 xB}\cdots e^{p_M xB} + O(x^{m+1}), \qquad (6)$$

where the selection of the parameters $\{p_1, p_2, \cdots, p_M\}$ is carried out in such a way that the correction term is of the order of x^{m+1}. In this paper we will benchmark only first and second order TS decompositions that are more relevant from practical point of view.

III. VARIANCE-BASED GLOBAL SENSITIVITY ANALYSIS

Modern physical systems can be very complex and when we try to model them the relationships between inputs and outputs are mostly poorly understood. The inputs and outputs can be subject to different sources of uncertainty, including external noise, errors in measurements and incomplete understanding of intrinsic mechanisms that manage the complex system under consideration. This uncertainty imposes a limit on our confidence in the output of the model and requires an knowledge of how much each input is contributing to the output uncertainty. Sensitivity analysis performs the role of ordering the inputs by impact in the variance of the output. In models involving many input variables, sensitivity analysis allows to make the dimension reduction. Here we

overview the theory of variance-based global sensitivity analysis [21–24].

A. Decomposition of variance and Sensitivity indexes

From a black box perspective, any model may be viewed as a function $y = f(x)$ defined in the d-dimensional unit hypercube, where x is a vector with d components $x_1, x_2, \ldots x_d$, and y is a scalar output. In order to intuitively understand how the sensitivity analysis works lets consider what happens with the output y when we fix one of the variables x_i at a particular value $\tilde{x}_i$. Denote the conditional variance of y over all variables but x_i as $Var_{x_{-i}}(y|x_i = \tilde{x}_i)$. It is evident that the resulting conditional variance with one frozen variable is less or equal to the total unconditional variance $Var(y)$. Furthermore, it is easy to notice that the smaller $Var_{x_{-i}}(y|x_i = \tilde{x}_i)$ in comparison with the total variance of the model, the greater the influence of x_i on the model. Thus, the conditional variance $Var_{x_{-i}}(y|x_i = \tilde{x}_i)$ can be used as a measure of the relative importance of x_i. In order to avoid the dependence of the measure on the particular $\tilde{x}_i$ we can average $Var_{x_{-i}}(y|x_i = \tilde{x}_i)$ over all possible points x_i in our hypercube and consider the expectation value $E_{x_i}(Var_{x_{-i}}(y|x_i))$ as the importance measure. This expectation value satisfies the following expression:

$$E_{x_i}(Var_{x_{-i}}(y|x_i)) + Var_{x_i}(E_{x_{-i}}(y|x_i)) = Var(y)$$

and a small $E_{x_i}(Var_{x_{-i}}(y|x_i))$ (or large $Var_{x_i}(E_{x_{-i}}(y|x_i))$) implies that x_i is an important variable in the model. Defining and ordering the sensitivity indexes of the first order as

$$S_i = \frac{Var_{x_i}(E_{x_{-i}}(y|x_i))}{Var(y)}$$

it is possible to estimate how much each input is contributing to the output uncertainty (a high value indicates an important variable).

More rigorously and generally speaking, according to the Sobol's approach the $y = f(x)$ may be decomposed as

$$y = f_0 + \sum_{i=1}^{d} f_i(x_i) + \sum_{i<j}^{d} f_{ij}(x_i, x_j) + \ldots + f_{1,2,\ldots,d}(x_1, \ldots, x_d)$$

$$(7)$$

with the conditions that f_0 is constant and the integrals

$$\int_0^1 f_{i_1,\ldots,i_s}(x_{i_1}, \ldots, x_{i_s}) dx_{i_k} = 0, i_k \in \{i_1, \ldots, i_s\}. \quad (8)$$

From the definition it follows that $f_0 = \int f(x)dx$ and the different terms in the decomposition are pairwise orthogonal due to (8)

and in fact if $\{i_1, \ldots, i_s\} \neq \{j_1, \ldots, j_l\}$ then $\int f_{i_1,\ldots,i_s}(x_{i_1}, \ldots, x_{i_s}) f_{j_1,\ldots,j_l}(x_{j_1}, \ldots, x_{j_l}) dx = 0$.

The process of proving (7) can be sketched as following (see [22] for the details):

- Consider the set of functions $\{g_i, g_{ij}, g_{ijk} \ldots\}$ defined as integrals over all variables but $\{\{x_i\}, \{x_i, x_j\}, \{x_i, x_j, x_k\} \ldots\}$ (for example, $g_i = \int \ldots \int f(x) dx_{-i}$, $g_{i,j} = \int \ldots \int f(x) dx_{-ij}$ and so on, where dx_{-i} denotes integration over all variables except the variable x_i over which the function is not integrated);

- Integrate both sides of (7) over all variables except $\{x_i\}$, $\{x_i, x_j\}$, $\{x_i, x_j, x_k\} \ldots$ and taking into account the condition of (8) we will get

$$g_i = f_0 + f_i(x_i),$$

$$g_{ij} = f_0 + f_i(x_i) + f_j(x_j) + f_{ij}(x_i, x_j), \ldots$$

and thus we can find all the terms in the decomposition of (7).

Without any loss of generality, it is always possible to map the unit hypercube to a desired distribution with the joint probability distribution $p(x)$ and in this case the terms in decomposition can be rewritten as the expectation values in integral form

$$f_0 = E(y) = \int f(x)p(x)dx$$

$$= \int \ldots \int f(x_1, \ldots, x_d)p(x_1) \ldots p(x_d) dx_1 \ldots dx_d,$$

$$f_i(x_i) = E(y|x_i) - f_0 = \int f(x)p(x_{-i})dx_{-i} - f_0,$$

$$f_{ij}(x_i, x_j) = E(y|x_i, x_j) - f_0 - f_i - f_j$$

$$= \int f(x)p(x_{-ij})dx_{-ij} - f_0 - f_i - f_j,$$

$$\ldots$$

$$f_{1,2,\ldots,d}(x_1, \ldots, x_d) = E(y|x_1, \ldots, x_d)$$

$$- (f_0 + \sum_{i=1}^{d} f_i(x_i) + \sum_{i<j}^{d} f_{ij}(x_i) + \ldots), \quad (9)$$

where $p(x_{-i})$ is a product of probability distributions for all variables but x_i.

Assuming that the $f(x)$ is square-integrable, the decomposition of (7) can be squared and integrated and we can rewrite the equation in terms of variances

$$Var(y) = \sum_{i=1}^{d} V_i + \sum_{i<j}^{d} V_{ij} + \ldots + V_{12\ldots d} \quad (10)$$

where

$$Var(y) = \int f^2(x)p(x)dx - f_0^2 = \int (f(x)-E(y))^2 p(x)dx,$$

$$V_i = Var_{x_i}(E_{x_{-i}}(y|x_i)) = \int f_i^2(x_i)dx_i,$$

$$V_{ij} = Var_{x_{ij}}(E_{x_{-ij}}(y|x_i,x_j)) - V_i - V_j$$

$$= \int f_{ij}^2(x_i,x_j)dx_i dx_j, \quad (11)$$

and so on. Here, V_i defines the effect of varying x_i alone (the first-order interaction effect), and V_{ij} is the effect of varying x_i and x_j simultaneously (the second-order interaction effect) and so on (x_{-i} denotes all variables except x_i). This representation shows that the variance of the model output can be decomposed into terms respective to each input and the interactions between them. All the terms sum up to the total variance of the model. Dividing the both parts of equation of (10) by the total variance it is possible to define the sensitivity indexes of higher orders as following

$$S_{\alpha\beta...} = \frac{V_{\alpha\beta...}}{Var(y)} \quad (12)$$

and the equation reads as

$$\sum_{i=1}^{d} S_i + \sum_{i<j}^{d} S_{ij} + \ldots + S_{12...d} = 1 \quad (13)$$

These sensitivity indexes allow to identify the inputs that have no effect on the output and remove redundant parts of the model.

B. Estimators for sensitivity indexes

In the most cases it is not possible to calculate the sensitivity indexes analytically and usually they are estimated with some numerical approximation methods. One of the most common approaches in this case is the Monte Carlo method. The Monte Carlo method involves generating a sequence of randomly distributed points in the input space and calculating some probabilistic characteristics of the process under consideration. In practice, in order to improve efficiency, the quasi-Monte Carlo method is used, in this case the genuine random sample sequence replaced with low-discrepancy sequence. One of the low-discrepancy sequences commonly used in sensitivity analysis is the Sobol sequence. In order to calculate the sensitivity indexes in the Sobol paradigm, the following steps are required: 1) Generate an $N \times 2d$ sample matrix where each row is a sample point with $2d$ dimensions; 2) Denote the first d columns of the matrix as matrix A, and the remaining d columns as matrix B. This effectively can be considered as two independent sets of samples of N points in the d-dimensional unit hypercube;

number of qubits	5	7	9
dense (no. of gates after reduction)	95	76	75
sparse (no. of gates after reduction)	104	95	77

TABLE I. Number of gates after truncation for different number of qubits for dense and sparse cases. Initial randomly generated system Hamiltonians consist of 105 terms for each number of qubits.

3) Construct a set of d matrices with dimension of $N \times d$ with replacing of ith column of matrix A with ith column of matrix B and the remaining columns leave unchanged, denote each such matrix as $A_i(B)$ where $i = 1, 2, ..., d$; 4) The matrices A, B, and $A_i(B)$ define $N \times (d+2)$ points in the input space (one for each row). Using this sample points calculate the corresponding outcome values $f(A)$, $f(B)$ and $f(A_i(B))$; 5) Then in order to calculate the sensitivity indexes it is possible to use the following estimators:

$$Var_{x_i}(E_{x_{-i}}(y|x_i)) \approx \frac{1}{N} \sum_{n=1}^{N} f(B)_n (f(A_i(B))_n - f(A)_n) \quad (14)$$

It is worth to emphasize that the accuracy of the estimators depends on N which can be chosen by sequentially adding points to reach convergence. In order to estimate the sensitivity indexes for all input variables it is required to run calculations $N(d+2)$ times. Since N is often quite large, computational cost can quickly become challenging when the model requires a significant amount of time for a single run. In order to reduce the computational cost some additional numerical techniques were developed, such as emulators, HDMR [68–70] and FAST [71].

IV. OPTIMIZATION OF THE TROTTER-SUZUKI DECOMPOSITION

In this section we present an algorithm for optimization of the TS decomposition by removing unimportant gates from consideration using the global sensitivity analysis with some increasing in error. We apply the algorithm to the first order TS decomposition and show that it allows to find unimportant terms in the system Hamiltonian. We present a proof-of-concept examples with randomly sampled dense and sparse Hermitian matrices as parts of a system Hamiltonian.

In order to construct the proof-of-concept examples, we randomly generate Hamiltonians with some fixed number of random Hermitian matrices, sparse (only 30 percent of each Hamiltonian term are non-zero) and dense:
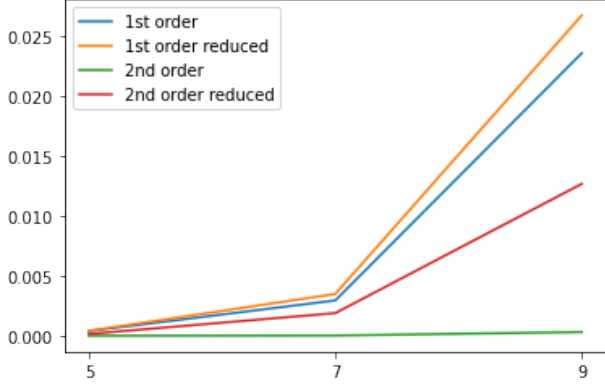
$$H = \sum_n H_n. \quad (15)$$

FIG. 2. Error of the TS decomposition versus number of qubits for the sparse case. The Hamiltonian has 105 randomly sampled Hermitian terms and about 30 of them are weaken in 50 times. The sensitivity analysis allows to reduce number of exponentiations to 104, 95, 77 terms for 5, 7, 9 qubits respectively with moderate increasing of the error.

Our algorithm is based on sensitivity analysis of the Frobenious norm of different linear compositions of the terms in our system Hamiltonian

$$\left\| \sum_n \beta_{in} H_n \right\| \qquad (16)$$

where the coefficients β_{in} constructed as quasi-random Saltelli sequence [24] of the coefficients and i is indexing similar for rows in matrices A, B, and $A_i(B)$ from the previous section. Then we calculate how the coefficients affect the variance of the norm of the system Hamiltonian of (14). In order to take into account contribution of each Hamiltonian term in the total output variance we consider sensitivity indexes of first-order only of (12) and trying to indicate with the indexes what terms of our Hamiltonian can be truncated. We use the SALib python library [72] to implement our calculations of the sensitivity indexes on the HPC [73].

In particular, we consider quantum systems with 5, 7 and 9 qubits. For each of them we construct a Hamiltonian on the basis randomly sampled Hermitian matrices with 105 terms $H = \sum_n H_n$ (separately for dense and sparse cases). Then we artificially weaken about 30 of randomly choosen $\{H_n\}$ in 50 times multiplying with 0.02 and construct a Hamiltonian $\tilde{H} = \sum_n \tilde{H}_n$ with the weaken terms and the rest (the total number of terms in the Hamiltonian $\tilde{H}$ does not change and still equals to 105). In order to calculate error of the TS decomposition in comparison with genuine evolutionary operator with Hamiltonian $\tilde{H}$ we calculate the norm of difference of matrix exponentiations:

$$\epsilon_1 = \left\| e^{-i\tilde{H}t} - \prod_{n=1}^{N} e^{-i\tilde{H}_n t} \right\|, \qquad (17)$$

for the first-order TS approximation and

$$\epsilon_2 = \left\| e^{-i\tilde{H}t} - \prod_{n=1}^{N} e^{\frac{-i\tilde{H}_n t}{2}} \prod_{l=N}^{1} e^{\frac{-i\tilde{H}_l t}{2}} \right\|, \qquad (18)$$

for the second-order TS approximation. In order to calculate the errors we use the HiQPulse python library [74] which allows to calculate the matrix exponentiations more than 5 times faster in comparison with SciPy library.

Then we implement the sensitivity analysis for the Frobenious norm of $\left\| \beta_i \tilde{H} \right\| = \left\| \sum_n \beta_{in} \tilde{H}_n \right\|$ with the use of the Saltelli sequence and find the first-order sensitivity indexes. We remove from consideration the terms with sensitivity indexes which are in 1000 times smaller than the average value of the sensitivity indexes for the dense case and in 5000 times smaller than the average value for the sparse case. After that, we calculate the error of the TS decomposition for the reduced Hamiltonian in comparison with genuine evolutionary operator with Hamiltonian $\tilde{H}$ using the HiQPulse python library. It allows to remove 4, 13 and 25 gates in the first-order TS decomposition for 5, 7 and 9 qubit systems respectively for the dense case (6, 11 and 26 gates for the sparse case) with very small increasing in error (estimated analogously to (17) and (18)), see Table I.

Choosing the same terms for reduction in the second-order TS decomposition we can reduce number of gates from 209 in about 5, 13 and 25 percents for 5, 7 and 9 qubit systems respectively with moderate increasing in error, see Table I.

On the Fig. 1, the results in error between actual system evolution and TS approximations with and without removing unimportant gates in the TS approximations of different orders for the dense case are presented. We can see that the error of the reduced second-order TS approximation is much better in comparison with the first-order TS decomposition with 50 percent more gates for implementation and moderately worse than the genuine second-order TS approximation with 25 percent less number of gates. This opens up some opportunities for more optimal TS approximations with truncation unimportant gates in the NISQ era devices. The similar results are obtained for the sparse case (see Fig. 2).

V. DISCUSSION

Thus we have shown that the Global sensitivity analysis can be used for removing from TS decomposition unimportant gates in quantative way and allows to design quantum circuit in more flexible way. The approach is based on removing the terms that give small contribution in the whole problem variance that can be indicated with the sensitivity indexes. Further development of the method can be around consideration of sensitivity indexes of higher orders that allows to take into account cross interaction in system Hamiltonian.

[1] Digital Economy Report, United Nations, Geneva (2019).
[2] LeCun, Y., Bengio, Y., Hinton, G. Nature 521, 436 (2015)
[3] Ghahramani, Z. Nature 521, 452 (2015)
[4] Esteva, A., Robicquet, A., Ramsundar, B. et al. Nat Med 25, 24 (2019)
[5] A. Carrio, C. Sampedro, A. Rodriguez-Ramos, P. Campoy, Journal of Sensors 2017, 3296874 (2017).
[6] D.l. Donoho, Invited lecture at Mathematical Challenges of the 21st Century, AMS National Meeting, Los Angeles, CA, USA, August 6-12, (2000).
[7] A.N. Gorban, V.A. Makarov, I.Y. Tyukin, Entropy 22, 82 (2020)
[8] H. Samet, Foundations of Multidimensional and Metric Data Structures. Morgan Kaufmann (2006).
[9] D. D. Lee and H. S. Seung, Nature 401, 788 (1999).
[10] S. T. Roweis, L. K. Saul, Science 290, 2323 (2000).
[11] B. Boehmke, B.M. Greenwell, Dimension Reduction Hands-On Machine Learning with R, Chapman and Hall, p. 343 (2019).
[12] A. C. Antoulas, Approximation of Large-Scale Dynamical Systems, SIAM (2005).
[13] Henneke, F., Liebmann, M. A Comput. Visual Sci. 17, 277 (2015)
[14] G. Vidal, Phys. Rev. Lett. 93, 040502 (2004).
[15] S. R. White and A. E. Feiguin, Phys. Rev. Lett. 93, 076401 (2004).
[16] R. Orus and G. Vidal, Phys. Rev. B 78, 155117 (2008)
[17] Benjamin D.M. Jones, George O. O'Brien, David R. White, Earl T. Campbell, John A. Clark, arxiv: 1904.01336 (2019).
[18] E. Campbell, Phys. Rev. Lett 123,070503 (2019).
[19] T. Barthel, Y. Zhang, Annals of Physics, 418, 168165 (2020)
[20] J. Kolorenc and L. Mitas, Rep. Prog. Phys. 74, 026502 (2011)
[21] I. M. Sobol, Math. Comput. Simulat. 55, 271 (2001).
[22] I.M. Sobol, Mathematical Modeling and Computational Experiment, 1, 407 (1993).
[23] A. Saltelli, P. Annoni, I. Azzini, F. Campolongo, M. Ratto, and S. Tarantola, Computer Physics Communications, 181, 259 (2010).
[24] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, Global Sensitivity Analysis. The Primer, John Wiley and Sons (2008).
[25] J. Biamonte, et al., Nature 549, 195 (2017).
[26] Carleo, G. et al., Reviews of Modern Physics 91, 045002 (2019).
[27] Jeswal, S. K., Chakraverty, S. Archives of Computational Methods in Engineering 26, 793 (2019).
[28] Broughton, M. et al., arXiv: 2003.02989 (2020).
[29] I.D. Lazarev, et. al., arxiv: 2009.09246 (2020).
[30] Cong, I., Choi, S., Lukin, M. D., Nature Physics 15, 1273 (2019).
[31] IBM Quantum Experience. URL https://www.ibm.com/quantum-computing/
[32] Lu, S., Duan, L.-M., Deng, D.-L., arXiv: 2001.00030 (2019).
[33] Nautrup, H. P., Delfosse, N., Dunjko, V., Briegel, H. J. and Friis, N., Quantum 3, 215 (2019).
[34] Jun Gao, et.al., Phys. Rev. Lett. 120, 240501 (2018).
[35] A.A. Melnikov, et. al., PNAS 115, 1221 (2018)
[36] Peruzzo, A., McClean, J., Shadbolt, P. et al., Nat Commun 5, 4213 (2014).
[37] R. P. Feynman, Inter. J. Th. Phys. 21, 467 (1982).
[38] S. Lloyd, Science 273, 1073 (1996).
[39] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, Comm. Math. Phys. 270, 359 (2007)
[40] P. J. J. O'Malley, et.al., Phys. Rev. X 6, 031007 (2016).
[41] A. Aspuru-Guzik, et.al., Science 309, 1704 (2005).
[42] I. Kassal, J.D. Whitfield, A. Perdomo-Ortiz, M. Yung, and A. Aspuru-Guzik, Annual Review of Physical Chemistry 62, 185 (2011).
[43] N. Cody Jones, et. al., New J. Phys. 14 115023 (2012).
[44] R. Babbush, et.al., Phys. Rev. X 8, 011044 (2018).
[45] J. Preskill, Quantum 2 79 (2018)
[46] J. Zhang, et.al, Nature 551, 601 (2017).
[47] N. Freitas, et.al., Int. J. Quantum Inf. 16, 1840008 (2018)
[48] D. Poulin, et.al., Phys. Rev. Lett. 121, 010501 (2018)
[49] I.D. Kivlichan, et.al., arXiv:1902.10673
[50] B. P. Lanyon, et. al. , Science 334, 57 (2011).
[51] N. K. Langford, et.al., Nat. Commun. 8, 1715 (2017)
[52] A. M. Childs, et.al., PNAS, 115, 9456 (2018).
[53] C. Gross, and I. Bloch, Nature 357 , 995 (2017)
[54] Zajac, D. M. et al., Science 359, 439 (2018)
[55] Huang, W. et al., Nature 569, 532 (2019)
[56] Barends, R. et al., Nature 508, 500 (2014)
[57] Wright, K. et al., Nat. Commun. 10, 1 (2019)
[58] Gaebler, J. P. et al., Phys. Rev. Lett. 117, 060505 (2016)
[59] Ballance, C., Harty, T., Linke, N., Sepiol, M. and Lucas, D., Phys. Rev. Lett. 117, 060504 (2016)
[60] Rol, M. et al., Phys. Rev. Lett. 123, 120502 (2019)
[61] Kjaergaard, M. et al., Annu. Rev. Condens. Matter Phys. 11, 369 (2019)
[62] Arute, F., Arya, K., Babbush, R. et al., Nature 574, 505 (2019).
[63] H. F. Trotter, Proceedings of the American Mathematical Society 10, 545 (1959)
[64] M. Suzuki, J. Math. Phys. 32, 400 (1991)
[65] R. D. Ruth, IEEE Trans. Nucl. Sci , 2669 (1983).
[66] W. Janke and T. Sauer, Physics Letters A 165, 199 (1992).
[67] A.M. Childs et.al., arxiv: 1912.08854 (2020).
[68] H. Rabitz, Science 246, 221 (1989).
[69] G. Li et. al., Journal of Physical Chemistry A 110, 2474 (2006).
[70] G. Li, Journal of Physical Chemistry A 106, 8721 (2002).
[71] R. I. Cukier, C. M. Fortuin, and K. E. Shuler, J. Chem. Phys. 59, 3873 (1973)
[72] Sensitivity Analysis Library (SALib), https://github.com/SALib/SALib
[73] 4 CPU, 12 core Intel(R) Xeon(R) CPU E7-4850 v2 @ 2.30GHz, 512GB RAM.
[74] Will be available on the GitHub later.