# Optimization for quantum computer simulation

Naoki Yoshioka
RIKEN Center for Computational Science
Kobe, Hyogo, Japan
naoki.yoshioka@riken.jp

Hajime Inaoka
RIKEN Center for Computational Science
Kobe, Hyogo, Japan
hajime.inaoka@riken.jp

Nobuyasu Ito
RIKEN Center for Computational Science
Kobe, Hyogo, Japan
nobuyasu.ito@riken.jp

Fengping jin
Institute for Advanced Simulation, Jülich Supercomputing Centre
Forschungszentrum Jülich
Jülich, Germany
f.jin@fz-juelich.de

Kristel Michielsen
RWTH Aachen University
Aachen, Germany
Institute for Advanced Simulation, Jülich Supercomputing Centre
Forschungszentrum Jülich
Jülich, Germany
k.michielsen@fz-juelich.de

Hans De Raedt
Zernike Institute for Advanced Materials
University of Groningen
Groningen, the Netherlands
f.jin@fz-juelich.de

**ACM Reference Format:**
Naoki Yoshioka, Hajime Inaoka, Nobuyasu Ito, Fengping jin, Kristel Michielsen, and Hans De Raedt. 2019. Optimization for quantum computer simulation. In *Proceedings of ACM Conference (Conference'17).* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/nnnnnnn.nnnnnnn

Study of quantum computation is very active especially in this decade. Quantum annealing [1] is now implemented and commercially available [2]. Meanwhile quantum logic gates or quantum circuits [3, 4] are also experimentally studied very much (e.g., [5]). Simulations of quantum circuits using classical computers are useful to study quantum information processing and quantum algorithms because it is still limited to obtain large resources of real quantum circuit computers. It is important to use supercomputers to simulate quantum circuits because it is required to calculate quantum systems exactly and then exponentially large memory is needed for such calculations. One of the early studies to simulate quantum circuits in classical supercomputers is done by De Raedt and his coworkers in 2007 [6]. At that time it was succeeded to simulate up to 36 qubits. The record of the number of qubits in quantum computer simulations is recently updated and now systems with 45 qubits can be simulated in some highest-performing supercomputers [7, 8]. Moreover, instead of reduced numerical precision, it is possible to simulate quantum circuits with 48 qubits [7].

We develop a software and a C++ template library [9] to simulate quantum circuits on classical computers, ranging from laptop computers to massively parallel computers. Although the method of implementation of quantum computer simulator is basically similar to the previous one [6], here we propose optimization techniques to speed up the simulations; a page method and a simple rule of initial permutation of qubits. We report performance of our simulations run on the K computer in RIKEN R-CCS up to 45 qubits, which we need 0.5 petabytes of memory if we use double precision floating-point number as a real number. Note that we do not try to perform the simulations with 48 qubits because such a technique with reduced precision is out of scope of this work.

We accept the same algorithm proposed by De Raedt and his coworkers [6] for MPI parallelization. In this algorithm we have to transfer half of total memory size for each gate operation if needed.

Let us divide $N$ bits to $M$ global bits and $L = N - M$ local bits such as $n = n_N \cdots n_{N-M+1} | n_L \cdots n_1$. Global bits represent MPI rank $r = n_N \cdots n_{N-M+1}$, and local bits $n_L \cdots n_1$ correspond to memory address in each computational node.

As an example, let us consider the case we have only 1 global qubits (2 MPI processes) and a gate operates on the global qubit. it is impossible to calculate results directly because $a(0 \,|\, n_L \cdots)$ and $a(1 \,|\, n_L \cdots)$ do not exist in the same MPI process, where $a(\cdots)$ is a complex coefficient of a state vector $|\Phi\rangle = \sum_{n=0}^{2^N-1} a(n) |n\rangle$. Instead we have to swap data between MPI processes with ranks 0 and 1 and permutate the qubits such as

$$a(0 \,|\, 1\, n_{L-1} \cdots) \leftrightarrow a(1 \,|\, 0\, n_{L-1} \cdots) \tag{1}$$

by using `MPI_Sendrecv`. We can calculate results in parallel in each MPI process after this data transfer.

In this study we examine two quantum circuits; the Hadamard gates and quantum adders. The Hadamard gates circuit is very simple. As shown in Fig. 1, the Hadamard gate operates on each qubit in the ascending order. Meanwhile, Fig. 2 shows a circuit of quantum addition of two registers with three qubits, which is proposed by Draper [10]. Quantum adder involves $(N/2)!$ controlled phase-shift gates and the (inversely ordered) quantum Fourier transforms $\tilde{\mathcal{F}}_N$ [4]. Figure 3 (a) and (b) show epalsed time of simulation results of the Hadamard gates and the quantum adder, respectively.

The *page method* is optimization to decrease intranode data transfer. Transfer speed of memory in the K computer is 64 GB/sec. We transfer 4 GB data for each operation in the case of a one-qubit gate while we do totally 6 GB data for a two-qubit gate. Therefore it ideally takes about 0.06 seconds and 0.09 seconds for a one-qubit


*Conference'17, July 2017, Washington, DC, USA*

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

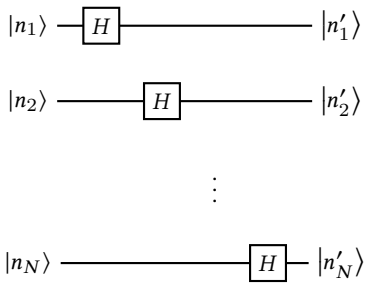


**Figure 1: Quantum circuit of the Hadamard gates. A gate $H$ is the Hadamard gate. Only one Hadamard gate is operated for each qubit.**

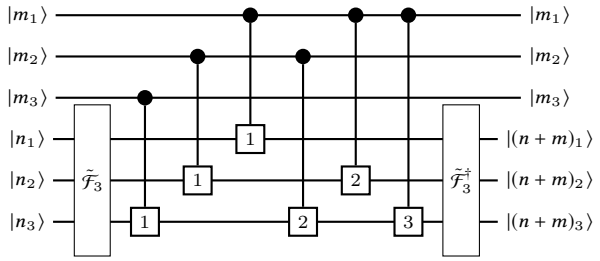


**Figure 2: Quantum circuit of "quantum adder" [10]. This example shows addition of two registers with values $n$ and $m$ and substitution to the register for $n$, and each register contains three qubits. Note that a gate with $k$ is a controlled phase-shift gate $R(\phi)$ with $\phi = 2\pi/2^k$. Moreover a gate with $\tilde{\mathcal{F}}_N$ is the inversely ordered quantum Fourier transform [4].**

gate and a two-qubit gate, respectively. On the other hand, for each operation it totally takes 0.3–4.3 seconds. This means costs of copy cannot be ignored in some cases.

In the page method, allocated memory is divided into $4 + 1$ blocks, which we name *pages*. One of five pages is prepared for a buffer of MPI communications, and other four pages are used for calculations. The fact we have four pages for data means the two most significant local qubits are *page qubits*. The page qubits are local qubits to which global qubits would be swapped. First we transfer data to the other process using `MPI_Sendrecv` as usual. Then, instead of copying data in a buffer to a page, we change labels of pages: A page that was a buffer page becomes a page with data, and vice versa.

We have to transfer 4 GB or 6 GB data for each gate operation if we apply a gate to global qubits. Therefore it is very important to decrease the number of data transfers for performances of our simulations, and we have to consider the initial permutation of qubits. For example, in the case of the quantum adder with two 3-qubit registers shown in Fig. 2, only one gate operates on $|m_3\rangle$ while nine gates operate on $|n_3\rangle$. Therefore it is intuitively supposed that we would have many data transfers if we initially set $|n_3\rangle$ as the most significant qubit.

Because it is not easy to find the best case generally, here we propose a simple rule of initial permutation: Sort by ascending order

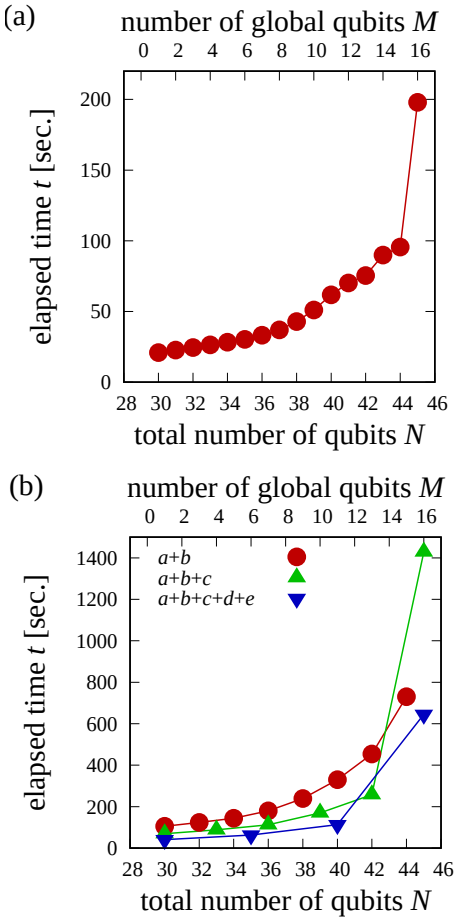


**Figure 3: (a) The elapsed time of simulation of the Hadamard gates (see Fig. 1) as a function of the total number of qubits $N$. Note that the corresponding number of global qubits $M$ are also shown. (b) The elapsed time of simulation of the quantum adder (see Fig. 2) as a function of the total number of qubits $N$. Note that $a_1 + \cdots + a_l$ in the legend means simulation results of additions of $l$ registers with the same size $N/l$.**

of the number of gates operating on each qubit. For example, in the case of Fig. 2, the initial permutation is given by $m_3 m_2 m_1 n_1 n_2 n_3$.

## REFERENCES

[1] Tadashi Kadowaki and Hidetoshi Nishimori. Quantum annealing in the transverse ising model. *Phys. Rev. E*, 58:5355–5363, Nov 1998.

[2] M. W. Johnson, M. H. S. Amin, S. Gildert, T. Lanting, F. Hamze, N. Dickson, R. Harris, A. J. Berkley, J. Johansson, P. Bunyk, E. M. Chapple, C. Enderud, J. P. Hilton, K. Karimi, E. Ladizinsky, N. Ladizinsky, T. Oh, I. Perminov, C. Rich, M. C. Thom, E. Tolkacheva, C. J. S. Truncik, S. Uchaikin, J. Wang, B. Wilson, and G. Rose. Quantum annealing with manufactured spins. *Nature*, 473:194, 2011.

[3] David Deutsch. Quantum theory, the church–turing principle and the universal quantum computer. *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, 400(1818):97–117, 1985.

[4] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, Cambridge, 2000.

[5] J. Kelly, R. Barends, A. G. Fowler, A. Megrant, E. Jeffrey, T. C. White, D. Sank, J. Y. Mutus, B. Campbell, Yu Chen, Z. Chen, A. Chiaro, B. Dunsworth, I.-C. Hoi,

C. Neill, P. J. J. O'Malley, C. Quintana, P. Roushan, A. Vainsencher, J. Wenner, A. N. Cleland, and John M. Martinis. State preservation by repetitive error detection in a superconducting quantum circuit. *Nature*, 519:66, 2015.

[6] Koen De Raedt, Kristel Michielsen, Hans De Raedt, B. Trieu, G. Arnold, M. Richter, Th. Lippert, Hiroshi Watanabe, and Nobuyasu Ito. Massively parallel quantum computer simulator. *Computer Physics Communications*, 176(2):121 – 136, 2007.

[7] Hans De Raedt, Fengping Jin, Dennis Willsch, Madita Willsch, Naoki Yoshioka, Nobuyasu Ito, Shengjun Yuan, and Kristel Michielsen. Massively parallel quantum computer simulator, eleven years later. *Computer Physics Communications*, 237:47 – 61, 2019.

[8] Thomas Häner and Damian S. Steiger. 0.5 petabyte simulation of a 45-qubit quantum circuit. In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis*, SC '17, pages 33:1–33:10, New York, NY, USA, 2017. ACM.

[9] https://github.com/naoki-yoshioka/braket.

[10] Thomas G. Draper. Addition on a Quantum Computer. *eprint arXiv:quant-ph/0008033*, August 2000.