Storing the quantum Fourier operator in the QuIDD data structure

Katalin Friedl, László Kabódi

One of the most known results of quantum computing is Shor's algorithm [2]. With a quantum computer running Shor's algorithm one can factor a composite number in expected polynomial time. But there is no known working quantum computer with more than a few qubits, so the algorithm is only a theoretical breakthrough.

Quantum algorithms can be simulated using classical computers, but the time complexity of the simulation is exponential. There are some data structures which can accelerate this. Viamontes et. al. described the QuIDD data structure [3] that is an extension of algebraic decision diagram, and wrote a quantum circuit simulator employing it, the QuIDDPro [4]. Using the QuIDD software one can simulate various quantum algorithms, for example Grover's famous search algorithm [1] in linear time. But the quantum operators used by Grover's algorithm differ from the ones used by Shor's factorization algorithm. The main difficulty in simulating Shor's algorithm is the quantum Fourier operator.

In this paper we examine the matrix of the Fourier operator, and it's QuIDD representation. In it's usual reresentation the decision diagram of an n qubit Fourier operator consists of approximately 2^{2n} nodes. There is an internal structure of the matrix, which we try to extract. First, we examine changing the ordering of the underlying decision diagram. This way we can use the recursive nature of the Fourier operator to achieve a diagram approximately $\frac{2}{3}$ times the size of the standard ordering. Then we try to emulate this change using permutation matrices, so the existing softwares can be used with greater efficiency. There are two permutation matrices we examine. One emulates the reverse ordering of the column variables, the other separates the columns based on their parity. Using the first one, we emulate the reversed ordering exactly, but the QuIDD representation of the permutation matrix is exponentially large. The second one exposes an internal symmetry, which can produce a diagram approximately $\frac{3}{4}$ times the original, but we can store the permutation matrix in linear space. Finally we propose a new method of storing the Fourier operator, using multiplyers stored on the edges of the decision diagram. This helps reducing the size of the operator, but we do not know how the operations can be efficiently transfered to this variant.

Acknowledgements

This research is partially supported by the National Research, Development and Innovation Office (NKFIH), by the OTKA-108947 grant.

References

[1] Lov K. Grover, A fast quantum mechanical algorithm for database search, Proceedings of the Twenty-eighth Annual ACM Symposium on Theory of Computing, 1996. http://arxiv.org/pdf/quant-ph/9605043

[2] Peter W. Shor, Algorithms for Quantum Computation: Discrete Logarithms and Factoring, Proceedings of the 35th Annual Symposium on Foundations of Computer Science, 1994.

[3] George F. Viamontes, Igor L. Markov, John P. Hayes, Improving Gate-Level Simulation of Quantum Circuits, Quantum Information Processing 2, 2003.

[4] George F. Viamontes, Igor L. Markov, John P. Hayes, QuIDDPro: High-Performance Quantum Circuit Simulation, http://vlsicad.eecs.umich.edu/Quantum/qp/