ISSN 2229-5518

Quantum algorithm for spectral diffraction of probability distributions

Nikolay Raychev

Abstract: This article is focused on the creation of own algorithm for simulation of spectral diffraction of probabilistic distributions (SDPD) in the developed for the needs of this research interactive environment for simulation of quantum algorithms. The quantum spectral diffraction of probabilistic distributions is algorithm for transformation of probabilistic distributions, designed for implementation in quantum computers. The proposed method is original and new, so not to be compared with other similar methods.

Key words: Quantum computing, diffraction, simulator, operators, gates

1. INTRODUCTION

This research presents original algorithm for approximation of quantum spectral diffraction of periodical probabilistic distributions over random Z_p basis. Application of this quantum algorithm can be many to one within each period. The initial state of the quantum calculation may have different effects on the evolution. Specifically the quantum process may evolve asymmetrically with trend to the initial state. This effect comes as a result of local properties of convertibility in the quantum systems. The correction of this effect is done through initialization of symmetric superposition. If the possible states are $|1\rangle, |2\rangle \ldots|j\rangle \ldots|d\rangle$, after the application of the function for spectral diffraction we obtain symmetrical superposition of periodically probable distributions, which is applicable as a single operator of size D

$$C^d_{\mathrm{DFT}} = \begin{pmatrix} 1 & 1 & 1 & \dots & 1 \\ 1 & w & w^2 & & w^{d-1} \\ 1 & w^2 & \ddots & & w^{2(d-1)} \\ \vdots & & & & \vdots \\ 0 & w^{d-1} & w^{2(d-1)} & \dots & w^{(d-1)(d-1)} \end{pmatrix}$$

SDPD reduces the computation time with several orders of magnitude. This could lead to the beginning of an entire revolution in probabilistic distribution processing and resulted in an abundance of fast algorithms for diffraction computation in a variety of situations. The property that allowed the creation of these fast algorithms is that, as it turns out, most diffraction formulas contain at their core one or more Fourier transforms which may be rapidly calculated using the SDPD. The key in discovering a new fast algorithm is to reformulate the diffraction formula so that to identify and isolate the Fourier transforms it contains. In this way, the fast scaled transformation were designed.

Remarkable improvements were the generalization of the QFT to scaled SDPD which allowed freedom to choose the dimensions of the output window for the SDPD probabilistic distribution diffraction, the mathematical concept of linearized convolution which thwarts the circular character of the quantum Fourier transform and allows the use of the SDPD, and last but not least the linearized discrete scaled convolution, a new concept of which we claim priority.

2. ALGORITHM FOR SIMULATION OF SPECTRAL DIFFRACTION OF PROBABILISTIC DISTRIBUTIONS

The quantum spectral diffraction of probabilistic distributions (SDPD) is an algorithm for transformation of periodical probabilistic distributions, designed for implementation in quantum computers and it is exponentially faster compared to the classic analogue computer for the cases when it is applicable. It suffers some serious limitations: the preparation of the input states may not be possible or may not be subjective to processing.

If we assume that it is possible to prepare applicable input state and processing delivering expected answer at the end, then the SDPD circuits have size $O(n \log(n))$ with upper bound of $O(n(\log n)^2)$ operations for quantum transformations of periodical probabilistic distributions in superposition.

a. CREATION OF SDPD CHAIN

The creation of own algorithm for classic simulation of SDPD in the developed for the needs of this research quantum simulator is a very complicated task. The puzzle contains eight inputs for testing the chain. Every testing input works at different frequency with its own phase superposition from i8 turns of the state. Every testing input is associated with the desired output: simple state i, index of the input frequency. The aim is to find some sequence of gates which provides desired results from the testing inputs. The first step in solving the puzzle is search of dependencies in the input models.

ISSN 2229-5518

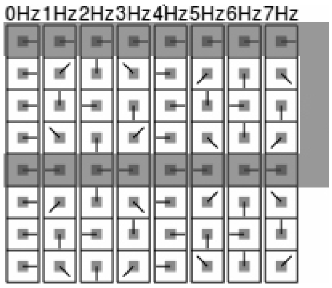

Fig. 1 Similar lines

The experiments with the simulator showed that after 4 turns of i8, the obtained state is always the same and exactly the opposite to the initial one. This is very useful in the case, because the Hadamart gate demonstrates exactly the same conformity. Placing the H gate on line/bit 2 (because the $2^2 = 4$ and the scheme is divisible to 4) simplifies the scheme:

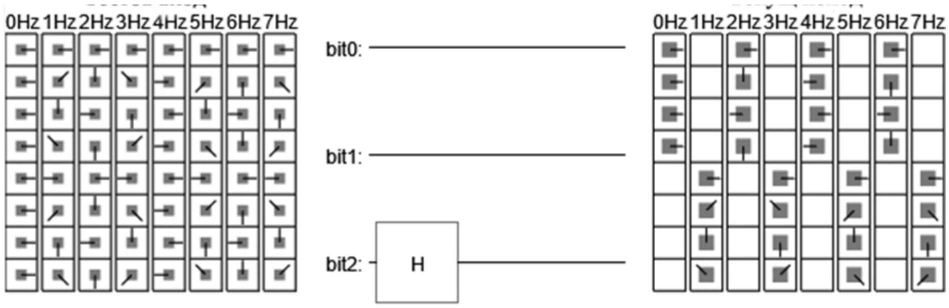

Fig. 2 Placing the Hadamard gate

The above scheme shows that placing of Hadamard gate on the second is a step in the right direction. The aim is that every input frequency ends with one and the same output, but after applying the Hadamard gate the outputs are already in four possible states instead of eight. In the cause of the experiments with the quantum simulator an attempt was made to apply Hadamard gate over the first line/bit, but this did not lead to improvement of the results:

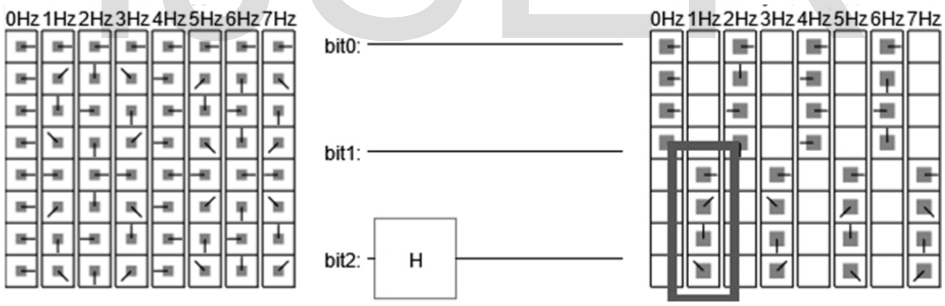

Fig. 3 SDPD over three bits

Despite of the fact that the upper four rows have reverse or equal state to the initial one (but divided to 2 instead of 4), the lower four rows does not correspond to this condition. The lowest two rows are with shift of 90^0. Shift of 6 and 7 is observed on the lower two rows, and this is expressed as 110 and 111 in binary form. These are the states where bit 1 and bit 2 are ON. The aim in this case is to rotate the phases of these states with 90^0. R(-90^0) does approximately what is aimed, but it is dependable on one single bit only. To make the gate dependable on two bits, another bit is placed on the line as a control one for the operation. With the 90^0 correction on the spot, the placing of the Hadamard gate on bit 1 will filter one more part of the unnecessary states:

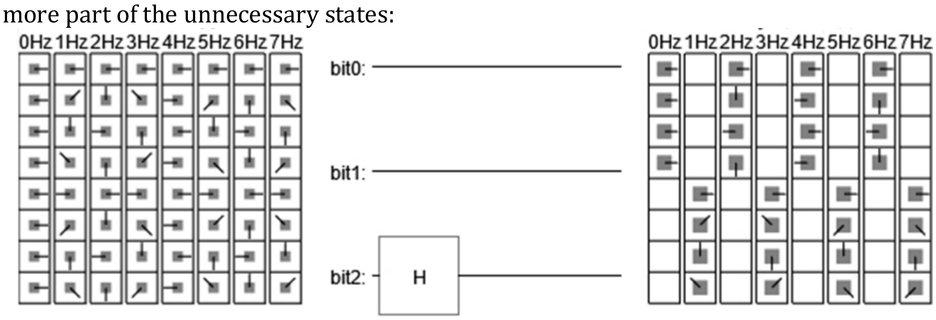

Fig. 4 SDPD over three bits

During the next stage of the experimental research de-phasing gates were added to guarantee that the rest two non-empty states in every column have equal or opposite amplitudes. To achieve this, the Hadamard gate was applied again. As a result every input frequency appears in one unique output state.

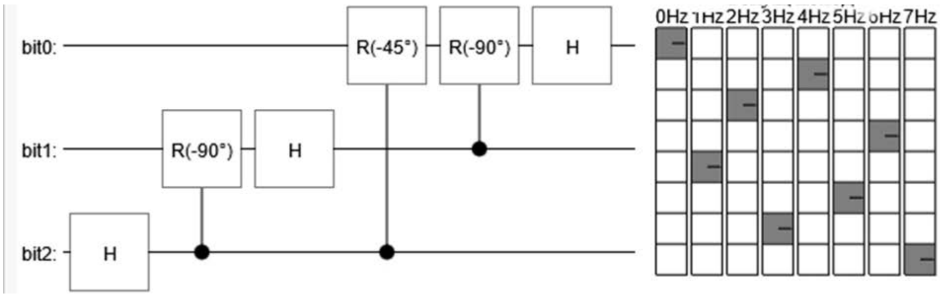

Fig. 5 SDPD over three bits

At this stage the algorithm looks like a cycle.
The problem is that with SDPD a tendency for reversing of the bits is observed. Despite that we may ignore this fact for the needs of this experiment, it is anyway useful to understand how to correct this. Everything which should be done is to interchange bits 0 and 2 to get the results in the correct order.

The obtained chain is SDPD over three bits:

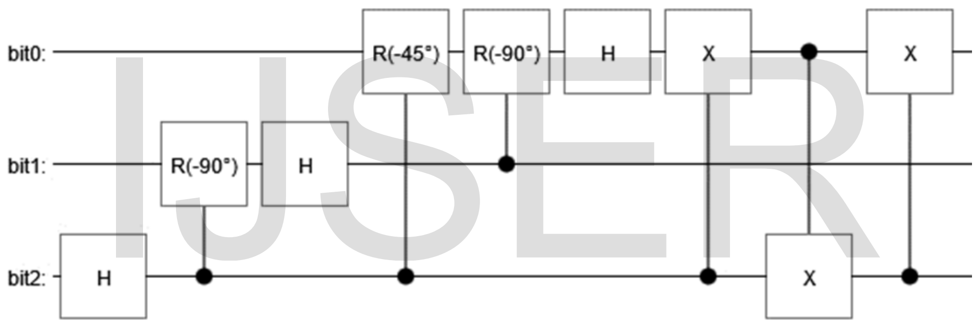

Fig. 6 SDPD over three bits

This is not the only solution of course.
Another question arises and this is if the so presented SDPD scheme will work correctly with combination of frequencies, because only single frequencies were tested during this experiment. Because the quantum chains always apply linear operations, this means that scaling of the input will scale the output, i.e. adding input will add output.

3. CONCLUSIONS AND FUTURE WORK

It appears at the end that the simulation of SDPD is implemented through a sequence of Hadmard gates with some phase corrections during the execution time and partial inverting if necessary. For the supposed SDPD algorithm which takes $|k\rangle \rightarrow |k+1 \bmod N\rangle$ eigenvalues are $e^{2\pi ij/N}$ for integer j. The SDPD algorithm can be used to approximately find j/N, and then use continued fractions to find N exactly. This functionality enables factorization and finding the discrete logs. The supposed SDPD algorithm can be a part of many quantum algorithms, for factoring and computing the discrete logarithm, the quantum phase estimation algorithm for estimating the eigenvalues of a unitary operator, and algorithms for the hidden subgroup problem.

REFFERENCES

[1] M. A. Nielsen and I. L. Chuang, Quantum Computation and Quantum Information (Cambridge University Press, Cambridge, United Kingdom, 2000).
[2] I. L. Chuang, N. Gershenfeld and M. Kubinec, "Experimental Implementation of Fast Quantum Searching", Physical Review Letters, 80(15), 3408-3411, 1998.
[3] R. Cleve and J. Watrous, "Fast Parallel Circuits for the Quantum Fourier Transform", Proceedings of IEEE Symposium on the Theory of Computing, pp. 526-535, 2000.
[4] W. Cooley and J. Tukey, "An Algorithm for the Machine Calculation of Complex Fourier Series", Math. Of Computation, 19:297-301, 1965.

[5] D. Coppersmith, "An Approximate Fourier Transform Useful in Quantum Factoring", IBM Research Report RC 19642, 1994.
[6] X. Deng, T. Hanyu and M. Kameyama, "Quantum Device Model Based Super Pass Gate for Multiple-Valued Digital Systems", Proceedings of Intl. Symp. Multiple-Valued Logic, pp. 130-138, 1995.
[7] D. Deutsch, "Quantum Theory, the Church-Turing Principle and the Universal Quantum Computer", Proc. Royal Soc. A 400:97-117, 1985.
[8] N. Gershenfeld and I. L. Chuang, "Bulk Spin-Resonance Quantum Computing", Nature, vol. 404, pp.350-356, 1997.
[9] R. Duan, Z. Ji, Y. Feng and M. Ying, "A Relation Between Quantum Operations and the Quantum Fourier Transform", Quantum Physics Archive, arxiv: quant-ph/0304145
[10] L. Hales, "Quantum Fourier Transform and Extensions of the Abelian Hidden Subgroup Problem", Ph. D. Dissertation, UC Berkeley, 2002.
[11] S. L. Hurst, D. M. Miller and J. Muzio, "Spectral Techniques in Digital Logic", Academic Press, London, 1985.
[12] R. Jozsa, "Quantum Algorithms and the Fourier Transform", Proceedings of Royal Society of London, 454:323-337, 1997.
[13] Nikolay Raychev. Dynamic simulation of quantum stochastic walk. In International jubilee congress (TU), 2012.
[14] Nikolay Raychev. Classical simulation of quantum algorithms. In International jubilee congress (TU), 2012.
[15] Nikolay Raychev. Interactive environment for implementation and simulation of quantum algorithms. CompSysTech'15, DOI: 10.13140/RG.2.1.2984.3362, 2015
[16] Nikolay Raychev. Unitary combinations of formalized classes in qubit space. International Journal of Scientific and Engineering Research 04/2015; 6(4):395-398. DOI: 10.14299/ijser.2015.04.003, 2015.
[17] Nikolay Raychev. Functional composition of quantum functions. International Journal of Scientific and Engineering Research 04/2015; 6(4):413-415. DOI:10.14299/ijser.2015.04.004, 2015.
[18] Nikolay Raychev. Logical sets of quantum operators. International Journal of Scientific and Engineering Research 04/2015; 6(4):391-394. DOI:10.14299/ijser.2015.04.002, 2015.
[19] Nikolay Raychev. Controlled formalized operators. In International Journal of Scientific and Engineering Research 05/2015; 6(5):1467-1469, 2015.
[20] Nikolay Raychev. Controlled formalized operators with multiple control bits. In International Journal of Scientific and Engineering Research 05/2015; 6(5):1470-1473, 2015.
[21] Nikolay Raychev. Connecting sets of formalized operators. In International Journal of Scientific and Engineering Research 05/2015; 6(5):1474-1476, 2015.
[22] Nikolay Raychev. Indexed formalized operators for n-bit circuits. International Journal of Scientific and Engineering Research 05/2015; 6(5):1477-1480, 2015.
[23] Nikolay Raychev. Converting the transitions between quantum gates into rotations. International Journal of Scientific and Engineering Research 06/2015; 6(6): 1352-1354. DOI:10.14299/ijser.2015.06.001, 2015.
[24] Nikolay Raychev. Quantum algorithm for non-local coordination. International Journal of Scientific and Engineering Research 06/2015; 6(6):1360-1364. DOI:10.14299/ijser.2015.06.003, 2015.
[25] Nikolay Raychev. Universal quantum operators. International Journal of Scientific and Engineering Research 06/2015; 6(6):1369-1371. DOI:10.14299/ijser.2015.06.005, 2015.
[26] Nikolay Raychev. Ensuring a spare quantum traffic. International Journal of Scientific and Engineering Research 06/2015; 6(6):1355-1359. DOI:10.14299/ijser.2015.06.002, 2015.
[27] Nikolay Raychev. Quantum circuit for spatial optimization. International Journal of Scientific and Engineering Research 06/2015; 6(6):1365-1368. DOI:10.14299/ijser.2015.06.004, 2015.
[28] Nikolay Raychev. Conflicts resolving by symmetry breaking. International Journal of Scientific and Engineering Research 05/2015; 6(5): 1481-1482. DOI:10.14299/ijser.2015.05.005, 2015.