engineering proceedings

MDPI

Proceeding Paper

Quantum Arithmetic for Directly Embedded Arrays [†]

Alberto Manzano [1,*,‡], Daniele Musso [2,‡], Álvaro Leitao [1], Andrés Gómez [2], Carlos Vázquez [1], Gustavo Ordóñez [3] and María Rodríguez-Nogueiras [3]

[1] Research Group M2NICA, Department of Mathematics, CITIC, University of A Coruña, Campus de Elviña S/N, 15071 A Coruña, Spain; alvaro.leitao@udc.gal (Á.L.); carlos.vazquez.cendon@udc.es (C.V.)
[2] Centro de Supercomputación de Galicia (CESGA), Avenida de Vigo, 15705 Santiago de Compostela, Spain; daniele.musso@cesga.es (D.M.); andres.gomez.tato@cesga.es (A.G.)
[3] Global Risk Analytics, HSBC, Centenary Sq, Birmingham B1 1HQ, UK; gustavo.ordonez-sanz@hsbc.com (G.O.); maria.r.nogueiras@hsbc.com (M.R.-N.)
[*] Correspondence: alberto.manzano.herrero@udc.es
[†] Presented at the 4th XoveTIC Conference, A Coruña, Spain, 7–8 October 2021.
[‡] These authors contributed equally to this work.

Abstract: We describe a general-purpose framework to implement quantum algorithms relying upon an efficient handling of arrays. The cornerstone of the framework is the direct embedding of information into quantum amplitudes, thus avoiding hampering square roots. We discuss the entire pipeline, from data loading to information extraction. Particular attention is devoted to the definition of an efficient toolkit of basic quantum operations on arrays. We comment on strong and weak points of the proposed quantum manipulations, especially in relation to an effective exploitation of quantum parallelism. We describe in detail some general-purpose routines as well as their embedding in full algorithms. Their efficiency is critically discussed both locally, at the level of the routine, and globally, at the level of the full algorithm. Finally, we comment on some applications in the quantitative finance domain.

Keywords: quantum computing; quantum simulation

check for updates

Citation: Manzano, A.; Musso, D.; Leitao, Á.; Gómez, A.; Vázquez, C.; Ordóñez, G.; Rodríguez-Nogueiras, M. Quantum Arithmetic for Directly Embedded Arrays. *Eng. Proc.* **2021**, *7*, 45. https://doi.org/10.3390/engproc2021007045

Academic Editors: Joaquim de Moura, Marco A. González, Javier Pereira and Manuel G. Penedo

Published: 21 October 2021

1. Introduction

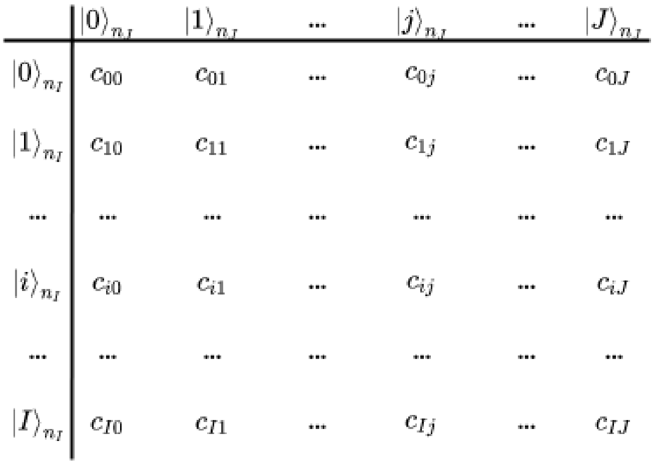
Quantum hardware and software are still in their early days of development; thus, the design of quantum algorithms typically focuses on low-level operations. Although one should always keep in mind the hardware limitations, especially when describing possible near-term implementations of quantum algorithms, it is convenient to pursue higher levels of abstraction. Apart from its long-term and algorithmic interest, a more abstract and standardized approach serves practical purposes too, for example, that of making the benchmarking of quantum computer performances a more solid and transparent process. In turn, this helps pushing the research and the development in quantum computation at all levels. In the present paper, we describe a novel framework for the design of quantum algorithms on a more abstract plane. To this aim, we have three proposals [1].

2. Results

Our first proposal consists in the definition of a quantum matrix, namely a quantum state organized in two registers:

$$|\psi\rangle = \sum_{i=0}^{I-1}\sum_{j=0}^{J-1} c_{ij} \, |i\rangle_{n_I} \otimes |j\rangle_{n_J} \, . \tag{1}$$

Specifically, we interpret the first register as the index running over the rows and the second register as the index running over the columns. This way of storing the information has a common ground with that of Flexible Representation of Quantum Images (FRQI) and the Novel Enhanced Quantum Representation (NEQR) [2,3]. The main difference with FRQI and NEQR is that we encode the information of the (*i*,*j*) entry of the matrix in the quantum amplitude. Intuitively, the matrix is a bi-dimensional memory array where the amplitude encodes the information stored in the (*i*,*j*) memory location (see Figure 1).



$$\begin{array}{c|cccccc}
 & |0\rangle_{n_J} & |1\rangle_{n_J} & \cdots & |j\rangle_{n_J} & \cdots & |J\rangle_{n_J} \\ \hline
|0\rangle_{n_I} & c_{00} & c_{01} & \cdots & c_{0j} & \cdots & c_{0J} \\
|1\rangle_{n_I} & c_{10} & c_{11} & \cdots & c_{1j} & \cdots & c_{1J} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
|i\rangle_{n_I} & c_{i0} & c_{i1} & \cdots & c_{ij} & \cdots & c_{iJ} \\
\cdots & \cdots & \cdots & \cdots & \cdots & \cdots & \cdots \\
|I\rangle_{n_I} & c_{I0} & c_{I1} & \cdots & c_{Ij} & \cdots & c_{IJ}
\end{array}$$

Figure 1. Quantum matrix structure.

The second proposal is a key technical feature about how we encode the information into the quantum amplitudes, the so-called direct embedding [4]. Namely, the information to be stored into the quantum matrix is directly loaded into the amplitudes without taking square roots, as it is instead usually done in the literature. Such loading choice has several and important implications in later stages of the quantum algorithms and—most importantly—the information stored into the quantum state is handled and combined more easily, because algebraic operations are not hampered by the presence of square roots.

The third and final proposal is to give a full overview of the complete pipeline, or overall structure, of a generic algorithm admitting implementation within this framework (see Figure 2).

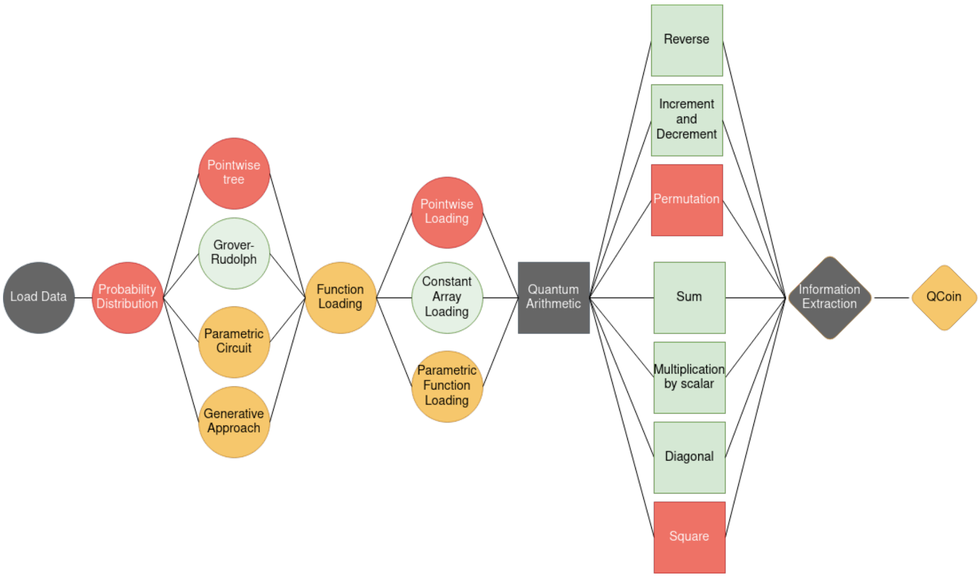


Figure 2. Structure of the pipeline. The dark gray nodes represent the three main steps; the nodes in red correspond to algorithms that are not efficient, for those in yellow the efficiency varies from case to case and, finally, the nodes in green represent efficient algorithms.

The first step of every algorithm corresponds to loading some input data. In the quantum case, it is often convenient to split this step into two sub steps:

1. Loading a probability distribution [4–6];
2. Loading a bi-dimensional function.

It is not strictly necessary to split the loading into two steps. Yet, we consider such splitting because—typically—we adopt different loading techniques for them: the probability distribution is loaded with a state preparation algorithm (e.g., a multiplexor binary tree); the function is loaded by means of another technique.

The second step of the pipeline in our framework corresponds to the implementation of various arithmetic operations, typically at the level of entire arrays or sub-arrays, and we refer to it as quantum arithmetic.

The third and last step of the pipeline corresponds to extracting the information that we have stored in the quantum state, namely the read-out of the state that encodes the result of the algorithm. There are multiple techniques for this purpose such as those appearing in [7–11].

3. Discussion

One of the advantages of organizing the pipeline as we have discussed is that it enjoys a modular structure; therefore, we can develop, analyze and improve each of the steps independently, achieving a better understanding of the problems in each domain.

In this paper, apart from introducing the framework, we provide some simple examples of arithmetic operations with finance in view. The possibility of implementing arithmetic operations has been considered in the literature since the early days of quantum computation. Apart from the quantum implementation of logical circuits corresponding to basic operations, such as the quantum adder [12,13], also the manipulation of "continuous" numbers has been studied. Let us mention some works which, at least in spirit, are closer to ours [14–17]. The difference with such approaches consists in the fact that we use a new embedding and organize the information into the matrix (1); these two aspects combined allow us to work in a transparent and simple manner.

Funding: This research was funded by the European Project NExt ApplicationS of Quantum Computing (NEASQC), funded by Horizon 2020 Program inside the call H2020-FETFLAG-2020-01(Grant Agreement 951821). Part of the computational resources for this project were provided by the Centro de Supercomputación de Galicia (CESGA).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Manzano, A.; Musso, D.; Leitao, Á.; Gómez, A.; Vázquez, C.; Ordóñez, G.; Rodríguez-Nogueiras, M. Quantum Arithmetic for Directly Embedded Arrays. *arXiv* **2021**, arXiv:2107.13872.
2. Le, P.; Iliyasu, A.; Dong, F.; Hirota, K. A flexible representation of quantum images for polynomial preparation, image compression and processing operations. *Quantum Inf. Process.* **2011**, *10*, 63–84. [CrossRef]
3. Zhang, Y.; Lu, K.; Gao, Y.; Wang, M. NEQR: A novel enhanced quantum representation of digital images. *Quantum Inf. Process.* **2013**, *12*, 2833–2860. [CrossRef]
4. Kubo, K.; Nakagawa, Y.O.; Endo, S.; Nagayama, S. Variational quantum simulations of stochastic differential equations. *arXiv* **2020**, arXiv:2012.04429.
5. Grover, L.; Rudolph, T. Creating superpositions that correspond to efficiently integrable probability distributions. *arXiv* **2002**, arXiv:quant-ph/0208112.
6. Nakaji, K.; Uno, S.; Suzuki, Y.; Raymond, R.; Onodera, T.; Tanaka, T.; Tezuka, H.; Mitsuda, N.; Yamamoto, N. Approximate amplitude encoding in shallow parameterized quantum circuits and its application to financial market indicator. *arXiv* **2021**, arXiv:2103.13211.
7. Shimada, N.H.; Hachisuka, T. Quantum Coin Method for Numerical Integration. *arXiv* **2020**, arXiv:1910.00263.
8. Brassard, G.; Hoyer, P.; Mosca, M.; Tapp, A. Quantum Amplitude Amplification and Estimation. *AMS Contemp. Math. Ser.* **2000**, *305*, 53–74. [CrossRef]
9. Suzuki, Y.; Uno, S.; Raymond, R.; Tanaka, T.; Onodera, T.; Yamamoto, N. Amplitude estimation without phase estimation. *Quantum Inf. Process.* **2020**, *19*, 75. [CrossRef]
10. Grinko, D.; Gacon, J.; Zoufal, C.; Woerner, S. Iterative Quantum Amplitude Estimation. *npj Quantum Inf.* **2021**, *7*. [CrossRef]
11. Giurgica-Tiron, T.; Kerenidis, I.; Labib, F.; Prakash, A.; Zeng, W. Low depth algorithms for quantum amplitude estimation. *arXiv* **2020**, arXiv:2012.03348.

12. Draper, T.G. Addition on a Quantum Computer. *arXiv* **2000**, arXiv:quant-ph/0008033.
13. Cuccaro, S.A.; Draper, T.G.; Kutin, S.A.; Petrie Moulton, D. A new quantum ripple-carry addition circuit. *arXiv* **2004**, arXiv:quant-ph/0410184.
14. Wang, S.; Wang, Z.; Cui, G.; Fan, L.; Shi, S.; Shang, R.; Li, W.; Wei, Z.; Gu, Y. Quantum Amplitude Arithmetic. *arXiv* **2020**, arXiv:2012.11056
15. Vedral, V.; Barenco, A.; Ekert, A. Quantum networks for elementary arithmetic operations. *Phys. Rev. A* **1996**, *54*, 147–153. [CrossRef] [PubMed]
16. Lloyd, S.; Braunstein, S.L. Quantum Computation over Continuous Variables. *Phys. Rev. Lett.* **1999**, *82*, 1784–1787. [CrossRef]
17. Xiaopeng, C.; Yu, S. *QBLAS: A Quantum Basic Linear Algebra and Simulation Library*; GitHub: San Francisco, CA, USA, 2019.