Quantum Multicore Processors

Nicolás Aguado and Adriana C. Bernal

1. Introduction

In recent years, the field of quantum computing has made significant strides, particularly with the advent of multi-core quantum processors. These methods promise to address some of the major challenges in quantum computing, such as scalability, error tolerance, and resource optimization. One of the possible functionalities for the implementation of communication in a *multi-core* approach is the **quantum state transfer**, which enables the transmission of quantum information between distant qubits. For these transfers at the chip scale, there are currently several techniques under investigation (4).

We explore the theoretical foundations of quantum teleportation (1) and its practical applications in multi-core quantum systems to enable quantum state transfers. Additionally, we propose a basic implementation framework for this purpose and evaluate its performance by increasing the number of cores and incorporating recent advancements in multi-core quantum processor architectures (7).

2. Quantum Teleportation

One of the technologies currently investigated for state transfer between qubits is the Quantum Teleportation protocol for two-level systems proposed by Bennett et al. (1). It is based on utilizing previously shared entanglement and classical communication to correlate measurement outcomes and reconstruct the original unknown quantum state for n number of qubits in a distinct remote subsystem.

For communicating two separate systems, A and B, a representation of an state of the art version of the proposed protocol for transmitting one qubit is as follows.

- An entangled Einstein-Podolsky-Rosen pair (2) (which is one of the Bell States) is generated in an intermediate source and then one of the qubits is distributed to A and the other one to B through fiber-optic quantum networks (5).
- A performs its processing and leaves its unknown outcome (or intermediate step) in qubit q_0^A. A then performs a Bell measurement on q_0^A and on the EPR qubit (its half of the EPR pair).
- Once the measurement on A's side is performed, the outcome is communicated to B via a classical channel. Since A obtains two classical bits of information from measuring, these are the two bits that are sent to B.
- Upon receiving the classical information, B can deduce a corresponding unitary operation from the set $\{I, X, Z, XZ\}$ that it will need to apply to its EPR pair qubit. This operation effectively corrects any phase or bit flip that arose due to the measurement outcome, thus reconstructing the original unknown state that A had into the EPR qubit that B has.

A diagram representation of this protocol can be found on Figure 1.

It is worth noting that the transmission of qubit states cannot be accomplished faster than light because B must wait for A's measurement result to arrive before B can recover the quantum state. Also, as A performs the Bell state measurement, it destroys the state of its qubits and they collapse to one of their possibilities. This loss of state is the reason that the QT does not violate the fundamental no-cloning principle (6).

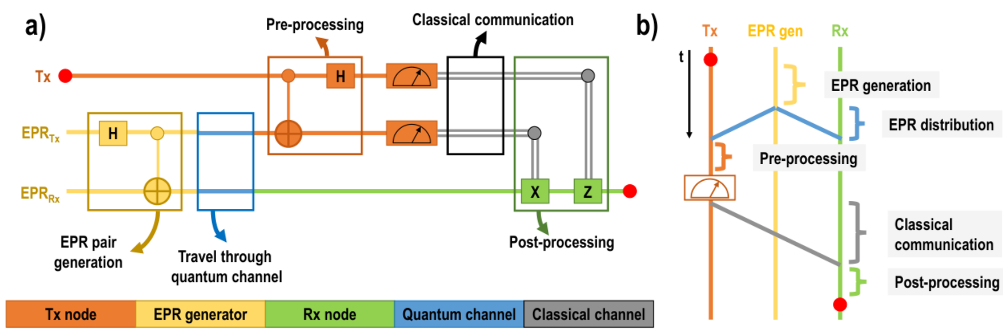

Figure 1: Teleportation circuit and time sequence diagrams, color-coded by the locations where each phase is performed. Extracted from Rodrigo et al. (7)

3. Implementation

3.1. Basic Multicore Architecture

A simple (unoptimized) implementation simulation of a quantum state transfer is shown on Figure 2.

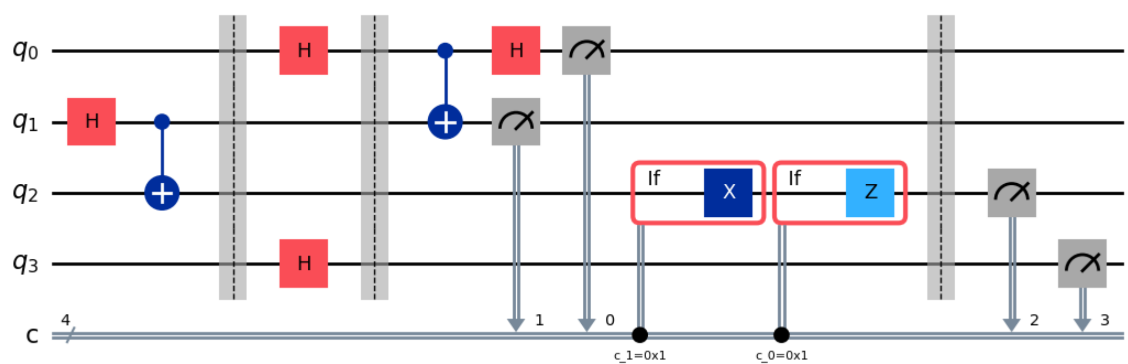

Figure 2: Quantum circuit for transfering a Hadamard state between two systems using QT

Two systems with one qubit each are displayed, using in total 4 quantum qubits (One from system A - q_0, another from system B - q_3 and two qubits for the EPR pair - q_1 and q_2) and 4 classical qubits (Two for performing the *teleportation* and the other two for getting the result). Barriers have been put to illustrate the different steps of the process.

In the first step, the EPR pair $\frac{1}{\sqrt{2}}|00\rangle + |11\rangle$ is created and distributed to both systems. As it is a simulation, both systems A and B are displayed together in the same circuit, with q_0 and q_1 belonging to system A and q_2 and q_3 to system B. Although this fact, logical isolation is mantained.

A hadamard state (H) is transferred from system A (q_0) to the ancilla qubit in system B.

In the second step, the wanted computation is performed on both systems in parallel. For this basic concrete example, the global task to compute is to put in superposition two Hadamard gates.

In the third state, the actual state transfer between both systems is made. As explained in Section 2., a bell measurement is performed and depending on the results some the bell pair qubit for system B (q_3) is transformed.

In the last section a measurement is performed on both qubits of system B. The results of the measurement, as shown on Figure 3, correspond to those of measuring two Hadamard gates in superposition, thus showing that the state transfer was performed correctly.

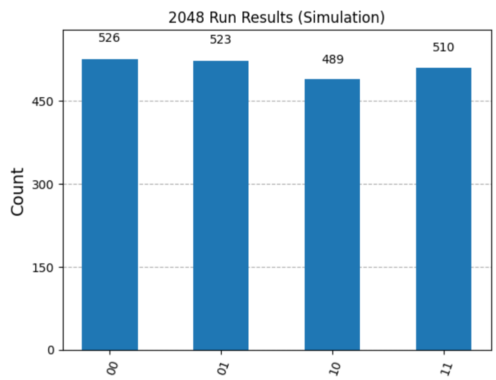

Figure 3: 2048 run results for the circuit shown in Figure 2

The complete qiskit code of the circuit can be found on the attached Notebook and in the following snippet.

MulticoreProcessorNAguACBernal.ipynb

```
qc = QuantumCircuit(4, 4)

qc.h(1)
qc.cx(1, 2)
qc.barrier()
qc.h(0)
qc.h(3)
qc.barrier()
qc.cx(0, 1)
qc.h(0)

qc.measure(0, 0)
qc.measure(1, 1)

with qc.if_test((qc.clbits[1], 1)):
    qc.x(2)
with qc.if_test((qc.clbits[0], 1)):
    qc.z(2)

qc.barrier()
qc.measure([2,3],[2,3])
```

3.2. Complex Multicore Simulation

We have explored the simulation of more complex quantum circuits using single-core and multi-core approaches. The analysis focuses on comparing the performance and fidelity. Moreover, we will explain the limitations towards speed. The code for this experiments can be found in the attached notebook.

The single-core simulation employs a quantum circuit with six qubits and three classical bits. The circuit is designed to perform basic quantum computations for three independent pairs of qubits. Each pair undergoes a Hadamard operation followed by controlled-NOT (CNOT) operation, and results are measured in three classical bits.

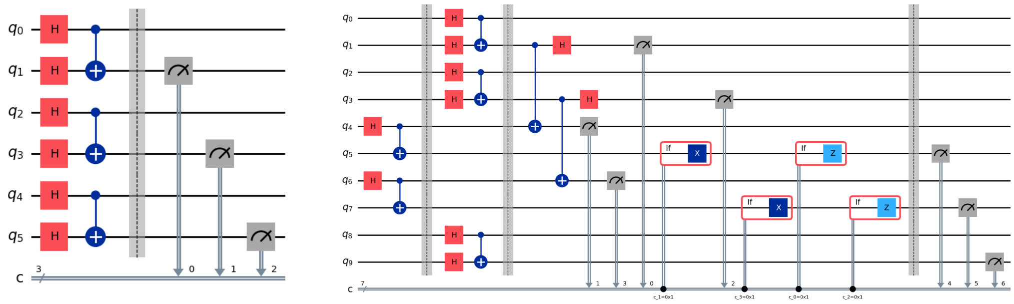

Figure 4: Single- and Multi-core circuits(respectively)

The single-core circuit shown on Figure 4 is designed as follows:

- Apply Hadamard gates on qubits 0, 2, and 4.
- Perform CNOT operations between qubits (0, 1), (2, 3), and (4, 5).
- A barrier is introduced to signify the end of the computations.
- Measure qubits 1,3, and 5 into classical bits 0, 1 and 2 respectively.

On the other hand, the multicore approach extends the complexity by distributing qubits across three systems (A, B, and C). It involves transmitting quantum states between systems using quantum teleportation via EPR pairs. This approach simulates the functionality of a multicore quantum processor where parallel computations are combined with inter-core communications.

The multi-core circuit shown on Figure 4 is designed as follows:

- Entanglement Creation: Create EPR pairs between qubit (4,5) and (6,7). Then, distribute the EPR pairs among the A, B and C systems.
- Parallel Computation: Perform the independent quantum computations on qubits in systems A, B, and C.
- State Transmission: Teleport the following states in order to get the intermediate states into System C for the final measurement
  - State of qubit 1 (from system A) to qubit 5 (in system C)
  - State of qubit 3 (from system B) to qubit 7 (in system C).

  (Using the classical measurement outcomes to accurately teleport the qubits).
- Measure the final states of qubits 5, 7, and 9 in system C.

As with the single-core approach, the multicore circuit is executed on a quantum simulator with 2048 shots, and the following results are shown for both single-and multi-core circuits in Figure 5)

The comparison of speed between the single-core and multicore circuit simulations is not straightforward, as we are simulating a single circuit in a quantum simulator. To assess speed more accurately, access to real quantum hardware would be required, which is not available in this context. Since we are using a simulator, we cannot provide a faithful comparison of speed with actual quantum hardware.

Regarding fidelity, the results from both simulations, single-core and multicore, are similar. As shown in figure 5, the fidelity of both approaches is comparable, suggesting that the additional complexity of the multicore simulation does not significantly impact the accuracy of the results in this case.

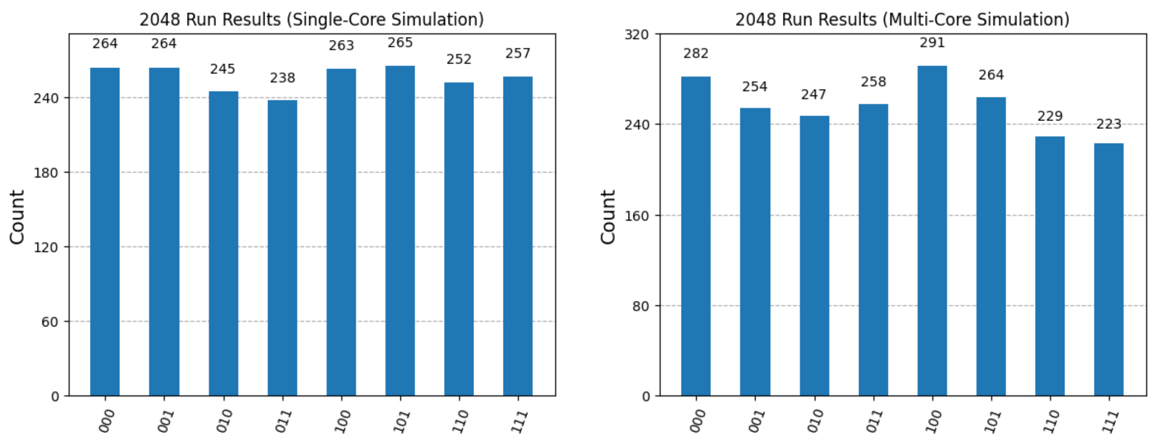

Figure 5: Comparison of Single-Core and Multi-Core Simulations

4. Task Distribution and Parallelization

Usually, in terms of distributed systems and parallelization, a *task* is referred to a predefined partial computation that performs parts of a global (common) computation. It would make sense to be able to leverage the performance benefits usual to these *task-based* parallel computations in a quantum computing context.

For that case, a task distribution algorithm would be designed to allocate computational tasks across the available resources (quantum cores in a multi-core environment). The goal of such an algorithm would be to optimize the performance of quantum computations by efficiently utilizing the available qubits.

4.1. Algorithmic Considerations

Usually, an algorithm is parallelized by allocating different iterations of loops to different cores. Thus, the majority of the time the mentioned *tasks* tend to be *symmetric*. Although quantum algorithms tend to operate in a different matter, we can assume quantum parallelization would work under the same principle. As a result, a task distribution algorithm for quantum multicore processors would operate by giving the same task to multiple cores. This means, the same perfectly parallelized task is assigned to multiple quantum cores. Each quantum core executes its assigned task using its local qubits and produces an intermediate result. Once all cores complete their tasks, an intermediate step combines the results, and often only one final result is retained for subsequent computations or as the overall output.

This algorithm would have a significant complexity to it, and it would certainly not be trivial. For each *task*, several things would need to be taken into account in order to create a *generalized* quantum distribution algorithm.

- Number of qubits needed to perform the computation
- Number of qubits available to store the intermediate result to be transmitted - m qubits
- Number of qubits available in order to perform the transmission - will be the same number as m.

Also, apart from those considerations, it has to be taken into accout that all practical quantum algorithms would need to be adapted using complex domain-related techniques (for example, as ilustrated for chemistry in Shang et al. (8)) and circuit analysis frameworks (for example, QuCT (9)) in order for them to be able to be parallelized. In the actual state of quantum algorithms, most of them have a very big level of interaction between qubits and the dependencies between them are too high in order to consider a automatic approach to quantum parallelization, rendering unfeasible to create a *generic* algorithm to perform this distribution.

4.2. Physical Considerations

One very important thing that must be taken into consideration is that for a quantum teleportation to take place, two qubits must be distributed (one to system that needs to *send*, and one to the system that needs to *receive*).

This EPR pair transmission needs to be performed through a quantum channel. There are several options to achieve this quantum channel (3) (5) but all of them are under heavy investigation and in very early stages. At this moment transmission of qubits is only possible at a very short range and under very specific circustances, and thus, quantum teleportation as a whole is limited to that range and conditions.

5. Conclusions

This investigation has explored the potential of quantum teleportation as a key enabler for communication and computation within multi-core quantum processor architectures. We have demonstrated the theoretical framework of quantum teleportation and its application in transferring quantum states between distinct systems, a crucial function for coordinating computations across multiple cores. Our simulations, while limited by the constraints of classical simulation of quantum systems, highlight the potential for parallel processing and distributed computation within a quantum multi-core environment. The comparison between single-core and simulated multi-core circuits, despite not providing a true speed comparison due to the simulation environment, demonstrates comparable fidelity, suggesting that the added complexity of teleportation and multi-core architecture does not inherently introduce significant errors in computation. This reinforces the viability of pursuing multi-core designs as a path towards scalable quantum computing.

However, significant challenges remain in realizing the full potential of this approach. Current quantum algorithms often exhibit complex qubit dependencies, hindering efficient parallelization and task distribution across multiple cores. Furthermore, the physical implementation of quantum teleportation is constrained by limitations in current quantum communication technologies, which restrict the range and practicality of inter-core communication.

While this work provides a valuable theoretical foundation and simulation-based exploration, future research must focus on developing more readily parallelizable quantum algorithms and advancing quantum communication technologies to overcome these limitations. Only then can the true computational power promised by quantum multi-core processors leveraging quantum teleportation be fully unlocked.

References

[1] Bennett, C.H., Brassard, G., Crépeau, C., Jozsa, R., Peres, A., Wootters, W.K.: Teleporting an unknown quantum state via dual classical and einstein-podolsky-rosen channels. Phys. Rev. Lett. **70**, 1895–1899 (Mar 1993). https://doi.org/10.1103/PhysRevLett.70.1895, https://link.aps.org/doi/10.1103/PhysRevLett.70.1895

[2] Einstein, A., Podolsky, B., Rosen, N.: Can quantum-mechanical description of physical reality be considered complete? Phys. Rev. **47**, 777–780 (May 1935). https://doi.org/10.1103/PhysRev.47.777, https://link.aps.org/doi/10.1103/PhysRev.47.777

[3] Escofet, P., Ovide, A., Almudever, C.G., Alarcón, E., Abadal, S.: Hungarian qubit assignment for optimized mapping of quantum circuits on multi-core architectures. IEEE Computer Architecture Letters **22**(2), 161–164 (2023). https://doi.org/10.1109/LCA.2023.3318857

[4] Escofet, P., Rached, S.B., Rodrigo, S., Almudever, C.G., Alarcón, E., Abadal, S.: Interconnect fabrics for multi-core quantum processors: A context analysis. In: Proceedings of the 16th International Workshop on Network on Chip Architectures. p. 34–39. NoCArc '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3610396.3623267, https://doi.org/10.1145/3610396.3623267

[5] Kozlowski, W., Wehner, S.: Towards large-scale quantum networks. In: Proceedings of the Sixth Annual ACM International Conference on Nanoscale Computing and Communication. NANOCOM '19, Association for Computing Machinery, New York, NY, USA (2019). https://doi.org/10.1145/3345312.3345497, https://doi.org/10.1145/3345312.3345497

[6] Nielsen, M.A., Chuang, I.L.: Quantum Computation and Quantum Information: 10th Anniversary Edition. Cambridge University Press (2010)

[7] Rodrigo, S., Abadal, S., Almudéver, C.G., Alarcón, E.: Modelling short-range quantum teleportation for scalable multi-core quantum computing architectures. In: Proceedings of the Eight Annual ACM International Conference on Nanoscale Computing and Communication. NANOCOM '21, Association for Computing Machinery, New York, NY, USA (2021). https://doi.org/10.1145/3477206.3477461, https://doi.org/10.1145/3477206.3477461

[8] Shang, H., Fan, Y., Shen, L., Guo, C., Liu, J., Duan, X., Li, F., Li, Z.: Towards practical and massively parallel quantum computing emulation for quantum chemistry. npj Quantum Information **9**(1), 33 (Apr 2023). https://doi.org/10.1038/s41534-023-00696-7, https://doi.org/10.1038/s41534-023-00696-7

[9] Tan, S., Lang, C., Xiang, L., Wang, S., Jia, X., Tan, Z., Li, T., Yin, J., Shang, Y., Python, A., Lu, L., Yin, J.: Quct: A framework for analyzing quantum circuit by extracting contextual and topological features. In: Proceedings of the 56th Annual IEEE/ACM International Symposium on Microarchitecture. p. 494–508. MICRO '23, Association for Computing Machinery, New York, NY, USA (2023). https://doi.org/10.1145/3613424.3614274, https://doi.org/10.1145/3613424.3614274