On the Quantum Performance Evaluation of Two Distributed Quantum Architectures

Gayane Vardoyan g.s.vardoyan@tudelft.nl
Matthew Skrzypczyk m.d.skrzypczyk@tudelft.nl
Stephanie Wehner s.d.c.wehner@tudelft.nl
QuTech and Kavli Institute of Nanoscience, Delft University of Technology

ABSTRACT

Distributed quantum applications impose requirements on the quality of the quantum states that they consume. When analyzing architecture implementations of quantum hardware, characterizing this quality forms an important factor in understanding their performance. Fundamental characteristics of quantum hardware lead to inherent tradeoffs between the quality of states and traditional performance metrics such as throughput. Furthermore, any real-world implementation of quantum hardware exhibits time-dependent noise that degrades the quality of quantum states over time. Here, we study the performance of two possible architectures for interfacing a quantum processor with a quantum network. The first corresponds to the current experimental state of the art in which the same device functions both as a processor and a network device. The second corresponds to a future architecture that separates these two functions over two distinct devices. We model these architectures as continuous-time Markov chains and compare their quality of executing quantum operations and producing entangled quantum states as functions of their memory lifetimes, as well as the time that it takes to perform various operations within each architecture. As an illustrative example, we apply our analysis to architectures based on Nitrogen-Vacancy centers in diamond, where we find that for present-day device parameters one architecture is more suited to computation-heavy applications, and the other for network-heavy ones. We validate our analysis with the quantum network simulator NetSquid. Besides the detailed study of these architectures, a novel contribution of our work are several formulas that connect an understanding of waiting time distributions to the decay of quantum quality over time for the most common noise models employed in quantum technologies. This provides a valuable new tool for performance evaluation experts, and its applications extend beyond the two architectures studied in this work.

1. INTRODUCTION

Quantum communication promises to fundamentally enhance internet technology by enabling application capabilities that are impossible to attain classically. To support distributed quantum applications, the architecture of a quantum network node should be capable of two key functions: first, it should enable local quantum computation, *i.e.*, the execution of quantum gates and measurements, at each end node [3] in the network on which applications are run. Second, it should enable the generation of quantum entanglement between any two nodes in such a network. A typical quantum network application consists of both local quantum computations and the generation of entanglement, where different applications may have more demand for local quantum processing, or for entanglement generation.

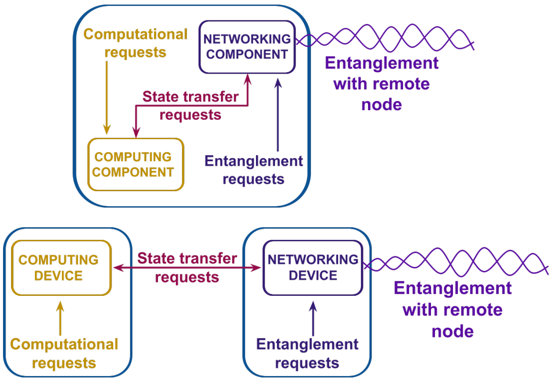

Figure 1: Two possible architectures for a quantum processor interfaced to a quantum network: in the *SD architecture* (top), the processor and the network device are the same device, with an internal logical or physical division into a computing or networking component. In the *DD architecture* (bottom), two separate devices are used. An application interacts with the system by making three types of requests: local quantum computations (on the computing component/device), network operations (entanglement generation), and movement (state transfer) of generated entanglement into the processor. The latter requires cooperation from both processing and network devices.

A key performance metric in a quantum network is the quality of entanglement being generated between two remote network nodes. On a quantum processor, we furthermore want to understand the quality of a quantum gate's execution, and consequently the quality of the quantum program being executed. The quality of a quantum state is measured by its *fidelity*, and the quality of executing a gate by its *gate fidelity*. This is a number in the interval $[0, 1]$ that measures the closeness of the state (or gate) to a desired target implementation – the larger it is, the closer we are to the target implementation. In this work, we focus on these *quantum performance measures* – specifically, we study gate fidelity in distributed quantum architectures, as well as the fidelity of entanglement generated by applications that run on them.

Given the need to perform both local quantum operations as well as network operations in order to realize distributed quantum applications, we consider two different general architectures for interfacing a networked quantum processor to a quantum network (Figure 1). In the first, which we call the single-device (SD) architecture, the same device is used to perform both network operations as well as local quantum computation (Figure 1, top). This is the case in all present-day implementations, such as networked quantum processors based on Nitrogen-Vacancy centers in diamond [2], or Ion Traps [1]. Abstractly, one can think of these as quantum processors that have two different types of qubits: communication qubits (networking component) with an optical interface for remote entanglement generation, and storage qubits which can only be used for local processing. Limits on experimental control typically prohibit the simultaneous execution of local (two-)qubit gates, and entangling operations. That is, while entanglement generation is in progress, local quantum processing is on hold, and vice versa. The time necessary for local gate execution only depends on the local processing speed. However, the time required for entanglement generation depends on the physical distance to the remote network node. Consequently, in a situation in which the remote node is at a distance, local processing may need to be suspended for a significant amount of time while entanglement generation is in progress.

In the second architecture, we hence consider a scenario in which the system is enhanced by the introduction of a dedicated network device solely used for the purpose of entanglement generation with remote network nodes (Figure 1, bottom); we refer to this as the double-device (DD) architecture. In this architecture, the network device is linked internally to the processor to accommodate entangled qubit transfers from the former to the latter. Since state transfers at such short (on-chip) distances are relatively fast, remote entanglement generation via the external networking device and computations on the processor only need to be suspended for a short amount of time while the state of the entangled qubit is being transferred.

While evaluating these architectures, we first note that performance depends on whether we execute a computation-heavy or a network-heavy application. Second, the performance of both architectures depends on the inherent quality of the quantum devices used to realize them. One key concern is the ability of the quantum device to store quantum states during waiting times: a lower memory lifetime means that waiting times have a much larger impact on the quality of execution. Similarly, the quality of the interface between the processor and the network device is of concern in DD architectures, as it may reduce the quality of the entanglement being transferred. Finally, while the DD architecture may be of great intuitive appeal, it is much more cumbersome to realize experimentally since one additional device must be constructed. This raises a very practical question as to what achieves more benefit to application performance: implementing the DD architecture, or investing efforts into improving the quality of the components (*e.g.*, to achieve higher memory lifetimes) in the SD architecture.

2. MODELING AND ANALYSIS

We model the SD and DD architectures as $M/HYPO_3/1$ queueing systems, where the arrivals correspond to entanglement requests and are a Poisson process with rate λ_e. The service times are hypo-exponentially distributed with three service stages, each of which are exponentially-distributed with parameters μ_e, λ_m, and μ_m, corresponding to the rate of entanglement generation, moving request arrival, and moving request completion, respectively. Entanglement requests are processed according to a first-in, first-out policy, and the next entanglement request cannot begin processing until all three stages of the previous request have been completed, since the communication qubit must be freed before entanglement generation may be attempted again. Local quantum computation is assumed to consume negligible time (this assumption is relaxed for simulations).

We evaluate the average gate fidelity for computation requests on the SD and DD architectures using a composite noise model $\mathcal{C}$ of amplitude damping and dephasing (with corresponding memory lifetime parameters T_1 and T_2), and waiting time distributions obtained from the analysis of the Markov chain, yielding

$$F_{avg}^{(1)}(\mathcal{C}, G) = 1 - \frac{\lambda_e}{2}\left(\frac{1}{\mu_e} + \frac{1}{\mu_m^{(1)}}\right) + \frac{\lambda_e}{6}\left(\frac{2T_2}{\mu_e T_2 + 1} + \frac{T_1}{\mu_e T_1 + 1} + \frac{2T_2}{\mu_m^{(1)} T_2 + 1} + \frac{T_1}{\mu_m^{(1)} T_1 + 1}\right),$$

$$F_{avg}^{(2)}(\mathcal{C}, G) = 1 - \frac{\lambda_e}{2\mu_m^{(2)}} + \frac{\lambda_e}{6}\left(\frac{2T_2}{\mu_m^{(2)} T_2 + 1} + \frac{T_1}{\mu_m^{(2)} T_1 + 1}\right).$$

Above, the superscripts denote the architecture ($^{(1)}$ for SD, $^{(2)}$ for DD) and G is any quantum gate. The average fidelity of the entanglement that is moved into memory is given by

$$F_e(\mathcal{C}) = \frac{1}{4} + \frac{\lambda_m}{4}\left(\frac{T_1}{\lambda_m T_1 + 1} + \frac{2T_2}{\lambda_m T_2 + 1}\right).$$

3. SUMMARY OF FINDINGS

From our analysis and numerical observations, we find stark contrasts between the two architectures. On the one hand, when implemented with memories of identical quality, the DD design dominates in terms of gate fidelity. However, in a more practical scenario, wherein the DD design's more complex manufacturing would impair its memory lifetimes, the SD design can yield higher gate fidelities, and is more robust to longer computation times. Further, for present-day parameters, the SD design is more hospitable to the entanglement fidelity; the advantages are especially evident in the high entanglement generation rate regime. This suggests that for present-day parameters, the DD design is more suitable for settings such as long-distance quantum communication, with lower entanglement generation rates and lighter computational demands. In contrast, the SD design is better suited for settings such as a distributed quantum computing cluster where high entanglement generation rates can be achieved and longer computations must be performed.

4. REFERENCES

[1] V. Krutyanskiy, M. Meraner, J. Schupp, V. Krcmarsky, H. Hainzer, and B. P. Lanyon. Light-matter entanglement over 50 km of optical fibre. *npj Quantum Information*, 2019.

[2] M. Pompili, S. L. Hermans, S. Baier, H. K. Beukers, P. C. Humphreys, R. N. Schouten, et al. Realization of a multinode quantum network of remote solid-state qubits. *Science*, 2021.

[3] S. Wehner, D. Elkouss, and R. Hanson. Quantum internet: A vision for the road ahead. *Science*, 2018.