Towards Efficient Resource Allocation and Request Scheduling in Quantum Networks

Poorvika S K[1] , Prajna[2], P Raghavendra[3], Pavan S N[4], Pradeep Nayak[5]

[1]Student, Computer science and Engineering (Internet of Things, Cybersecurity, Including Blockchain Technology), Alvas Institute of Engineering and Technology, Karnataka, India
[2]Student, Computer science and Engineering (Internet of Things, Cybersecurity Including Blockchain Technology), Alvas Institute of Engineering and Technology, Karnataka, India
[3] Student, Computer science and Engineering (Internet of Things, Cybersecurity Including Blockchain Technology), Alvas Institute of Engineering and Technology, Karnataka, India
[4] Student, Computer science and Engineering (Internet of Things, Cybersecurity Including Blockchain Technology), Alvas Institute of Engineering and Technology, Karnataka, India
[5] Faculty, Information Science and Engineering (ISE), Alvas Institute of Engineering and Technology, Karnataka, India

Emails: poorvikakarkera@gmail.com[1], prajna202327@gmail.com[2], 538pu1pedada@gmail.com[3], hedgepavan41@gmail.com[4] , pradeep@aiet.org.in[5]

Abstract

Quantum networking is a new field for study, to see what it can do. It works by linking quantum systems, by making a path of connected qubits from one end to the other using repeaters that can work over long distances. People have made a good design in the past that can keep the link between two quantum devices separated by time, using a method called time slot. They break up the process of making an end-to-end path into two parts: path choosing and timing. Path choosing finds the best repeaters to link the two ends, but only if they are already entangled well. Timing finds which pairs of ends get serviced in each time slot and which have to wait for later slots. No one has studied timing much before, especially when quantum noise is part of the system, which makes both tasks even harder. We plan to solve this with a set of simple algorithms, three of which we give examples of here. The goal is to keep delay low and system use high, meaning lots of entangled qubits with good quality and rate, from a single source to the many quantum devices the system connects. We test our idea using detailed simulations of quantum networks with noise, using different layouts of nodes and requests arriving randomly as in the real world. Our results show there is a trade-off at the system level between fairness for the applications and the quality of the entanglements, including the rate and fidelity.

Keywords: *End-to-end entanglement, noisy quantum repeaters, quantum Internet, routing.*

1. INTRODUCTION

Quantum computers are now pushing the bounds of what they can do, and their benefits will grow when a quantum network is built to connect remote quantum computers together. This quantum network will help with distributed quantum computing and many more uses. A roadmap for building this network is carefully summarized in [43], while a detailed list of the needed technologies and where they stand now is reported in [2]. The main part of the network is the quantum repeater, which lets the transfer of quantum states go through entanglement swap [5]. Fig. 1 [35] shows an example of end-to-end entanglement between two separate quantum systems (call them nodes) connected with a repeater using quantum sources. The quantum sources send local entangled EPR pairs to both of the end systems (the two end systems are called Alice and Carol here, and the repeater is called Bob once he has done a projective measurement on his 2 particles (|0 Bob and |1 Bob)) and communicated the measurement results to, say, Carol. There, Carol can do local Pauli operations, X- and Z-gates, on her qubit |0 Carol to fix it, which then makes a true end-to-end entanglement with the qubit Alice has, |0 Alice. Even though the quantum repeater is not yet a commercial product, many experiments have shown it should work fine ([4], [38], [39], [45], [46]). Also, connecting these repeaters together can, in theory, create end-to-end entanglement over any distance between nodes. In reality, technical limitations are:

1) signal power drops as it moves down Fiber optic cables or through air from ground to satellite links [22], and 2) quantum memories have short coherence times [7]. These limits give a lot of challenges for making these quantum networks with entanglement swapping. While advances in how the devices are built, and in areas like distillation [18], purification [40], and quantum error correction [16], will slowly lessen these early problems, we don't expect complete answers anytime soon.

So, in this article, we look at a area that is based on near-future tech and has already got some attention in published work: we assume that the local entanglement at each repeater for each link to its neighbour can fail with a reasonable chance (but it is heralded using some non-described technology), and that the system keeps track of which links can be used to send out the entanglement status of the qubits (it's roughly the quantum Internet version of sending information through the traditional Internet). So, the controller tries to fill demands from the applications above by choosing a route of several links that only lasts until the next round of local link entanglements is established. Whether this is a practical way to do this in a real quantum net, in a local or wide area, only the future will tell. For now, we build on prior work in this area, which is reviewed in Section II, and study one specific part that we haven't seen discussed in other papers before: request scheduling. This is about picking which end-to-end entanglement demands to satisfy at one time, based on the most recent local entanglement outcome, if it's not possible to satisfy all current demands right away. The model of this system is explained in Section III, while the idea of

request scheduling is covered in Section IV, as part of a more general issue, quantum routing. Still in the same section, we also suggest a general idea, iterative scheduling, and three examples to show how they can reach different goals, which are then studied in Section V.
These results show there is a trade-off between how much the system can do, in terms of how many entanglements it can create and how good they are (fidelity), and how fair these benefits are across applications, which is why it is an interesting place to do more work on the details, as explained in the conclusions (see Section VI).

2. THE STATE OF THE ART

This part tells about the state of the art on quantum routing, which we have talked about in Section I and fully explain in Sections III and IV below. For the record, we mention that there is more and more in the lit about sparse parts of quantum routing, but it is not that relevant to the issue we are after and so we do not go into it. It includes, among others: how to define protocol stacks [12], [26], [32]; how to identify capacity regions [42]; study tools for quantum network performance [3]; how to get multipartite entanglement all over [27]; design networked quantum apps [13]. A key area, instead, is the path choice, i.e., picking which repeaters to connect along to end-to-end entangle two systems among multiple possible choices in a quantum network. This is very similar to choosing a route based on a criterion (or cost) in traditional networks, as those generally used in older networks don't account for the specifics of quantum networks, like the chance that the link will fail due to a lack of local entanglement or its fidelity. This is the starting point of [41], where many possible metrics are defined and tested for use with the classical Dijkstra's shortest path first (SPF) algorithm [15]. Two key learnings are that SPF does well for certain goals, and that the number of Bell pairs produced per second on a single link beats other physically related metrics, like the loss of the channel or the inverse of the channel transmittance. A more complex routing metric is explained in [6] and has been proven to work well for selecting the best path between two nodes when used with a special routing protocol that runs in polynomial time.
Key difference: These past works did not address the opportunities and challenges of simultaneously activating multiple end-to-end links in the quantum network, which we do using our system model.
Instead, they study multipath routing in [29], which our system model also inspired (see Section III). They argue for both a global link state routing method, where decision making is done by a main controller, and a local link state protocol, where repeaters act on their own and make decisions without coordination. In [28], entanglement swapping is used as a way to boost the total capacity that would be available if only local links were used. They have used a maximum flow problem to make their model, which we also do, to create an ideal quantum routing problem in Section IV-A.

Key difference: We have extended the two models seen before by adding the question of when to run the requests. Also, we have added decoherence to the quantum memories in our performance testing, which was not included in those studies.
Other works explored a different idea where the concept of a virtual quantum link was used. A virtual link is an entanglement between a bunch of nodes, with the middle ones being repeaters, which can be linked together with other virtual links to make the final entanglement between the two end nodes. This idea lets us separate the quantum problem in two parts: the first is to make the virtual links and connect them to make an arbitrary virtual topology which we assume to be regular; then, route on the virtual topology to make the end-to-end links you want. This problem has been studied in [34], which also offers a resource-efficient protocol for decentralized routing in a ring and sphere topology, and in [9], with ring, grid, and recursive topologies, i.e., topologies that adapt dynamically by replacing edges in the topology at each step. Also, Chakraborty et al. [9] explored two other models: continuous, where virtual Background links form even if there is no demand for full entanglement, and on demand, which reacts to the demand arriving with no pre-made virtual links available. Big difference: virtual link routing isn't just for our system model, but in theory it is a candidate alternative to our contribution, and we plan to compare them in future work. End, we mention the research area of stochastic routing in wireless sensor networks (WSNs), which has some similarities to quantum routing. In fact, a WSN is usually multiloop, i.e., the nodes are expected to forward messages not meant for them, in addition to doing their regular sensing jobs. But energy is often a limited resource, so the nodes switch between active and inactive periods: when they do this randomly, the routing process can't rely on all links being active at once, which is similar to what occurs in a quantum network due to a lost local entanglement. The issue has been studied before, for example, Ribeiro et al. [33] created a distributed algorithm that finds the best routing probabilities based on the physical-layer features of the links, which is then expanded in [1], which also looks at the sensing process in addition to multiloop data transfer.
Big difference: there is a key difference between stochastic routing in WSN and the quantum routing problem discussed in this article: as we will explain more fully in Section III, we assume that local link entanglement is broadcast each time slot, so a (reduced) network topology is known with certainty when routing decisions are made. We acknowledge a possible additional research path where routing occurs before the local link entanglement results are known, but this is outside the scope of this article.

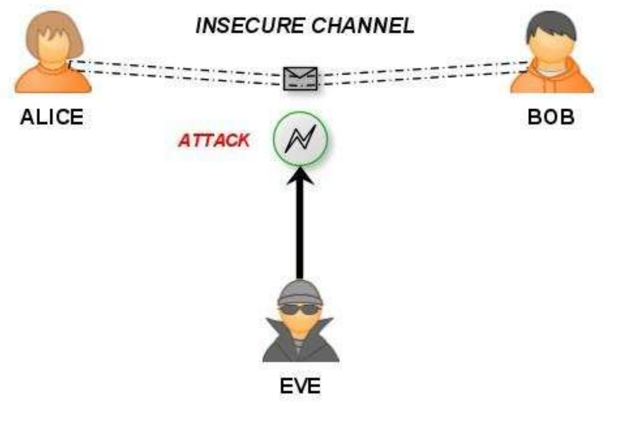
3. SYSTEM MODEL

This part explains the system model in this paper.
First, we show the network architecture and its main points (see Part III-A), then show that the method is time-slotted (see Part III-B).

A. QUANTUM NETWORK ARCHITECTURE

The form of a quantum network is shown in Fig. 2.
It has the following parts:

1) quantum computers: machines that want to form end-to-end entanglement with other computers through the network, e.g., for distributed quantum tasking;
2) quantum repeaters: tools that link quantum computers and other quantum repeaters just to make such entanglement possible over a long distance beyond what current quantum communication tech can do.
As shown in Section I, end-to-end entanglement between two units, for instance, Host A and Host C in the sample, at a specific moment, can only occur if there is a suitable route, such as σ = {a, b, c, d}, and the following occurs:
1) entanglement takes place locally between every pair of repeaters along the route (that is, ab, bc, and cd).
2) the intermediate points (b and c) apply a swap step (like Bob in Fig. 1) by a projective measurement;
3) the measurement's outcome is sent, through classical signals, to one end point (either a or d) to correct the local qubit with a series of X- and Z-operator quantum memory steps.
Without losing generality, we no longer make a distinction between quantum computers and quantum repeaters hereafter, and refer to both simply as nodes. Two nodes are seen as neighbors if quantum information in a qubit (called a flying qubit) can be shifted between the two directly. Usually, this involves two nodes linked by a fiber optic wire that can carry quantum bits, encoded in light pulses. The link that physically connects two neighbors is called a link; it's feasible that this link can also carry classical data, though the quantum and classical channels may vary in their properties.



4. QUANTUM ROUTING

This section covers the quantum routing problem, as defined by the system model in the section before. First, we give the problem in a formal way (see Section IV-A), and then we show we can split it into two smaller problems: scheduling and path choice.

A. ROUTING PROBLEM

In our model, the routing problem is solved by the quantum network controller using regular computers. Here are the inputs for the routing problem.

1) The logical layout of the network as a graph (V, E) and the physical lengths between any two nodes with an edge in (V, E).

2) The set of local entanglements S that have been successful in the first part of this time slot. End-to-end entanglement paths will have to be set up in a reduced subgraph(V,E), which is given by E includes all the quantum network links that are successful, that is,where the local entanglements from this time slot are found to be successful by both repeaters.

3) The list of pairs of nodes that want to set up an end-to-end entanglement path, including the pairs that arrived in old time slots but could not be served because there were not enough resources, and the new pairs that arrived this time slot.The routing problem output is a set of end-to-end entanglement paths; for each path the following information is needed.
1) The list of swap nodes, and for each swap node, the location of the memory slots that must be measured
2) The positions of the memory slots at each of the end nodes
3) A path number, to tell apart different paths where they coexist in the same time slot
4) An end node number, where all the corrections
are to be sent. All of the above need to be coded in messages that are sent over the regular links, i) to make the swaps run the measurements. ii)for all the nodes on the way to carry the messages with the correction bits toward the right end node of the path iii) for the end node to do the corrections on the local qubit and iv) for both end nodes to put the local qubit in the local layer.
Although we have a simple protocol designed to run the simulation experiments, the results of which will be shown in section V, we do not go into this subject and it is still an open question research wise (see [10], [12], and [32]). Hereafter, we assume there exists such a protocol and that it is effective enough not to cause a noticeable delay, except for the propagation of the messages over the physical links. In general, for a given input, there may be more than one possible output, thus the controller will choose the best.

However, the decisions made in a given time slot also affect future ones: pairs not assigned a path during this routing step will remain unserved and will need to be assigned a path later, or dropped. So, at least in principle, when a decision is made in a certain time slot t, the controller has to think about what happened in all previous slots (= the pairs that were requested but not yet served) and what will happen in all future ones (= the chances to better serve the pending requests in the next slots and the further requests that will arrive). We set mathematically an ideal routing problem as a maximum multicommodity flow problem over the time horizon from t=0 to t=T

Algorithm 1. The Iterative Partition Scheduling Algorithm

```
set optimal makespan to ms = ∞ and iteration number iter = 1;
do
    schedule G(V, E) by a specific algorithm to get makespan mt;
    find the critical path of G(V, E);
    initialize gain and target node, gain = ∞, n_opt = n_1;
    for node n_i in the critical path do
        try to split n_i and estimate gain gt;
        update gain of split node gain and critical node n_opt;
    end
    update G(V, E) to G'(V', E') by splitting critical node n_opt evenly;
    update G(V, E) by G'(V', E') , ∇(m) = mt − ms, ms = max(ms, mt)
    and increase iter by 1;
while iter < iter_threshold or ∇(m) > makespan_threshold;
return G'(V', E')
```

5. PERFORMANCE CHECK

Here, we check how well our system can do under two plans and two network shapes:

Nodes in a grid, see part B. Nodes placed randomly on a flat circle, see part C. Before we start, we tell how and what we use to check (see part A).

A. HOW AND WHAT WE USE

The results in this part are made with a tool that simulates quantum info using event steps (Net Squid)2, a Python script you can download and run for free. For example, I used this tool to get results in studies [12]. I used the NetSquid tools to write the parts needed to build the system model in part III; for those who want to make the same or add more, the code is free on GitHub3. This also has the plans and post-check steps, and all the results from the tests.

The key parts of the tests are shown in Fig. 9. It shows three nodes (Alice, Bob, Carol) connected by quantum and normal links, both having the same size; this size can change for different node pairs, depending on the test type, as I will explain next. In the middle of each quantum link is a quantum source that makes EPR-entangled pairs go to each node its connected to. The quantum channel adds a fixed delay δ = d c for each link of length d (km), where c is the speed of travel (set to 200 000 km/s in the tests). Also, the quantum channel might lose one qubit with the chance:

P init is the chance that the qubit is lost right after it is made, because of nonperfect parts of the gear used to make/entangle, and η is how much the optical cable gets weaker over length in dB/km.

The normal channel is thought to not have errors, and it just adds a delay δ = d c , which are safe guesses given the best links now for quick chats over optical cables, if the message has not too many parts.

B. GRID TOPOLOGY

In this part, I tell about what we got in square grid logical topologies, which we have also used in [9] and [29].

Such a setup can be interesting because it shows how we need to cover a big space with a lot of evenly (or nearly evenly) spaced quantum repeaters in a nice grid.

We have duplicated the same tests in two ways: 1) physical regular grid, where the distance between nodes was set to exactly 1.8 km; 2) physical irregular grid, where the distance between nodes was picked from a uniform range in [1.08, 2.52] km.

The number of nodes was 25, and the offered load 1/λ was increased from 1 to 10 in different tests to see how things change with more load.

We tested 10 000 time slots for each setup.

First, I show the normal entanglement rate, which I define as the ratio between the number of paths made and the number of requests that come in (so it is always less than or equal to 1).

This measure is in Fig. 12 with Strict FIFO only as the load gets bigger for these three cases: 1) fidelity more than 0.8, 2) fidelity more than 0.9, and 3) any fidelity.

The results show that the quantum network is not very busy until 1/λ drops below 4, because the Any F lines are very close to 1 (the regular and irregular lines overlap in the figure).

Even with such low load, for F > 0.8 and F > 0.9 the entanglement rate is much lower: this is because there are requests for pairs or nodes that are far apart physically.

With 1/λ above 4, all lines start to fall off (especially Any F), which means resources are no longer enough to handle most incoming requests.

Regardless of load, with F > 0.8 and F > 0.9 a regular grid shows a nontrivial performance gain over an irregular grid.

This indicates that the physical setup of nodes is important in determining overall performance.

6.CONCLUSION

In this paper, we looked at the quantum routing problem in a timeslot system model. We divided it into path selection, which means choosing a path for a specific application request, and scheduling, which means deciding which application requests to serve this time slot. Path selection was already studied before, but scheduling is new. For it, we created a general framework. We named three types: Best FIFO and Random FIFO. Best FIFO prefers shorter paths over longer ones by queueing the shorter ones first in a deterministic or stochastic manner. Random FIFO queues application requests randomly. Best FIFO simply serves application requests in the order they arrive. Our simulations showed that scheduling greatly affects performance. It influences entanglement rate, fidelity, and delay. Also, we found a fairness tradeoff in how application requests are treated versus system-level metrics, which allows more research and understanding of quantum networks and their uses.

Besides the future work we already listed, we see the following future research topics for this paper: Extending to multipartite entanglement, such as in [42], Integrating with communication protocols like [26], and link layer stacks like [12], Analyzing how distributed quantum applications relate to the underlying network [14], Studying the fairness versus efficiency tradeoff in local link state routing protocols like [9], Extending to multiple local entanglement per node pair, as in [29], Defining an SLA for distributed quantum applications [11], and service differentiation [19], and running more simulations with bigger network layouts and application request types.

REFERENCE:

[1] R. Arroyo-Valles, A. Simonetto, and G. Leus, “Consistent sensor, relay, and link selection in wireless sensor networks,” Signal Process., vol. 140, pp. 32–44, 2017, arXiv: 1705.02298.

[2] D. Awschalom et al., “Development of quantum interconnects (QuICs) for next-generation information technologies,” PRX Quantum, vol. 2, no. 1, Feb. 2021, Art. no. 017002, doi: 10.1103/PRXQuantum.2.017002.

[3] B. Bartlett, “A distributed simulation framework for quantum networks

and channels,” Aug. 2018, arXiv:1808.07047.

[4] M. K. Bhaskar et al., “Experimental demonstration of memory-enhanced quantum communication,” Nature, vol. 580, no. 7801, pp. 60–64, Apr. 2020.

[5] A. S. Cacciapuoti, M. Caleffi, F. Tafuri, F. S. Cataliotti, S. Gherardini, and G. Bianchi, “Quantum internet: Networking challenges in distributed quantum computing,” IEEE Netw., vol. 34, no. 1, pp. 137–143, Jan. 2020, doi: 10.1109/MNET.001.1900092.

[6] M. Caleffi, “Optimal routing for quantum networks,” IEEE Access, vol. 5, pp. 22299–22312, 2017, doi: 10.1109/ACCESS.2017.2763325.

[7] M. Caleffi and A. S. Cacciapuoti, “Quantum switch for the quantum internet: Noiseless communications through noisy channels,” IEEE J. Sel. Areas Commun., vol. 38, no. 3, pp. 575–588, Mar. 2020, doi: 10.1109/JSAC.2020.2969035.

[8] M. Caleffi, A. S. Cacciapuoti, and G. Bianchi, “Quantum internet: From communication to distributed computing,” in Proc. 5th ACM Int. Conf. Nanoscale Comput. Commun., 2018, pp. 1–4, doi 10.1145/3233188.3233224.

[9] K. Chakraborty, F. Rozpedek, A. Dahlberg, and S. Wehner, “Distributedrouting in a quantum internet,” 2019, arXiv:1907.11630.