Chapter 1

Overview

"Teleportation" is a magic word, exotic and evocative, but it has been appearing in serious technical literature with increasing frequency. Both theoretically fascinating and experimentally demonstrated, teleportation is the key to quantum networks [GIS 07, KIM 08]. When used in discussions about quantum information, teleportation refers not to Captain Kirk stepping into a machine on the starship Enterprise, dissolving and reappearing on a planet's surface, but to an operation in which a quantum variable dissolves *here* and reappears *there*, on a different physical device. Only the quantum *state* moves; the electron or other physical device remains where it was, and the receiver can in fact be a very different form of physical device than the sender. The quantum state is destroyed at the sender in the process.

Classical networks communicate by physically copying data and transmitting the copy, but the rules of quantum mechanics forbid the creation of independent copies of an unknown, arbitrary quantum state. Instead of risking the loss of valuable, fragile quantum data by directly transmitting our only copy, networks will prepare generic states that are used to teleport data or to perform teleportation-derived operations on the data.

Quantum networks bring new capabilities to communication systems. Quantum physical effects can be used to detect eavesdropping, to improve the shared sensitivity of separated astronomical instruments or to create distributed states that will enable numerical quantum computation over a distance using teleportation. *Quantum communication* is the *exchange of quantum states* over a distance, generally requiring the support of substantial classical communication.

The quantum states that are exchanged may be "standalone" states, an individual element of quantum data. They may also be part of a larger quantum state, spanning

devices or even network nodes in a way no shared classical state can. These latter states we refer to as *entangled* states, which we will study extensively in this book.

Applications running on classical computers will use these quantum states to accomplish one of the above tasks. The classical computer is connected to a quantum device, which may do no more than *measure* the quantum states to find a classical value (such as a bit of a secret key), or may store them for use in more complex quantum computers. A classical computer will treat a quantum computer as a type of coprocessor; likewise, the classical computer will see the quantum network through the eyes of a separate device.

Because quantum data is fragile and some quantum operations are probabilistic, errors and distributed calculations must be managed aggressively and perhaps cooperatively among nodes. Solutions to these problems will have both similarities to and differences from purely classical networks. Architectures for large-scale quantum networking and internetworking are in development, paralleling theoretical and experimental work on physical layers and low-level error management and connection technologies. Unentangled quantum networks have already been deployed, starting in the early 2000s; as of early 2014, entangled networks are not yet deployed, but may appear within the next few years and will form a vibrant research topic in the coming decade.

1.1. Introduction

The motivations for building networks are the same for both quantum and classical networks: the desire to connect people, devices such as computers or sensors, or databases that are in separate locations, for technical, economic, political, logistical, or sometimes purely historical reasons. What differs is the type of data and operation involved. Quantum computers, and quantum networks, use quantum variables rather than classical ones; the analogue of the classical bit is the quantum bit, or *qubit*.

Proper use of quantum information opens up new possibilities, making feasible solutions to some problems that are computationally intractable for classical computers (most famously, factoring large numbers) [SHO 97, LAD 10, NIE 00, VAN 13a] and adding new physical capabilities (most famously, detection of eavesdropping, leading to new, secure, distributed cryptographic key generation mechanisms) [BEN 84]. Other applications for distributed quantum systems include long-baseline optical interferometry for telescopes [GOT 12], high-precision clock synchronization [JOZ 00, CHU 00] and quantum forms of distributed tasks such as leader election [TAN 12] Byzantine agreement [BEN 05a] and coin flipping. Quantum and classical networks and computing systems will hybridize, allowing

applications to select the most efficient mechanism for accomplishing a particular function.

Modern work on quantum communications can be said to have begun with Stephen Wiesner's quantum cryptography proposal, originating around 1970 [WIE 83], followed by Charlie Bennett and Giles Brassard's 1984 proposal for *quantum key distribution* (QKD) [BEN 84, DOD 09], which utilizes the new low-level quantum capability of eavesdropping detection to build a specific system function, namely the creation of shared, secret random numbers for keying of classical cryptographic systems. However, QKD in its basic form is limited in distance to a few hundred kilometers in optical fiber or perhaps more through free space, and is a single-application system.

Bennett *et al.*'s 1993 proposal for *quantum teleportation* made it possible to move data and execute simple calculations remotely, extending the feasible distance for QKD and vastly expanding the range of conceivable distributed quantum applications [BEN 93]. Teleportation involves local quantum operations at each end and classical messages from the sender to the receiver. It consumes a quantum state known as a *Bell pair* (introduced below), shared between the two end points, so, a key function of quantum networks is to replenish the supply of distributed Bell pairs as necessary. As with any physical operation, teleportation operates imperfectly, requiring an extensive system that labors to suppress errors. More than a goal in itself, teleportation serves as a building block for distributed quantum applications.

The need to deal with imperfect quantum states and to span multiple hops spurred the development of the concept of *quantum repeaters* [DÜR 07, SAN 11], which are a vibrant area of research in both experiment and theory. Classical repeaters amplify a signal at the physical level, or receive a weak, distorted or noisy signal then regenerate a clean, strong signal. Quantum repeaters, however, are prevented by the laws of physics from performing such operations directly. Instead, they support high-fidelity, long-distance quantum communication using teleportation over shorter distances and forms of error correction ranging from a simple parity check on a Bell pair to extraordinarily complex, full error correction schemes based on the mathematics of topology. Some repeater architectures manage data movement using computations distributed across all of the nodes in a path between source and destination, while others are more akin to the hop-by-hop packet forwarding used in the Internet; the best approach for a given set of physical capabilities remains an important open question. The basics of teleportation and simple forms of error correction have been experimentally demonstrated, and the race is on to build more complete repeaters.

Although QKD networks using trusted relays and optical switches are in use in medium-scale testbeds, the key architectural issues in large-scale repeater networks are only beginning to be addressed. Protocols to actually implement the repeater

functionality must be developed. Path selection and resource management, both at the node level, where memory resources are precious, and the network level, including choosing who gets access to the network, will play a role in determining whether the networks actually work.

Beyond single networks lies the issue of *internetworking*. An individual network will be built and managed by a single organization. Initially, it will be built using a single quantum networking technology. What happens when we want to bring in a second technology? What happens when we want to connect our network to another organization's network? How do we get them to exchange quantum information? How do we manage the connection between the networks? Such a multi-network configuration is called an *internetwork*, or *internet*, for short. (Spelling it with an uppercase "I", and sometimes attaching the article "the", implies the primary, worldwide classical Internet we all use every day.)

Such an internet, of course, begins with the ability to recode quantum data from one form to another and physically connect heterogeneous technologies. Internetworking will require classical sharing of the correct abstraction for describing quantum states or computation requests and the ability to translate protocols for error management, as well as settling the issues of resource management and path selection.

Our goal, in this book, is to begin from scratch and build an understanding of quantum information, quantum repeaters and classical networking thorough enough to propose and evaluate a quantum internet architecture, including writing the classical software implementing the protocols.

1.2. Quantum information

To understand teleportation and distributed quantum information in principle, only a few concepts are required: superposition, measurement, interference, entanglement, no-signaling and no-cloning. To understand quantum networks in practice, it is equally imperative to study quantum systems in an imperfect world; all of the important behaviors of quantum networks arise from dealing with noise and loss using purification and quantum error correction. The primary mathematical tool for studying algorithms and basic concepts is the *state vector*, and for studying imperfect states, the primary tools are the *density matrix* and the *fidelity*, all of which we will see in the next chapter. Here, we restrict ourselves to a qualitative introduction to the key ideas.

1.2.1. *Principles*

Quantum computers have attracted interest because they are expected to asymptotically outperform classical computers on some important real-world problems [BAC 10, LAD 10, MOS 09, VAN 13a]. These gains in capability arise from the differences in storing and manipulating information using quantum states; here, we will restrict our discussion to qubits, though other forms of quantum information are possible. A qubit may be e.g. the direction of spin of a single electron, the direction of polarization of a single photon, or any of a large number of other proposed state variables. Like a classical bit, a qubit has two states, but unlike a classical bit, a qubit may be in a *weighted superposition* of the two states, allowing certain functions to be evaluated for *both* input values at the same time. A register of n qubits can, like a classical register, hold any of 2^n possible values. The quantum register can in fact hold a superposition of *all* of these values and can, in principle, be used to compute on all 2^n possible states at the same time.

The difficulty lies in extracting useful answers from a quantum computer. To read the results of a computation, dedicated hardware components *measure* the state of the system. The state of the quantum register *collapses* when the system is measured. It randomly picks one state out of the states that are part of the superposition, based on their relative weights. The other states go away, and it is as if they never existed.

A quantum algorithm manipulates the system to *reduce* the probability of undesirable states and *increase* the probability of desirable states, until the system has a high probability of measuring the quantum register and getting an answer to our problem, ideally in substantially fewer computational steps than a classical system would require. This is done by creating *interference* on the quantum states to reinforce good answers.

The concept of *entanglement*, in which the states of two or more quantum subsystems are correlated in a fashion that is not possible in classical systems, is the most difficult quantum concept to grasp. Two qubits can be entangled in a continuous spectrum of possible states; four types of entangled states known as *Bell states* or *Bell pairs* are commonly used. One such Bell state is a superposition of the state where both qubits are 0 and the state where both qubits are 1. In this state, when measured, each qubit has a 50% probability of being found in a 0 state and a 50% probability of being found in a 1 state. However, their probabilities are not independent; both values will be found to be the same.

Bell pairs form the basic communication and computation components for most distributed quantum computation, including teleportation, but are not the only form of entangled state. Bell states can be generalized into multi-party states called GHZ states or W states, and we also use entangled states known as *graph states*. Most distributed quantum computing algorithms will build around one or more of these key

flavors of entangled state; so, the network must be able to create them efficiently. We will see Bell pairs in more mathematical detail in section 2.5, GHZ and W states in section 6.1.2 and graph states in section 6.1.3.

Basic teleportation is accomplished by first creating a Bell pair between the source and destination. The source entangles the qubit to be teleported with the source's half of the Bell pair; then, both qubits are measured, destroying the entanglement of the Bell pair and any superposition state of the data qubit. The measurement results in two *random* classical bits, uncorrelated with the state of the data qubit, which must be transmitted to the destination. Local quantum operations at the destination determined by those classical bits then recreate the original data qubit's state on the remaining Bell pair member. This sequence is illustrated in Figure 1.1. The latency of the classical information transmission prevents information from being transferred faster than the speed of light, and is known as the *no-signaling* constraint, and applies in many situations with quantum information.

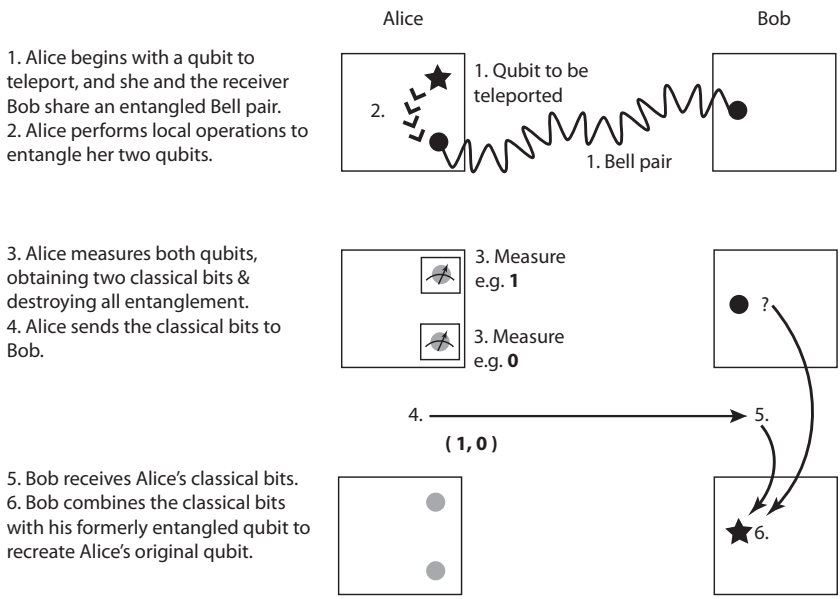


Figure 1.1. *Operations in teleporting a qubit from Alice to Bob*

The final concept required to understand both quantum computation and communication is the *no cloning* theorem, as we will see in more mathematical detail in section 2.6. Perfect *independent* copies of an unknown quantum state cannot be made. "Copies" of some states remain entangled with the original state. This entanglement is actually useful in many quantum algorithms, but an unentangled

copy would be wildly more useful, allowing faster-than-light communication. It would be and is too good to be true.

A major consequence of the no-cloning theorem is that the system cannot copy and send precious quantum data when there is a risk of losing the data; loss of the in-transit copy would destroy even the copy kept due to the effects of entanglement and inadvertent measurement. This fact drives the common quantum networking approach of first building a high-quality, generic entangled state, then using that state to teleport or compute on our valuable data. We turn to handling these imperfections next.

1.2.2. *Imperfect quantum systems*

The central fact of all experimental quantum systems is this: the state of a quantum system is exceedingly fragile. Errors result in continuous degradation of our knowledge about the state of the quantum register. As the state drifts from its assigned value, the probabilities of the zero and one states change and the desired effects of interference may become muted or even incorrect. Beyond these errors that quickly accumulate, isolation of qubits from the environment is difficult, and qubits may be *accidentally* measured, destroying the valuable quantum state.

A measure known as the *fidelity* is one tool for tracking the quality of the state. Fidelity ranges from 0 to 1.0, with the latter being perfect. It is, essentially, the probability that our qubit or set of qubits is actually in the state we believe it ought to be in.

Various techniques for managing errors have been developed, some based on classical error correction and erasure correction techniques, others on uniquely quantum approaches [DEV 13, TER 13]. *Purification*, in which two or more multiqubit states are manipulated to form one higher-fidelity state, uses few quantum memory resources and simple quantum operations, but operates only on well-understood states such as Bell states rather than arbitrary application data. Purification is a type of error *detection*.

More complete protection of an arbitrary quantum state requires *quantum error correction*, in which we use a large number of physical qubits and add redundancy. It is possible to represent more than one qubit in an error correction block, as is done in classical error correction, but holding a single logical qubit is more common. The number of physical qubits can range from tens to possibly thousands, depending on the physical memory lifetime, quantum operation error rates and the performance required to successfully execute a given algorithm.

Besides errors involving the drift of the state, quantum communication systems are also subject to loss in the channel; for those systems expecting to use a single photon,

this loss is fatal for that particular operation. Because losses in optical channels tend to be high, any communication system must be designed to manage this loss. Quantum optical states cannot be simply amplified without destroying the entanglement and superposition; so, other techniques must be used. Losses in the channel generally force a return message to be used acknowledging success or failure.

1.2.3. *Quantum computers*

Let us take a very short detour to look at quantum computers. After all, quantum networks will have some standalone applications, but a major goal is to use networks to connect computers!

The original concept goes back to the early 1980s, when Richard Feynman suggested that it was possible to simulate one quantum device using another, more efficiently than a classical computer could run such a simulation [FEY 02]. Paul Benioff suggested a quantum Turing machine [BEN 82]. David Deutsch explored some of the ideas behind such machines and proposed the first concrete quantum algorithm [DEU 85, DEU 92]. Seth Lloyd proposed the first plausible implementation of a real quantum computer in 1993 [LLO 93].

Theoretical approaches to organizing a computation using quantum effects include the gate model (similar to Boolean logic circuits), adiabatic quantum computation [AHA 04a, FAR 01], direct (analog) simulation, measurement-based quantum computation [RAU 03] and quantum random walks [AHA 93]. All have similar computational power, though the methods of creating algorithms for them are as different as classical digital and analog computers. To the extent that this difference affects quantum networks, in this book, we assume, and work with, the gate model. Measurement-based QC builds on top of a basic gate model and thus can benefit from the networks we describe here, but the adiabatic and direct models would need a very different form of network.

Peter Shor's 1994 announcement of his algorithm for factoring large numbers on a quantum computer generated huge excitement and an increase in research budgets [SHO 94]. The algorithm can factor composite numbers or take discrete logarithms in time polynomial in the number of bits, whereas the best known classical algorithm is superpolynomial [LEN 03]. Realization of such a speedup would dramatically affect the security of encryption algorithms such as RSA and the Diffie-Hellman key exchange used on the Internet, in e-commerce websites and site-to-site network encryption.

Numerous other algorithms have been developed. Lov Grover showed how to get a polynomial speedup on any combinatoric search problem, and it is known that it is impossible to get an exponential speedup on any arbitrary problem with no known

structure [GRO 96, ZAL 99]. More recent algorithms cover various types of quantum chemistry calculations and simulations [BRO 10, BUL 09, KAS 11, LAN 11], certain classes of linear algebra problems [HAR 09], vector space problems [REG 02], graph problems [MAG 05], algebra [HAL 07], Boolean formula evaluation [AMB 07] and machine learning [LLO 13]. Bacon and van Dam [BAC 10] and Mosca [MOS 09] have published surveys which we recommend.

It is worth noting that the resource consumption of these algorithms is an area of ongoing research; how big, how fast and how accurate does a quantum computer have to be to solve interesting problems correctly [VAN 13a]? Current designs suggest that a computer will have to consist of many millions of qubits in order to apply error correction effectively [JON 12a, THA 06]. Execution times of algorithms on potentially buildable machines are also being investigated, although first-cut answers suggest that some algorithms with apparently attractive characteristics will in fact have dismayingly long run times [CLA 13, CLA 09, JON 12b].

A discussion of complexity classes and their application to quantum computation would fill a book, and we will not attempt to delve into it here. Scott Aaronson's PhD thesis is a good survey [AAR 04]. Key ideas here are again due to Charlie Bennett and to Ethan Bernstein and Aaronson's adviser Umesh Vazirani [BEN 97, BER 97].

All of this would have remained purely an exercise in theory, if not for the development of methods for suppressing errors, as discussed in the last section. John Preskill, Peter Shor, Andrew Steane, Charlie Bennett, Manny Knill and others contributed key insights to fault tolerant operation of a quantum computer [BEN 96c, KNI 96, PRE 989b, SHO 95, STE 96]. Excellent surveys on this topic have proliferated in the last few years [DEV 13, GRA 09, RAU 12, TER 13].

For a non-mathematical treatment of the ideas, the book by Williams and Clearwater is excellent [WIL 99]. Nielsen and Chuang [NIE 00] is the canonical text and covers algorithms as well as the underlying technology. The collection edited by Bouwmeester, Ekert and Zeilinger in 2000 [BOU 00] remains an excellent introduction to the technology.

1.2.4. *Applications of distributed quantum information*

The concepts of quantum communication are inherently fascinating, worthy of basic research by anyone's definition. However, as engineers striving to build networks, we must understand how the networks will be used, in order to evaluate our design decisions. Moreover, systems will only be deployed in large numbers when a compelling economic case appears. Thus, the study of quantum networks involves an equal measure of studying applications for distributed quantum states.

Earlier, we introduced QKD as an application. Implementations of QKD are well beyond the experimental phase [ELL 03, DOD 09]. A few commercial products are available, and metropolitan-area testbed networks exist in Boston, Vienna, Geneva, Barcelona, Durban, Tokyo, several sites in China and elsewhere throughout the world. In fact, the BB84 technique deployed in most links in these networks does not use entangled quantum states, although another approach, developed by Artur Ekert, does [EKE 91]. QKD is certainly the most practical, commercially attractive use of quantum networks in the near term. QKD has been integrated into custom encryption suites and the Internet standard IPsec suite and has been proposed for use with the TLS protocol common on the World Wide Web [ELL 02, MIN 09, NAG 09].

Other security-related functions have been proposed, including leader election and Byzantine agreement under assumptions of very powerful adversaries [BEN 05a, TAN 12]. Executing these algorithms would require nodes with more functionality than the ability to measure qubits for QKD, but likely would not require a fully functional, large-scale quantum computer.

We can reason that, like classical systems, one quantum computer is useful, but two are even more so, and connecting them together brings immediate benefits. Especially given that quantum algorithms (such as Shor's algorithm for factoring large numbers) are security-related, it seems reasonable to suppose that clients would like to be able to use remote quantum servers securely. A form of computation known as *blind computation* would allow a client to use the services of a remote machine, without revealing the algorithm, input or output data [BRO 09]. This will require *very* high rates of teleportation, low residual error rates and a powerful server; various schemes proposed alter the demands made of the client [MOR 13].

We can view QKD as a type of sensor network in which the interaction between the physical world and our quantum information devices figures prominently. Even more directly, distributed quantum states can be used as a form of *reference frame*, so that physical measurements can be conducted over a distance, more accurately or efficiently than using purely classical means. For example, synchronization of clocks is a common, critical use of communication signals and quantum algorithms have been proposed that will converge with asymptotically fewer operations than a classical method requires [JOZ 00, CHU 00]. A mechanism for improving the resolution of optical interferometry for astronomy has been proposed [GOT 12]. All of these will be very demanding applications with respect to both Bell pair production rates and the precision of those states.

1.3. Quantum repeaters

Quantum networks, like classical networks, will involve nodes and links and a layered communication architecture with individual protocol modules

communicating vertically up and down a protocol stack and horizontally with peers. This section focuses first on the physical components that make up a link, before the discussion moves to arranging multiple links into a chain, then a network.

1.3.1. *Physical communication technologies*

Quantum communication channels are implemented by sending states of light down a physical channel. These states may be single photons, or other quantum optical states with either large or small numbers of photons. A channel may be a waveguide such as an optical fiber, or free space. It may involve a single transmitter and receiver, or multiple receivers that can individually be enabled or disabled in a shared bus configuration. A link uses a quantum channel and associated classical channel to connect two or more nodes.

A node may have quantum memory that can be used to store a qubit that is entangled with the pulse as it is sent out. When receiving a pulse, a node may either directly measure the pulse using, for example, an avalanche photodiode (APD), or may transfer its quantum state to a memory for later use or analysis. The pulses may come from weak lasers, flourescing atoms, or emission of single photons from a quantum dot, a structure created to exhibit some of the behavior of an atom.

One of the most promising hardware approaches for entangled networks uses microscopic pieces of diamond. When a carbon atom in the diamond lattice is replaced with a nitrogen atom, a positive electrical potential in the lattice capable of trapping a single electron is created. This approach, called *nitrogen vacancy (NV) centers in diamond*, may work at room temperatures, in contrast to most other solid-state quantum systems, which require cryogenic temperatures. Other promising experimental approaches include various forms of quantum dots. Ion traps that hold individual atoms in a vacuum are perhaps the most experimentally advanced approach. Entanglement of up to fourteen qubits in a single trap has been accomplished.

All of these experimental approaches have drawbacks; most do not operate at telecom wavelengths, which will dramatically shorten feasible link distances, though wavelength conversion schemes are also under development. They suffer from short memory lifetimes to differing extents, and the probability of correctly transferring the optical state to the static qubit remains inadequate for reasons ranging from low optical coupling efficiency to basic physics. None of these approaches is ready for mass production, and currently, all require hand-tuning and complex experimental setups.

The necessary classical messages include heralding at the physical layer to coordinate timing of the quantum pulses and many messages for coordinating the

higher-level error management, data movement, distributed state creation and application functionality. Researchers often assume that the classical messages follow the same path through a network as the quantum messages, though except for the physical herald, this is not strictly necessary. When the classical messaging uses a different network topology, analysis of the communication efficiency must be done with care.

1.3.2. *Multi-hop Bell pairs: quantum communication sessions*

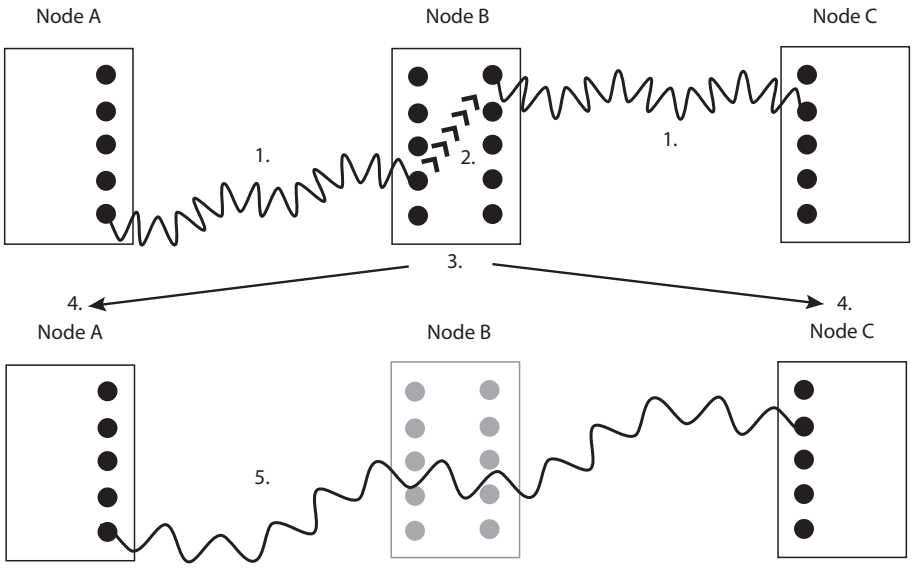
The purpose of the technologies just described is to create link-level entanglement. Interesting communication requires extending that entanglement across multiple hops while maintaining adequate fidelity. The *quantum repeater*, building on basic entanglement functionality with purification and teleportation, lays the foundation for quantum networks. Here, we discuss direct transfer of quantum information, and the generation of long-distance Bell pairs over a fixed chain of links and nodes. Below, we will take up the more general question of how such a chain can be part of an actual network.

The most obvious method of moving quantum data from place to place is direct hop-by-hop transmission by transferring the qubit state onto a photon and firing that photon down the link toward a node at the far end. If that link does not reach all the way to the destination, then it is received and forwarded on to the next node, relay fashion. However, as noted above, this places the valuable quantum data at unacceptable risk of loss. Instead, we could build Bell pairs over each link and teleport our qubit one hop at a time. Disappointingly, hop-by-hop teleportation is only marginally better than direct transmission because each hop degrades the fidelity of the data qubit.

Alternatively, what about creating a Bell pair at the source and performing hop-by-hop teleportation not on our valuable data qubit, but on one of the two Bell qubits instead? This will extend the length of the Bell pair, as shown in Figure 1.2. If the qubit being sent is lost before reaching the destination, the Bell pair can be discarded and restarted. Once the qubit reaches the destination, the Bell pair can be used to teleport the important data qubit, without fear of loss. However, the same problem arises: the fidelity of the Bell pair degrades with each teleportation operation as well as over time if the system keeps the qubit in memory.

One solution is executing purification in a distributed fashion, as in Figure 1.3. When purifying Bell pairs, node A holds one half of each of two pairs, and node B holds the other half of each. Using local quantum operations, including measurement of one of the Bell pairs, A and B can probabilistically improve the fidelity of the other Bell pair. Note that, because purification does not require direct quantum communication, it can operate over any distance, provided the requisite Bell pairs and a classical communication channel are available. The biggest drawback to

distributed purification is that it requires that each end convey the results of its local measurement to the far end. Assuming that both nodes can independently identify the set of operations to perform, the minimum time for completion of purification is the one-way classical message latency between the two nodes.



1. Nodes begin with two entangled pairs, AB and BC.
2. Node B selects pairs to teleport, performs local operations, measures one qubit of each pair.
3. B communicates measurement results and new entanglement status to A and C.
4. Receive partner's measurement result and new entanglement status, including node/qubit addresses.
5. Result is single lower-fidelity, longer-distance Bell pair.

Figure 1.2. *Teleportation can lengthen one Bell pair using another*

The one-hop-at-a-time extend-purify-extend approach will work, but fails to take full advantage of the fact that *the distributed states being created are generic*, which allows the network to effectively build the Bell pair in parallel. The network can choose to build from both ends of the needed connection, or from the middle; note that the teleportation operation shown in Figure 1.2 operates independently of the length of the two Bell pairs. In the late 1990s, Wolfgang Dür, Hans Briegel and their collaborators proposed "nesting" Bell pair purification and teleportation so that the length of entanglement *doubles* in each round, allowing a logarithmic-depth number of rounds to create end-to-end entanglement over a large number of short hops: eight one-hop Bell pairs become four two-hop Bell pairs, then two four-hop Bell pairs and

finally one eight-hop Bell pair [DÜR 99]. This has become the benchmark approach to repeaters, with much research assuming a power of two number of hops.

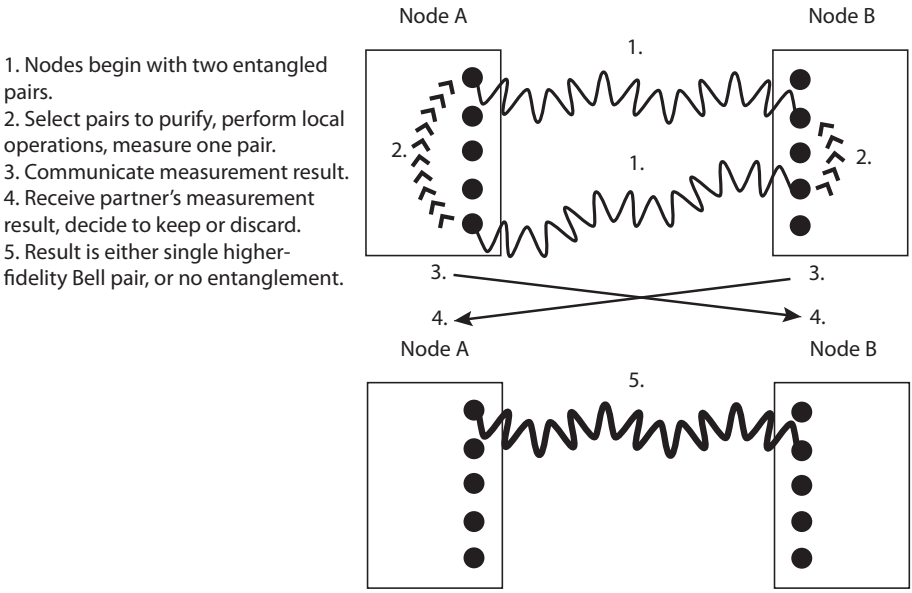


Figure 1.3. *Steps involved in purification of Bell pairs*

This purify-and-teleport architecture is not the only known approach; quantum error correction (QEC) can replace purification. Managed properly, QEC protects the data more completely, reducing the need for multi-hop purification and its associated need for round-trip delays. These advantages come at the expense of substantially more memory and computation resources at each node. Overall, in this book, we will study five approaches to connecting a source to a destination via a chain of links and nodes.

Layered communication describes how protocol functions are vertically composed within a communications node to provide increasingly complex capabilities. Layered quantum communication relies on five key functions that are unique to quantum networks, but only the first of these is actual quantum communication; the rest are classical functions for managing quantum states.

– *Physical layer*: we rely on a quantum physical layer using light to encode quantum state. Many technologies for this layer are under development.

– *Link-level entanglement*: because most physical entanglement mechanisms are probabilistic, the link layer will include an acknowledgment to the sender indicating which attempts succeeded.

– *Remote state composition*: in the Internet, links are composed into paths by copying packets from one link to the next. In a quantum network, links are less readily composed due to the no-cloning theorem. Quantum paths thus either establish end-to-end entanglement from entangled links or use that entanglement to teleport the quantum state from one end to the other. This layer is very sensitive to the link-layer capabilities as well as the error management mechanism.

– *Error management*: in the classical Internet, errors are managed using redundancy (e.g. forward error correction) or error detection and retransmission. As noted earlier, the no-cloning theorem prevents straightforward use of either of these mechanisms. The fidelity of quantum states is critical in reducing the need for error management.

– *Application*: the application may be QKD, or a physical reference frame for an instrument, or a numeric computation or decision algorithm based on shared state. The application will determine if end-to-end entanglement is required, or if the QKD-like model of direct measurement is adequate. Some applications may desire quantum states other than Bell pairs, including any of several common forms of three-party or larger states, such as the W and GHZ states we mentioned above.

One of the most difficult tasks in quantum networking is for the software running in separate nodes to maintain an adequately consistent idea of the density matrix of a quantum state that spans two or more nodes. The fidelity of the state will change, depending on the quality of the hardware at each node and the actions taken by the node. Unless the nodes exchange information about such matters, it is impossible to accurately assess the current state and the need for further action such as error management.

1.4. Network architectures

The physical entanglement, error management and communication session technologies we have discussed will get us to laboratory-scale demonstrations, but do not form a complete, deployable network architecture. Real-world networks must accommodate heterogeneous links arranged in complex topologies, managed by many autonomous organizations; traffic sources competing for use of the network; and ongoing network events such as a node or link suddenly becoming unavailable. An architecture for a large-scale network must support independent decision making by the nodes in a manner that will result in robust, efficient operation of the network as a whole. Although classical design principles can be applied to quantum networking, the resulting architectures can be quite different due to the radical restrictions and unique capabilities of the quantum domain.

The chosen communication session architecture will be executed over a *path* composed of links and nodes. Each node, link and software service, and even the

quantum states themselves, must have an *identifier*, a name that software can use when composing those paths and tracking states as they are fabricated. Because the global network does not exist solely for the exclusive use of one communication session, a *resource management discipline*, either explicit or implicit, will govern. All of these interlocking aspects of the architecture depend on an understanding of the *semantics* of the requests on the network; so, let us begin there.

1.4.1. *Semantics of distributed quantum information*

Before constructing actual networks, we must decide how we want nodes to communicate across a chain of links. Above, we assumed that our goal was to move quantum data from place to place, either directly or by building Bell pairs. Now that we have some idea of what is in our quantum toolbox, we can ask a sophisticated question: what should the semantics of a request across the network be?

The request model depends first on what the network is designed to *do*. In classical networks, traditionally the network layer only sends data, via unicast, multicast or broadcast, with other functions delegated to higher-layer protocols. To support the applications we have discussed, a quantum network can operate in one of three modes: (a) it can teleport data from place to place, (b) it can execute certain computational operations over a distance (a technique known as *teleporting gates*) or (c) it can create distributed quantum states. Each of these options results in a different form of contract between the requesting end node and the network.

Perhaps the most fundamental operation, at the network layer, would be creation of high-fidelity distributed Bell pairs, which alone are adequate for building more complex distributed states, teleporting data or executing remote operations. Conservative engineering practice would suggest that, as with operating system APIs and IP packet semantics, simple is best, favoring this as the lone network-supplied operation. However, data movement as a primitive may improve performance by operating more asynchronously. Providing remote computation requests, or a richer set of state-creation services in the network, may reduce application complexity or improve overall system performance by reducing the total number of operations that must be performed. With appropriate network protocols, all three modes can be mixed in the same network.

Above, we noted that most of the functionality in a quantum network is actually classical. The same is true of the applications that consume the services provided by the network, whether QKD, or distributed digital computation, or use as a reference frame in instrumentation. Perhaps the most commonly-used approach in software for sending or receiving data over a network is the *sockets* interface, developed at UC Berkeley in the early 1980s [WRI 95a, WRI 95b]. Software engineers studying quantum networks have begun to ask, “What does a quantum socket look like?” The

answer to this question will, of course, be different depending on which of the above options is chosen.

1.4.2. *Identifiers*

Because we must identify where we want to send data, networks naturally require names for the nodes. These names may be symbolic and easy for humans to remember or numeric *addresses* that are used more directly by internal systems and help to guide the choice of *route* to the destination.

On the Internet, end points of the communication must be able to name each other, but in general know nothing about the other nodes involved in completing their communications. For quantum communication, end nodes using a purify-and-swap approach must communicate directly with nodes along the path and so must be able to name and discover those nodes.

On the Internet, a node can generally determine where to next send a packet using only the destination address (a type of name) carried in the packet itself. When the packet arrives at the destination, further information carried in the packet (other types of names) is used to distinguish which software program (process) is waiting for the packet. In other types of networks, this discrimination can be done either explicitly or implicitly.

In quantum networks, the entangled states built within the network must have some sort of identifier, to facilitate the software work of management and delivery to applications. This is complicated by the fact that a Bell pair in the middle of the network might not yet be assigned to serve a particular end-to-end session.

1.4.3. *Paths*

Multihop networks require a means of selecting a path through the network [PER 00]. The need to understand the network topology manifests itself both in the choice of path through the static, physical topology, and in the dynamic operation of the network, as in Figure 1.5. This is a complex operation in networks of thousands of nodes and links with as much as eight orders of magnitude difference in bandwidth, made vastly more difficult when tens of thousands of such networks are interconnected.

On the Internet, each network is run by an independent organization, and the internal structure of each of those networks is kept hidden from the outside. Path selection is therefore a two-level process, enhancing scalability and autonomy. Moreover, calculation of the best route from a given node to anywhere else is an

ongoing task, but a complete path for a given Internet connection is not determined before data is sent. Information held locally allows globally consistent decisions to be made for each data packet independently inside each node.

One common approach to path calculation in medium-scale networks is a distributed form of Dijkstra's shortest path first algorithm [DIJ 59, GOV 02, MOY 97]. From network to network, a separate protocol is used to minimize the number of networks that a packet passes through.

Quantum repeater networks and internetworks will undoubtedly be organized in a similar fashion. Dijkstra's algorithm can be adapted to medium-scale repeater networks [VAN 13b], but no concrete proposal yet exists for routing at the global level.

The path selection mechanism will be affected by the choice of communication session architecture and in turn makes demands of the identifier and naming architecture. In the purify-and-swap approach, nodes in a path must communicate with more than just their immediate neighbors, as in Figure 1.4. In its most natural form, this requires that any node be able to discover and name any other node in the entire internetwork, but we have just seen that a network typically keeps its own internal structure hidden from the outside. A scalable, robust solution to this problem is imperative.

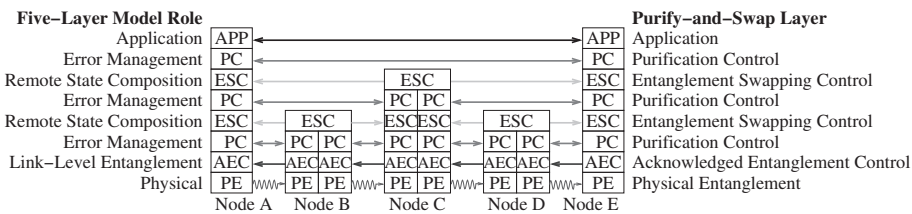


Figure 1.4. *Protocol layers and their interaction in purify-and-swap repeaters, in a five-node, four-hop chain. The labels on the left indicate the model layer represented, and the labels in the boxes and on the right indicate the protocol name for purify-and-swap repeaters. Double-headed arrows indicate bidirectional classical communication is required. The only quantum portion of the stack is the physical layer, shown with all links propagating left to right*

1.4.4. *Resource management discipline*

The network architecture must specify resource management for data requests: are the qubit memories in a node and the Bell pairs created across a link committed to the exclusive use of a single quantum communication session, or are they shared? This choice is affected by the communication session architecture, and in turn affects the

construction of network paths and the naming of quantum states that currently span more than one node.

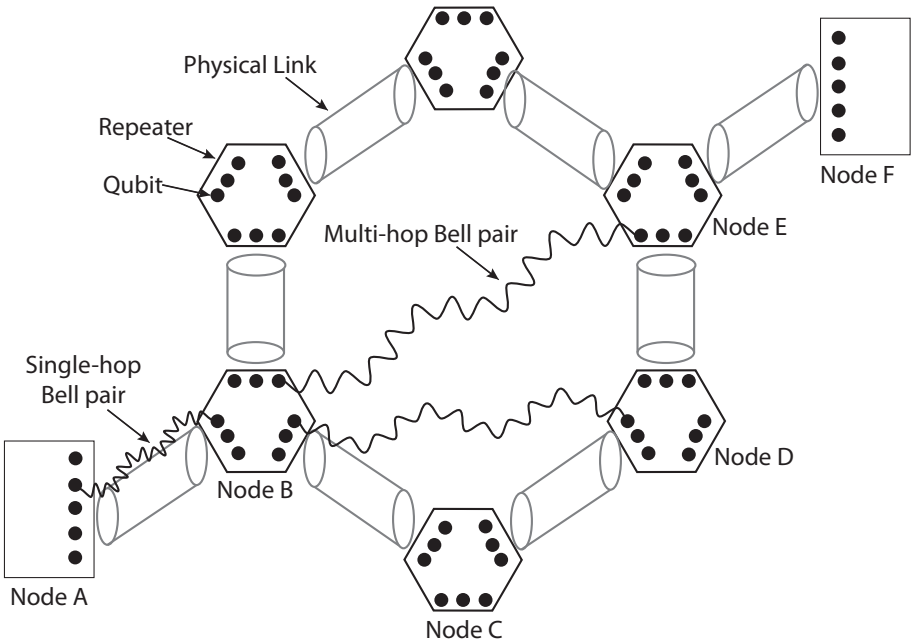


Figure 1.5. *Even when Node B knows that A is trying to build a Bell pair with F, B may be uncertain whether its Bell pair connected to Node D or Node E is "closer" to the destination*

Completely dedicating all of the resources along a path to a single session is the most obvious approach, especially as fragile quantum memories impose fairly stringent real-time constraints. However, this approach blocks other connections from using the same links and may reduce the overall rate at which work is completed across the network. Our recent research suggests that sharing the resources using one of several forms of *multiplexing* may raise the aggregate throughput of the network if implemented carefully.

The generic states generated throughout the network could serve the needs of many pending requests, much as production from a factory can be sent to serve any of a large number of customers. An interesting dynamic resource assignment problem arises: does an entangled Bell pair between two nodes "belong" to a specific end-to-end request, or are all Bell pairs "up for grabs"? Bell pairs that span long distances along routes carrying a lot of traffic may be especially desirable commodities.

1.4.5. *A quantum internet*

We have discussed some of the difficult problems in networks as they scale up in the number of nodes and links: heterogeneity, autonomy of management, naming, resource management and path selection and control. Distributed management of density matrices requires complex protocols and real-time decisions based on inevitably out-of-date information. The choice of quantum communication session architecture, or the design decision to allow nodes to choose one of several session architectures, affects all of these architectural choices.

All of these problems are in turn made more difficult in an internetwork. My research group and our collaborators have begun developing a quantum internetwork architecture called quantum recursive network architecture (QRNA) [VAN 11], which we hope will provide a structure for addressing these problems at global scale. QRNA may or may not ultimately become the architecture for the Quantum Internet, but we will present it in some detail in Chapter 15 to provide a platform for discussing the issues in a concrete manner.

QRNA is based on the classical recursive network architecture (RNA) developed by Joe Touch and collaborators [TOU 08]. A recursive network architecture can be viewed as a natural fit for quantum repeater internetworks. The naming structure, network topology, path composition and even the creation of multi-hop entangled states can all be simplified by judicious use of recursion.

QRNA provides a general-purpose request mechanism abstracted from underlying layers to accommodate any of the communication session architectures presented above. It supports requests for creation of distributed states (including both two-party Bell pairs and multi-party states) and operations on those states. Requests may be recursively decomposed and distributed throughout the network in order to build the end-to-end state requested by an application, meshing smoothly with the protocol layering model in Figure 1.4.

Like the Internet, nodes need not understand the topology of the entire network or even the names of all of the nodes involved in a communication session. This is accomplished by allowing a link as seen by a node to be either a physical link or a recursively organized network.

1.5. Conclusions

Quantum networks come in both entangled and unentangled forms. QKD networks are already up and running in various metropolitan areas throughout the world, although the coupling at intermediate nodes is strictly classical at the moment. In contrast, entangled networks remain rudimentary, existing only at small scales in

laboratories and have yet to demonstrate all necessary functionality in a single experiment [SAN 11].

For entangled networks, by far, the most important ongoing research is on the physical layer – if quantum memories and local operations do not reach sufficiently high fidelities and entanglement success probabilities do not rise, quantum networks will remain a laboratory exercise. Applications for distributed quantum states, whether numerical computation or sensor networks, will drive the need for quantum networks; without them, no one will buy and deploy quantum networking equipment. Both of these areas are being addressed in depth, the first by experimental physicists, the second by theorists in both computer science and physics. It bears pointing out that the performance required for some of these applications remains several orders of magnitude beyond even optimistic hardware predictions for the next several years.

To bridge the considerable gap between theoretical large-scale, wide-area applications and small-scale experiments, an overarching network architecture and matching protocols must be developed. These protocols must emphasize optimized use of quantum memory, both spatially, by reducing the number of qubits that must be stored, and temporally, by reducing the length of time a qubit must be held in an intermediate state, e.g. by eliminating round-trip messaging where possible. The real-time factors in physical memory decay, or the high resource requirements of error correction-protected memories, must be managed properly. Moreover, the form of requests within the network is critical to efficiency. Improvements in the request model can alter the demands on the size, quality and capabilities of the physical system. Ultimately, the problems exemplified in Figures 1.5 and 10.6 are matters of giving each node enough information to make high-quality, autonomous decisions. The design of robust, efficient classical protocols usable in multi-user, multi-technology quantum internetworks will demand the technical skills of the data networking community.

The Quantum Internet, once realized, will allow us to exploit entanglement over long distances for new computational capabilities and for new physical capabilities such as eavesdropping detection. More speculatively, we can imagine uses such as sensor networks for quantum-enhanced telescopes and tests of the fundamental correctness of quantum mechanics. Within a few years, quantum networking and teleportation will move out of the physics laboratory and into the network engineering domain, offering some of the most exciting and intellectually challenging research and development topics of the coming decade.