Computer Science Review 57 (2025) 100747

Contents lists available at ScienceDirect

Computer Science Review

journal homepage: www.elsevier.com/locate/cosrev

Review article

Review of Distributed Quantum Computing: From single QPU to High Performance Quantum Computing

David Barral [a], F. Javier Cardama [b], Guillermo Díaz-Camacho [a], Daniel Faílde [a], Iago F. Llovo [a], Mariamo Mussa-Juane [a], Jorge Vázquez-Pérez [a,b], Juan Villasuso [a], César Piñeiro [c], Natalia Costas [a], Juan C. Pichel [b,c], Tomás F. Pena [b,c,*], Andrés Gómez [a]

[a] Galicia Supercomputing Center (CESGA), Avda. de Vigo S/N, Santiago de Compostela, 15705, Spain

[b] Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela, Santiago de Compostela, 15782, Spain

[c] Departamento de Electrónica e Computación, Universidade de Santiago de Compostela, Santiago de Compostela, 15782, Spain

ARTICLE INFO

Keywords:
Distributed quantum computing
High-performance computing
Teleportation
Quantum networks
Distributed quantum compilers
Circuit knitting
Distributed quantum applications

ABSTRACT

The emerging field of quantum computing has shown it might change how we process information by using the unique principles of quantum mechanics. As researchers continue to push the boundaries of quantum technologies to unprecedented levels, distributed quantum computing raises as an obvious path to explore with the aim of boosting the computational power of current quantum systems. This paper presents a comprehensive survey of the current state of the art in the distributed quantum computing field, exploring its foundational principles, landscape of achievements, challenges, and promising directions for further research. From quantum communication protocols to entanglement-based distributed algorithms, each aspect contributes to the mosaic of distributed quantum computing, making it an attractive approach to address the limitations of classical computing. Our objective is to offer a comprehensive review that serves both experts in the field and researchers or enthusiasts in quantum computing looking for a starting point to explore the area of distributed quantum computing.

Contents

---

[*] Correspondence to: Centro Singular de Investigación en Tecnoloxías Intelixentes (CiTIUS), Universidade de Santiago de Compostela, 15782 Santiago de Compostela, Spain.

E-mail address: tf.pena@usc.es (T.F. Pena).

https://doi.org/10.1016/j.cosrev.2025.100747
Received 10 April 2024; Received in revised form 24 February 2025; Accepted 9 March 2025

Available online 24 March 2025
1574-0137/© 2025 The Authors. Published by Elsevier Inc. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

In pursuing superior computational abilities, quantum computing has emerged as a promising frontier with huge potential. While individual quantum systems have shown impressive capabilities, distributed quantum computing introduces a new approach that could vastly increase computational power. This study aims to explore in depth the current landscape of DQC, also known in certain literature as modular quantum computation, from physical devices and interconnection networks to distributed algorithms. In this review, we will analyze the different solutions proposed and the challenges posed by this rapidly advancing field.

As we examine distributed quantum systems more closely, it becomes clear that collaborative and interconnected quantum processors are essential for overcoming the constraints faced by standalone systems. Problems of both fundamental origin – decoherence, dissipation, and crosstalk – and practical origin – processor topology, cabling, connectors, and control electronics – hinder the fabrication of ultra-large Quantum Processing Units (QPUs) [1]. It is thus foreseeable in the short term that quantum computers will not scale in a local device with a large number of qubits in a single quantum processor. A distributed infrastructure with several quantum processors that contain a limited number of qubits could overcome this difficulty. In fact, there is a clear consensus among leading academic and industry stakeholders that the practical realization of large-scale quantum processors should adopt a distributed approach based on clusters of small, modular quantum chips within a network infrastructure, with classical and/or quantum communications [2–4]. QPUs are intended to be seamlessly integrated into a classical High-Perfomance Computing (HPC) infrastructure, alongside CPUs, GPUs, and other hardware accelerators [5–9]. This integration allows for their utilization in collaboration within a shared development environment, leading to what is already called quantum-centric supercomputing centers [10].

As an example of this trend, IBM recently unveiled Quantum System Two [11], a modular architecture that will serve as the basis for building their new quantum-centric HPC infrastructures. The model unveiled features three IBM Quantum Heron processors, each with 133 fixed-frequency qubits and tunable couplers. According to IBM, Heron yields a 3-5x improvement in performance with respect to the previous 127-qubit Eagle processor, virtually eliminating crosstalk.

However, the interest in DQC is not new. We have to go back to the end of the 20th century to find the first works that analyzed the possibility of using non-local effects to perform distributed computing [12,13]. This interest grew after Cirac et al.'s work, where it was shown that DQC is superior to classical computing for the phase estimation problem even under non-ideal conditions [14]. Shortly after, Eisert et al. [15] and Collins et al. [16] took a step forward, introducing resource-optimized protocols for non-local quantum gates necessary to move from specific problems like phase estimation to universal quantum computing. At the same time, DiVincenzo [17] included, in his famous criteria for a quantum computer, two additional not-so-well-known items related to DQC and the interconnection of QPUs: the ability to interconnect stationary and flying qubits and to transmit flying qubits between specified locations faithfully.

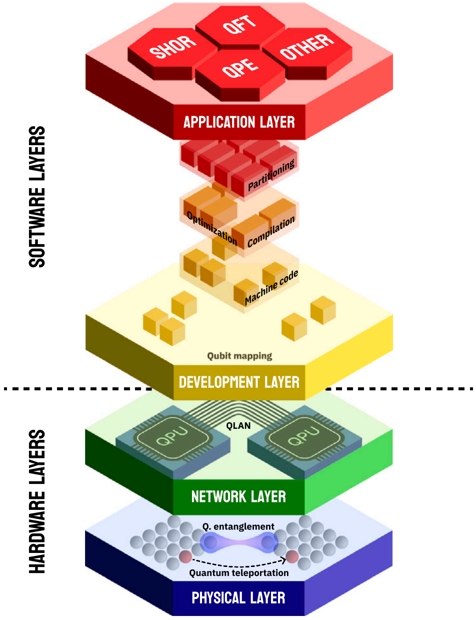


Fig. 1. Layered model for distributed quantum computing.

After the first theoretical studies on the feasibility of DQC, a series of proposals for experimental realizations began to appear gradually [18–21]. At the same time, several interesting developments regarding DQC algorithms were made, such as the distributed versions of the Grover and Shor algorithms [22,23]. The first taxonomy of DQC systems was proposed by Yepez [24] in the early 2000s, where two types of systems were described: those with entanglement between nodes, called type-I, and those with only inter-node classical communication, called type-II. Jozsa and Linden later demonstrated that a type-II quantum computer cannot achieve exponential speedup when the computation requires entanglement across the full set of qubits [25].

Considering these initial works as a starting point, this review extensively examines the current advancements in the field of DQC, extending and updating previous surveys on this subject. The previous[1] survey by Caleffi et al. [26] provides a comprehensive overview of DQC archetypes. It covers a range of configurations, from single QPU with multi-core execution to single datacenter with multiple

---

[1] During the review process of this article, we were notified that an updated version of the Caleffi et al. survey available in arXiv was published in the Computer Networks journal.

QPUs and even multi-farm networks, also comparing them to their classical counterparts. Additionally, it offers a detailed section on simulation tools for DQC development, organized into three distinct layers: hardware-oriented, protocol-oriented, and application-oriented.

Our review is specifically focused on the integration of HPC with DQC, and we excluded multi-farm networks due to the substantial latency introduced by geographically separated centers. Instead, we concentrate on multi-core and multi-QPU configurations, drawing parallels with classical HPC paradigms. For example, we explore techniques such as circuit cutting and embarrassingly parallel distribution of circuits, whenever fully realized quantum networks may not yet be available. Additionally, we emphasize the critical importance of the physical layer in DQC systems.

To facilitate the readers' understanding, this survey is structured according to a layered model, as depicted in Fig. 1, similar to the full-stack architecture presented by [27], the abstract model in [28], or the DQC simulator tools structure in [26].

The two lower layers in Fig. 1 encompass the hardware developments needed to implement a distributed quantum system and would be equivalent to the three lower layers of the classical OSI model. So, the physical layer refers to the mechanisms that allow two physically separated QPUs to be connected, while the network layer defines how to establish communication between multiple QPUs. Directly above this layer, we discuss advances in development tools that allow applications to be distributed and executed on a distributed quantum system, including partitioning, compilation, optimization, and mapping algorithms. Finally, in the uppermost layer, we address distributed algorithms. It is important to note that these layers are interdependent, with each layer influencing those immediately preceding and succeeding. For instance, the development of a compiler is influenced by the underlying hardware and provides support for different partitioning techniques in the application layer.

Following this structure, the review is organized as follows. Section 2 describes the available quantum mechanical tools to transmit quantum information. In Section 3, we present proposals oriented to creating networks interconnecting multiple QPUs. Next, Section 4 discusses solutions that allow applications to run in distributed environments, including partitioning, distribution, compilation, and mapping techniques. Section 5 presents different proposals for applications running in these environments. We will end the paper with a summary of the current state of the art and open lines in the field.

2. Physical layer for distributed quantum computing

DQC aims at performing arbitrary computational tasks between unknown quantum states at the distant nodes of a quantum network. These networks, identically to their classical counterparts, coordinate and distribute information across devices. However, quantum networks have multiple features and limitations that make these tasks difficult, primarily arising from the *no-cloning* theorem: arbitrary quantum states cannot be *perfectly* copied; therefore, quantum information cannot be replicated and broadcast [29]. Fortunately, the properties of quantum systems can be exploited in a way that allows us to circumvent this impediment and reliably transmit quantum information or control quantum systems remotely. This section will briefly describe which quantum mechanical tools are available for this purpose.

First and foremost, the physical resource that enables performing non-local computation is *entanglement*, a unique correlation of joint quantum systems stronger than any classical counterpart but very fragile, hard to create and to maintain long. Entanglement lies at the heart of quantum communications, facilitating the distribution of quantum states encoding quantum information through a protocol known as *quantum teleportation* or *teledata*. Multiple teleportation variants exist, which are designed to either transmit data in one direction – *quantum teleportation* or *teledata* – but also bi-directional communication – *entanglement swapping* – and gate operation at a distance – *gate teleportation* or *telegate*. Furthermore, the basic two-node teleportation can be extended to multi-party distribution networks composed of many nodes. Some parties may either help the rest of the network in the quantum communication protocol – *assisted teleportation* –, or the quantum information may be imperfectly broadcast from one sender to the rest – *quantum telecloning*.

In the following sections, we will introduce these protocols in detail.

2.1. Quantum entanglement

Entanglement is the property of a quantum system that illustrates the impossibility of describing a composed system in terms of just its individual components due to nonclassical correlations of certain degree(s) of freedom of the subsystems [30]. Typical examples of these degrees of freedom are the position and momentum of free particles, the polarization of light, energy levels of trapped ions, or transverse atomic spins. These degrees of freedom are related to observables that present a discrete and finite spectrum or a continuous and infinite one. Hence, the terms discrete variable (DV) and continuos variable (CV). This review focuses on DV because it is the most common in quantum computing.

Archetypical examples of DV entangled quantum states are the pure states

$$\begin{aligned} |\Phi^{\pm}\rangle &= \frac{1}{\sqrt{2}}\left(|0\rangle_A|0\rangle_B \pm |1\rangle_A|1\rangle_B\right), \\ |\Psi^{\pm}\rangle &= \frac{1}{\sqrt{2}}\left(|0\rangle_A|1\rangle_B \pm |1\rangle_A|0\rangle_B\right), \end{aligned} \tag{1}$$

dubbed Bell states or Einstein, Podolski and Rosen (EPR) pairs, where two parties – Alice and Bob – share two qubits A and B encoded in a dichotomic degree of freedom as polarization, spin, or any other two-level quantum variable [31]. A perfect non-local correlation arises as Alice's measurement outcome determines Bob's measurement outcome. This property allows us to build an intuition of how Bell states are a natural choice for quantum communication: if a quantum gate, whose matrix representation is symmetric, is applied to one of the qubits of the Bell state $|\Phi^+\rangle$, it is the same as if the gate was applied to the other qubit. The gate somewhat 'slides' between qubits through the entanglement, like beads on a string [32].

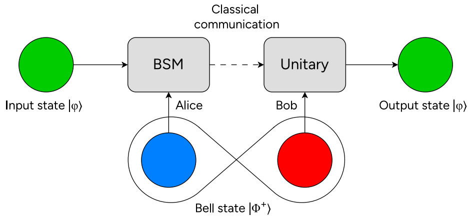
These entangled states are the basis of a large number of quantum information protocols, one of which is quantum teleportation, which we introduce in the following section.

2.2. Quantum teleportation or teledata

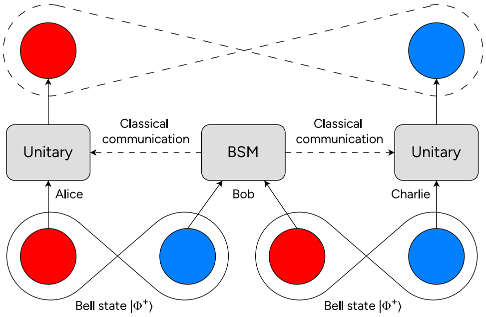
Quantum teleportation, one of the more remarkable quantum information protocols, was introduced thirty years ago in a landmark paper [33]. This quantum protocol enables the reconstruction of an unknown quantum state of a given physical system at a different location without actually transmitting the system. Quantum teleportation requires two key ingredients: Quantum entanglement and classical communication between the locations (which excludes superluminal communication).

Quantum teleportation plays a pivotal role in the development of quantum technologies [34]. It overcomes some of the limitations of quantum communications and quantum computing using the non-local transfer of unknown information. Quantum teleportation networks [35], entanglement swapping [36], and quantum repeaters [37] enable the distribution of entanglement over long distances [38], while quantum gate teleportation [39] and measurement-based quantum computing [40] are examples of techniques that distribute local gate operations among physically disconnected parties [41]. Entanglement swapping and gate teleportation will be discussed further in the next section.

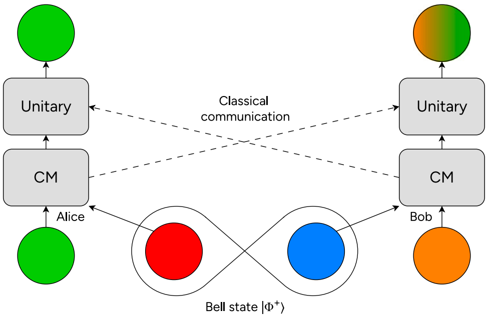
Proof-of-principle demonstrations of quantum teleportation were successfully achieved using diverse physical substrates as photonic



(a) Quantum teleportation.



(b) Entanglement swapping.



(c) Gate teleportation.

Fig. 2. Sketch of quantum communication protocols: (a) Quantum-state teleportation (teledata), (b) entanglement swapping, and (c) quantum-gate teleportation (telegate). BSM: Bell-state measurement. CM: controlled operation and projective measurement.

qubits [42], optical modes [43], atomic ensembles [44], nuclear magnetic resonance [45], trapped atoms [46,47], and solid-state systems [48]. Over the last years, the focus has moved to teleporting more complex states – larger number of degrees of freedoms or higher dimension qubits [49,50] – and to real-world applications in quantum communications and computation [38,51,52].

In the teledata protocol, Alice and Bob share an entangled Bell state as that given by Eq. (1) [42], see Figs. 2(a) and 3(a) in physical and circuit representations, respectively. A third party provides Alice with a qubit C to be teleported to Bob. Importantly, the quantum state of qubit C –represented by ρ– is unknown to both Alice and Bob unlike in remote state preparation [54]. Alice then performs a Bell-state measurement (BSM), which randomly projects with equal probability her qubits A and C into one of the four Bell states $|\Phi^{\pm}\rangle$ or $|\Psi^{\pm}\rangle$. As a result, Bob's qubit B is simultaneously projected onto the state $T^{\dagger}\rho T$, where $T \in \{I, X, Z, ZX\}$ is an elementary or a combination of Pauli operators. As the last step, Alice informs Bob of the BSM outcome through the classical channel using two classical bits – feed-forward – and Bob applies the suitable gate T to his qubit to recover the unknown state ρ of qubit C at his location.

Regarding the figures of merit of quantum teleportation, there are mainly two:

1. The *BSM efficiency* or Alice's success probability for distinguishing a complete basis of entangled states – like the four Bell states. This varies for different information encodings: for instance, for a simple realization of Bell-state measurement using DV photonic qubits, the Bell efficiency is 50% at maximum [55].
2. The *teleportation fidelity* $F \in [0, 1]$ between the input state ρ and Bob's output state averaged over all Alice's measurement results and input states. The benchmark for the teleportation fidelity is surpassing the fidelity for state transfer without quantum resources, using, for instance, just classical correlations, i.e., $F > F_{\text{class}}$, where $F_{\text{class}} = 2/3$ for DV [56].

Table 1
Some milestones in quantum teleportation in terms of Bell efficiency, fidelity, distance of teleportation, and quantum memory. QED: quantum electrodynamics.

Quantum technol.	Bell eff.	Fidel.	Max. dist.	Memory
Polarization [38]	25%	0.80	1400 km	NA
Integrated opt. [51]	25%	0.894	10 m	NA
Superconduct. [41]	100%	0.79	chip	1 ms
Cavity QED [64,65]	100%	0.833	60 m	–
Ion Trap [66]	100%	0.845	chip	–
Rare-earth [67]	50%	0.86	1 km	17.5 μs

Table 1 shows examples of recent milestones in quantum teleportation in different technologies. More details on the state of the art can be found in [57,58].

Quantum teleportation has seamlessly made the leap from laboratory conditions to real-world implementation in urban environments, showcasing its adaptability and robust functionality. Teleportation networks allow for the reliable transfer of quantum information between a number of distant nodes, even in the presence of non-ideal features such as noise and loss. Recent advances include demonstrations of two-node teleportation over a metropolitan network [59,60], links between nanophotonic memories and ion traps in an urban network [61,62], and multinode entanglement over a metropolitan network with a cloud of Rubidium atoms in a ring cavity acting as a quantum memory [63]. More on quantum networks will be delved in Section 3.

2.3. Variants of quantum teleportation

Quantum teleportation is a primitive of quantum information science and has a number of variants essential for DQC. In the following, we review the most important three: entanglement swapping, quantum gate teleportation – telegate – and multipartite teleportation.

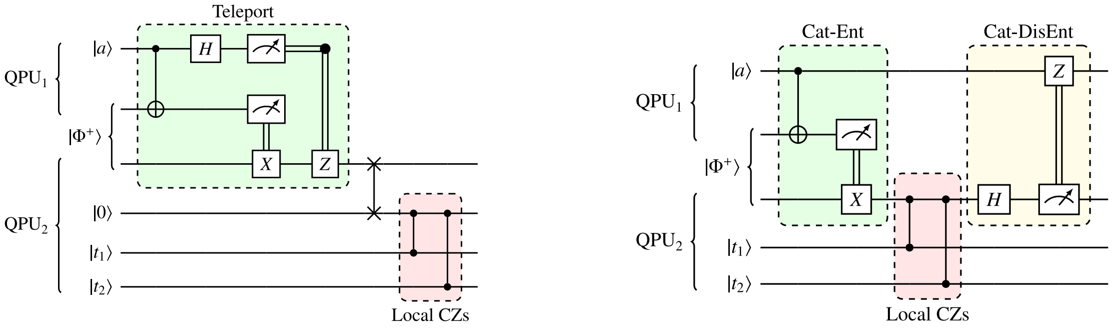
2.3.1. Entanglement swapping

Entanglement swapping is a variant of quantum teleportation that enables remote correlations by the transfer of quantum entanglement between distant end-users that do not directly share a quantum resource. In this case, Bob shares two entangled states, one with Alice and the other with Charlie, as shown in Fig. 2(b). Bob acts as a relay between them, performing Bell measurements and broadcasting the outcomes by a classical channel to them, who apply the suitable gates to their qubits. As a result, Alice and Charlie now share an entangled state conditioned on the result of Bob's measurement [36]. This protocol, together with entanglement distillation[2] [68], enables the distribution of entanglement over large distances, being the basis of quantum repeaters [37]. Related to entanglement swapping are fusion gates [69, 70], where projective measurements probabilistically *fuse* small entangled states in order to produce large entangled states – cluster states – useful for measurement-based quantum computing [40].

The first demonstration of entanglement swapping was carried out by Pan et al. using polarization-entangled photons [71]. Swapping

---

[2] Entanglement distillation, aka entanglement purification, involves converting N copies of any entangled state ρ into a certain quantity of nearly pure Bell pairs, solely through local operations and classical communication.



(a) Teledata.

(b) Telegate.

Fig. 3. Examples of teledata and telegate circuits for the application of CZs gates over $|t_1\rangle$ and $|t_2\rangle$ with the remote state $|a\rangle$ as control. (a) The state $|a\rangle$ in QPU$_1$ is teleported to the first qubit of QPU$_2$ (b) Cat-entangler and cat-disentangler primitives [53] are used to implement the remote control.

has been recently applied to connect two spatially-separated solid-state quantum memories by telecom links [67], and to entangle non-neighboring Nitrogen Vacancy (NV) qubits in a multinode teleportation network [72].

2.3.2. Quantum gate teleportation or telegate

In gate-based quantum computing, a sequence of unitary operations (usually single- and two-qubit) are applied on a set of qubits. However, sometimes there is no direct interaction between qubits on which we want to apply a two-qubit gate [20]. Quantum gate teleportation, also known as telegate, reduces the topological requirements by substituting two-qubit gates with other cost-effective resources: auxiliary entangled states, local measurements, and single-qubit operations [39]. Typically, Alice and Bob want to perform a non-local operation on unknown control and target states using a shared Bell state as a quantum channel. To this end, both perform locally controlled operations and projective measurements (CM) on their half Bell state and control/target states. After this step, partial quantum information is transferred between the two parties conditioned to the measurement outcomes. Cross communication of the results through a classical channel enables Alice and Bob to perform suitable corrections to the control and target states. This procedure results in a controlled gate operation on two non-interacting input states – see Figs. 2(c) and 3(b) for physical and circuit representations, respectively. The first experimental demonstration of quantum gate teleportation was a remote CNOT operation carried out through photon entanglement and linear optical manipulations [73]. Recent advances in remote operations comprise superconducting qubits, trapped ions, and quantum electrodynamics cavity nodes [41,64,66].

When applied to multipartite entangled states with a given topology, suitable measurement on a given network node teleport unitary-transformed-state to other nodes. This is the basis of measurement-based quantum computing [40].

2.3.3. Multipartite teleportation

Multipartite entangled states as the Greenberger-Horne-Zeilinger (GHZ) state enable a natural extension of quantum teleportation to more than two parties [74]. These N-party protocols for multipartite teleportation enable two variants: assisted and unassisted teleportation – commonly referred to as quantum telecloning. In the first case, *assisted teleportation*, Alice *helps* the communication between Bob and Charlie by performing a tailored measurement and broadcasting the result to them, thus improving the entanglement between them [35]. In the second case, *quantum telecloning*, Charlie teleports to Alice and Bob simultaneously, hence with a teleportation fidelity, limited by the no-cloning theorem, given by $F = (MN + M + N)/(MN + 2M)$, for N senders and M receivers of qubits [75].

Examples of assisted teleportation are open-destination teleportation [76] and, more recently, shared-quantum-secret teleportation [77]. Quantum telecloning was, in turn, demonstrated in DV by means of partial teleportation [78]. Cloning of entanglement [79] and copy distribution [80] are recent examples of this variant of teleportation.

2.4. Quantum devices for entanglement distribution

In the search for maximum performance and demonstrating quantum advantage for distributed, scalable quantum computing systems, modular architectures featuring specialized, single-purpose hardware are currently under development [81]. The quantum devices that are part of these architectures, apart from QPUs, can be categorized in one of the following categories: *quantum transducers*, *quantum memories*, *quantum repeaters*, and *entanglement routers and switches*. This section will describe the aforementioned devices in detail and discuss the current research advances in each technology.

This section will detail the aforementioned devices in detail and discuss the current research advances in each technology.

2.4.1. Quantum transducers

The communication between *local* qubits of systems where the quantum operations take place (e.g., QPUs, memories or repeaters) requires the conversion, or *transduction*, of their states to a different system used for delivery of quantum states in the form of *flying* qubits, which have the requirements of being highly mobile and well coupled to the specific local platform. Multiple flying qubit systems have been proposed, such as short-distance electronic states in semiconductor devices [82], direct delivery of nuclei with long-lived nuclear-spin qubit encoding [83] and, more commonly, single photons, given their naturally mobile nature and their low coupling with the environment.

In classical communications, the high-rate transfer of current technologies is only possible due to the high bandwidth, and low attenuation and latency provided by light in fiber optics, enabling the underwater connection of continents at tens of thousands of kilometers [84]. The current state-of-the-art telecommunication systems also implement multiplexing, i.e., encoding information at multiple wavelengths through the same fiber [85]. As DQC requires deterministically distributing entanglement, the requirements for flying qubits are, primarily, good coupling to the particular local quantum system, either by direct emission or interaction, and that the fidelity of the resulting entangled states is maximal. Quantized states of light are also the most natural information carrier choice for the distribution of quantum states at a distance, and extensive research has focused on the accurate manipulation of photonic states using linear and nonlinear optical devices [86,87].

The most widely studied way of generating entanglement between remote systems is via entanglement swapping between two independently entangled flying qubit-matter qubit systems, i.e., generating entanglement between photons and local qubits (trapped ions, neutral atoms, or NV centers), then performing BSM on the photons of each pair. Hence, their joint wavefunction collapses in the same non-separable state, and the matter systems become entangled. To this purpose, heralded entanglement of photons emitted after de-excitation from prepared excited states has been shown in trapped-ion qubits [88–

90], neutral atoms [91] and diamond NV-center qubits [92–96]. After the subsequent BSM, fidelities to Bell states of up to 88% at 230 m have been demonstrated in trapped-ions [97], and even above 60% between NV-centers separated by a 25 km metropolitan fiber link [98]. Deterministic qubit state transfer between different NV-center nodes has also been achieved [72]. One promising proposal is the coupling of ion- or Rydberg atom-chains in optical cavities [99], which has been shown capable of providing any-to-any entanglement for large systems with over 500 qubits in trapped ions by using two atomic species, one of which acting as a communication qubit and another as memory qubit. Quantum dots are also promising due to their tunable emission wavelengths in the infrared range, yet some challenges remain such as extending the qubit lifetime [100,101]. Rare-earth doped crystals, commonly Eu^{3+}- or Pr^{3+}-doped Y_2O_3 crystals also have emission near the 1550 nm band with sharp linewidths and long coherence times [102]. Spin-photon coupling in the microwave range has also been demonstrated in Si double quantum dot spin qubits by coupling the charge dipole of a trapped electron to the electric field component of a cavity photon stored in a superconducting resonator [103]. The promising advantages of these links is that multiple quantum dot qubits can use the same resonator, as their coupling can be switched on and off in nanoseconds, and the resonators' dimensions are large compared to the double quantum dots.

Finding mechanisms to link superconducting chips together to overcome their scaling needs is a current technological challenge. One of the most interesting experiences has been the deterministic transmission of excitations between superconducting QPUs using cryogenic microwave waveguides [104,105]. Applying modulated microwave pulses, an effective coupling between energy levels in the transmon qubits and their respective transfer resonators can be achieved, transferring an excitation from node A to its transfer resonator, which then emits a single microwave photon towards node B where it is absorbed, exciting its qubit. Cryogenic, lattice-based quantum networks have been proposed based on this method of connecting superconducting chips [106], demonstrating high fidelity, in excess of 85% when correcting for readout errors. However, some serious drawbacks remain, such as the high cost of cryogenic equipment, complexity and low modularity of these systems.

Correlated photon sources such as spontaneous parametric down-conversion (SPDC) or quantum dots can also be utilized to achieve the initial photon–matter qubit entanglement. SPDC sources consist of a non-linear crystal pumped by a strong laser beam generating pairs of maximally entangled photons with some probability, which can then be frequency-filtered and used to interact with the physical qubits. Using this technique, distant solid-state quantum memories have been entangled at distances of 1 km and fidelities of up to 86% using two photons at 606 nm and 1550 nm [107]; the former is stored in the collective excitation of Pr^{3+} in a doped crystal using the Atomic Frequency Comb (AFC) protocol, while the latter is sent to a BSM analyzer and corrected, resulting in entanglement. Hyperentanglement, where more than one degree of freedom can simultaneously be maximally entangled (e.g., polarization and direction of two photons) has also been demonstrated using this type of sources [87,108]. Alternative quantum dot-based sources have very attractive properties for this purpose, such as being triggered on-demand and energy-tunable [109–111], and reaching fidelities over 90% [112,113].

A less widely studied possibility is making single photons interact *in-flight* with two separated quantum systems. Using ancillary doubly reflected single photons followed by a measurement of the photon and a conditional rotation of the target qubit, heralded deterministic teledata [65] and telegate entanglement of remote qubits [64] have been demonstrated. Achieving fidelities up to 90% and 85% respectively at 60 m, this technique could enable deterministic, short-distance, low-latency DQC. Its advantages are two-fold: teleportation is performed without the need for preformed Bell pairs, so it can be done just-in-time; and losing the photon (i.e., not being able to measure it) does not lead to a mixed state in the matter qubits, so it can be tried again. While it is a new and promising avenue for the distribution of quantum states, further research is required to bring its quantum efficiency and fidelity closer to unity.

Moreover, interconnecting quantum systems may require coupling platforms that operate at different photon frequencies. For this purpose, techniques are being developed to implement frequency conversion of single photons on demand, maintaining certain properties (such as polarization) intact, which would enable the transcoding of qubits between platforms. One such technique is heralded up-conversion from infrared to visible light, which has been achieved through sum frequency generation in nonlinear crystals [114,115]. More recently, Murakami et al. [116] have demonstrated frequency conversion from visible to infrared using pairs of non-degenerate photons generated by SPDC, and Weaver et al. [117] have shown frequency bidirectional transduction from microwave to infrared light using transduction assisted by a resonant mechanical mode. However, the quantum efficiency of these techniques is currently low and significant efforts are underway to push it towards unity. In addition to the aforementioned frequency conversion techniques, recent work by Sahu et al. [118] has demonstrated deterministic entanglement between the quadratures of propagating microwave and optical photons in cryogenic waveguides, a first step towards interconnecting superconducting qubits with long-range communication systems and memories.

In summary, there are multiple competing techniques which allow distant QPU to generate entanglement in virtually all matter qubit technologies. However, much research is still required to push both fidelity and efficiency towards unity. Frequency conversion is a promising technique which may allow future interfacing of different systems, enabling heterogeneous DQC.

2.4.2. Quantum memories

To fully take advantage of the entanglement distribution and distillation protocols for both short and long distance quantum communication, it is paramount that the coherence time of the communication qubits is longer than the protocol itself, surviving multiple rounds of qubit exchange and entanglement purification. These long-lived qubits, organized as large registries, are known as quantum memories or quantum Random Access Memorys (qRAMs).

The simplest quantum memories are photonic memories, in which photons are stored and then retrieved after a given time. Multiple approaches exist, such as using free space optical loops triggered by heralding [119] or fiber delay lines [120] and cavities with tunable Q-factor [121,122]. Stimulated photon-echo is a more advanced technique based on the absorption and delayed reemission of single photons with the same quantum state after an ensemble of atoms is rephased [123–125], which has been demonstrated e.g., using slow light by electromagnetically-induced transparency (EIT) [126], controlled reversible inhomogeneous broadening (CRIB) [127] and atomic frequency combs (AFC) in rare-earth doped crystals [102,128,129]. All-photonic systems (i.e., photonic quantum computing) can already take advantage of photonic memories, as they do not require transduction [130,131].

However, both the difficulty of retrieving single photons with high fidelity as well as the low scalability of photonic-based memories have pushed forward extensive research on multiple alternative quantum memory technologies, demonstrating high-fidelity single-qubit gates in excess of the threshold needed for quantum error correction [132,133]. Notable examples are trapped-ion and -neutral atom qubits, which use the hyperfine structure of atomic ensembles of ions [134], or neutral alkali or alkaline earth single atoms in optical tweezers [135–137] to encode the quantum states, which can be individually addressed by microwave pulses [138]. Quantum memories based on diamond NV-centers have also been demonstrated (see [139] and references therein). Some of these technologies have demonstrated long coherence times, of up to 10 min in single trapped-ion qubits [140] and up to six hours in cryogenically cooled Eu^{3+}-doped yttrium orthosilicate nuclear spin qubits [83]. More recently, Barnes et al. [137] have demonstrated an individually addressable 21-qubit register of highly coherent and independent qubits with coherence times of about 40 s using nuclear spin qubits in optical tweezers, opening the gate to intermediate-scale quantum memories.

2.4.3. Quantum repeaters

As we previously discussed, light is the most natural long-distance carrier of quantum states. However, the absorption of light imposes intrinsic physical limits on the distance at which single photons can travel. In long-distance fiber communications, absorption is mainly produced by the fiber, with an attenuation coefficient in the range ∼0.14–0.4 dB/km in low loss telecom fibers [141,142]. Furthermore, even in the short-distance communication range of a datacenter, the rate at which photons are lost is nontrivial: the typical loss per SC connector is ∼0.25 dB [143], so the shortest possible connection between two nodes accounts for ∼0.5 dB of attenuation, i.e., ∼11% of the photons are lost. Hence, if frequent quantum communication is required for a distributed algorithm, the error probability quickly increases as $e = 1-10^{n\cdot \mathrm{dB}/10}$ after n exchanges, limiting the scalability and reliability of the calculation.

It is important to understand that any improvements in the connector losses and fiber attenuation cannot and will not solve the problem of exponential decay with n. Given that standard telecommunications erbium-doped fiber amplifiers (EDFA) cannot be used to amplify arbitrary quantum states due to the no-cloning theorem, *quantum repeaters* are essential to the implementation of entanglement distribution and teleportation which enable deterministic transmission of quantum states and remote quantum operations between nodes [144,145]. An early solution to the problem of implementing a quantum repeater was proposed by Briegel et al. [37], which consisted of first entangling noisy and imperfect qubits and then creating a high-fidelity entangled pair through entanglement distillation. Recent proposals have extended the idea of entanglement distillation to qudits (i.e., d-state systems) [146], multiple simultaneously entangled degrees of freedom (hyperentanglement) [147,148], and logical qubits [114,149]. Van Leent et al. [150] have demonstrated single-atom entanglement over a 33 km telecom fiber using quantum repeaters, proving that long distance entanglement is already a technical possibility. Recent work has also shown that Er^{3+} inclusions in calcium tungstate greatly diminish optical spectral diffusion [151], a requirement to generate indistinguishable single photons needed for optical repeaters, as this ion is well coupled by its telecom band optical transition.

2.4.4. Entanglement routers and switches

As previously explained, the execution of general quantum algorithms in multiple qubit-limited QPUs requires entanglement to be generated on demand between pairs of arbitrary qubits [152]. For this reason, recent research has focused on implementing teleportation protocols between non-neighboring nodes. The simplest way to obtain arbitrary entanglement with interconnected QPUs is pre-establishing shared entanglement, as discussed in Section 2.4.1, in a *one-to-one* fashion between specific communication qubits in different nodes. In these *one-to-one* schemes, not every pair of QPUs ought to be physically connected, reducing the complexity of implementation for small integrated systems.

However, this apparent simplicity faces a major scalability challenge, resulting in substantial qubit swap and distillation overhead in complex, strongly entangled algorithms [14]. While compilation optimizations can reduce swap operations, more general and modular quantum networks will need *entanglement routers* and *switches* to distribute entanglement between arbitrary qubits, akin to classical counterparts [153–155].

For quick reference, classical routers are capable of finding optimal routes in a complex network and understand the Internet Protocol (IP), while switches only recognize which physical addresses are routed through their connections to redirect traffic. The current absence of a quantum IP standard makes the distinction of the quantum counterparts difficult, so authors have been using these terms interchangeably. Moreover, the quantum hardware required is essentially the same and any differences would arise from the higher-level classical network management. Following this description, any two QPUs in the network can be connected through either one or multiple switches and/or routers in a Quantum Local Area Network (QLAN), or through an efficient routing path that connects multiple routers (which may require repeaters to maintain entanglement) and lead to a Quantum Wide Area Network (QWAN) [106,156]. The interconnection of quantum networks could eventually lead to a worldwide Quantum Internet [157,158]. However, this escapes the scope of this review [156,159,160].

Entanglement switches and routers can then be thought of as single-purpose QPUs: their sole objective is establishing entanglement among compute nodes through entanglement swapping, for which implement all the quantum technology required, such as quantum registries, entanglement sources and means to perform BSM, as well as all the hardware required for networking logic and classical communications [160]. Moreover, these devices may also be built on different quantum platforms than the proper QPUs, e.g., not requiring the implementation of a complete set of quantum gates but only those required for the swapping protocol and instead requiring registries of qubits with very high fidelity and coherence times longer than the entanglement distillation protocol, or access to quantum memories that fulfill these two requirements. Some proposals suggest networks based on single atoms trapped and coupled to optical resonators as memory qubits, which have long coherence times and good photon coupling (see [161] and references therein).

3. Networks for distributed quantum computing

The scientific literature on quantum networks is really extensive and a dedicated review would be needed to properly address all aspects regarding architecture, entanglement creation and distribution, network orchestration, network software stack and protocols. Therefore, our intention in this section is to provide sufficient information to give the reader an outlook of some relevant progress on this subject. Although considerable amounts of research has been oriented towards communication systems for the Quantum Internet, much of it can be applied to DQC. However, in DQC, the focus should be on short/datacenter distance limits.

Quantum networks (QNs) enable the execution of distributed operations among two or more qubits that may be very close to each other or separated by long distances. The mechanisms used for communication could be based on the transmission of the quantum states, or on the creation, distribution and consumption of entanglement. The entanglement resources provided by these QNs can be used both in DQC and in other applications of quantum technologies e.g., sensing or encryption. A comprehensive review on entanglement networks covering the fundamental mechanics, the enabling technologies, the network architecture and elements and research challenges associated can be found in [162].

Entanglement networks allow the execution of the previously discussed swap, teleport or telegate mechanisms. Besides these well known network mechanisms, Miguel-Ramiro et al. propose the inclusion of full quantum functionalities that increase the parallelism of operations using superposition of tasks with quantum control [163]. Examples of this are the execution of superposed tasks such as superposed measurement/non-measurement, superposed paths, superposed teleporting/non-teleporting or superposed merging/non-merging of graph states.

Classical network architectures and protocols cannot be directly extrapolated to quantum networks for entanglement distribution due to their particularities compared to the transmission of classical bits, such as:

- The duration of entanglement and the lifetime of the qubits due to decoherence.
- The probabilistic nature of some quantum mechanisms.
- The need for mechanisms to improve fidelity, such as distillation.
- The possibility of joining entanglement links not only through sequential operations but also through operations carried out in parallel on the various links.
- The different entangled resources – bipartite, multipartite by means of GHZ, W, cluster states, etc.
- The need for both quantum and classical channels to achieve the desired functionality.
- The possible use of quantum networks not only for the transmission of quantum information but also for the distribution of entanglement between distant points, which can be used as a resource by itself.
- The resource reservation strategy, if needed.

Software development requires the definition of a set of protocols that satisfies the different communication requirements from the most basic physical level communication to the application level communication, usually defined as a stack of network software layers. The survey [164] summarizes the main works on network protocol stacks, compared to the classical OSI or TCP/IP, and provides a comparison of the different stacks. Also noteworthy is the publication of the Internet Research Task Force (IRTF) Architectural Principles for a Quantum Internet [165], where the general guidelines for the design of quantum networks are presented. Regarding hardware architecture, much work has focused on tackling the delivery of entanglement in different technologies for the development of quantum repeaters. However, this is also to applicable to the development of quantum network devices for DQC: all optical and matter qubits architectures, based on discrete variable, based on continuous variable, based on bipartite entanglement or multipartite. In what follows, a list of relevant work related to DQC about quantum network hardware architecture and software stacks is presented:

- [166] presents a CV all-photonic switch for entanglement creation among end-nodes that uses Gottesman–Kitaev–Preskill qubit encoding and Steane codes error correction.
- [167] proposes an architecture for a CV continuous variable quantum switch where end nodes share entanglement links error corrected by means of Noiseless Linear Amplification (NLA) [168].
- Dür et al. [169] propose an architecture and network stack for quantum networks based on multipartite entanglement (GHZ graph states) allowing the generation of graph states of any type among clients.
- Van Meter et al. [170,171] propose a Quantum Recursive Network Architecture (QRNA) describing the layers of network communications that tackle entanglement distribution end to end. They introduce a recursive layer architecture in which swapping and purification functions are repeated to build *end-to-end* entanglement paths from a sequence of links, being entanglement performed at link level. Physical and link layer are in charge of entanglement establishment at link level (point-to-point), while *Remote State Composition* and *Error Management* layers are recursive and are continuously repeated performing swapping and purification from entangled links until the system is able to build an end-to-end entangled path.
- Li et al. [172] and Dahlberg et al. [173] propose protocol architectures for quantum networks based on bipartite entanglement where the mission of physical and link layers is the establishment of reliable entanglement, the network layer's goal is the establishment of long distance entanglement, and the transport layer copes with the qubits reliable/deterministic qubits transmission.
- Pirker and Dür [169] propose an architecture and network stack for quantum networks based on multipartite entanglement (GHZ graph states) allowing the generation of graph states of any type among clients. This architecture is composed of four layers: physical, connectivity, link, and network. The main difference to the traditional OSI layer architecture relies on the introduction of the connectivity layer, which is responsible for allowing *point-to-point* or *point-to-multipoint* connectivity, as well as error correction and establishment of long-distance links. The link layer allows the creation of graph states in the network that clients will subsequently use for the creation of end-to-end graph states.

In relation to the protocols necessary for the QNs they are classified in layers of the Open Systems Interconnection (OSI) model, in an analogue way to the classical counterparts. Several implementations of some of the functionalities of each communication layer have been proposed:

- Layer 1 (Physical): A. Dahlberg et al. define a protocol for physical entanglement generation based initially on NV center platforms [173]. A. S. Cacciapuoti et al. compare the addressing needs in quantum networks and classical networks (for instance, entanglement to a destination might require to perform entanglement to an intermediate node instead of the destination), and also the implications in the superposition of paths for both the addressing and routing design [174], while J. Miguel-Ramiro et al. assign each network device an identification register and an activation register which depend on the target node [163]. When a Toffoli operation on both registers is successful the node takes an active role in the operations.
- Layer 2 (Link): J. Illiano et al. propose an entanglement access control mechanism (analogue to the ethernet MAC mechanism) to grant access to a subset of the nodes to the entangled state (contention resource) [175]. The mechanism is based on quantum communications using multipartite entanglement Dicke states and preserves the anonymity of the granted nodes. R. Hanson and S. Wehner define a link layer protocol for robust entanglement generation sensitive to specific application needs (create and keep or create and measure, number of entangled links, atomicity of the links creation, fidelity, and other relevant parameters for the link creation) [173].
- Layer 3 (Routing): M. Caleffi designs a routing protocol and metric for quantum networks considering the key parameters for entanglement generation and the needed optimization to determine the optimum path between two points in the network [176].
- Layer 4 (Transport): Yu et al. propose a protocol for the reliable transmission of quantum information [177]. The protocol is based on the three way handshake of the classical counterpart TCP [178] and on a recursive quantum secret sharing method ((2,3) threshold scheme [179]) to achieve the transmission of the quantum data reliably, where the message to be transmitted is encoded in segments, being able to recover the message when only 2 out of 3 segments are available. If one of the segments is lost, one of the remaining segments will be reencoded and the method repeated.
- Layer 5 (Application)[3]: T. Satoh, R. van Meter et al. include the design of quantum sockets in analogy to the classical communications sockets, allowing the applications to access the services and having similar functions (creation and destruction of the socket, connection to the socket; reading from the socket, writing to the socket and configuring the socket) [171,181].

Regarding the control plane, [182] propose a control architecture for entanglement generation in quantum networks that moderates the requests for entanglement resources with the goal of fair distribution of the service to the network end nodes. [183] proposes a protocol to manage entanglement requests and resources (memories) management.

---

[3] A quantum operating system for quantum network applications has been implemented and experimentally tested running a client–server application on NV nodes in [180].

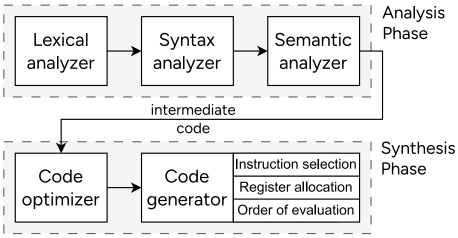


Fig. 4. Sequential phases of classic compiler process: analysis and synthesis stages.

4. Development layer

In the realm of classical computing, compilation serves two primary purposes: translating complex programming constructs into machine-specific executable instructions and optimizing machine resources to produce efficient code. Typically, this process follows a common scheme, as illustrated in Fig. 4, which consists of two main phases: analysis and synthesis. The analysis phase is responsible for conducting the code's lexical, syntactic, and semantic analysis to ensure correctness. Once validated, the code is translated into an Intermediate Representation (IR), which simplifies the implementation of optimizations in the synthesis phase.

Regarding quantum compilation, the scheme followed is usually the same as in the classical world. This is mostly because quantum compilation turns out to be a fully classical task, leaving the quantum workload just for the execution part. This leads to the situation where many quantum development software tools are actually built on top of classical languages, allowing the analysis phase to be integrated into an existing implementation.

Adding distribution to this task does not alter the compilation scheme; it remains largely the same with some additional steps and restrictions. To fully picture the differences and intricacies of compiling a distributed program, this section will be divided into two parts: Section 4.1 will elucidate the various methods by which a quantum process – usually referred to as a quantum circuit – can be distributed, while Section 4.2 will delve into how the compilation process is executed considering the distributed nature of the task.

4.1. Types of distribution

Distributed computing makes it possible to organize the computation of a problem in different Processing Units (PUs), which are connected through an interconnection network. The advantages of this model are evident: reducing the execution time by leveraging multiple PUs computing in parallel or, for large problems that do not fit within a single node, partitioning them to enable their solution. The time reduction comes with its own set of disadvantages, notably the increased difficulty in adapting algorithms and codes to a distributed approach. This is due to the significant overhead caused by communications and synchronizations, which must be carefully considered and managed [184].

Therefore, the complexity of developing a code increases when it is distributed. This complexity especially impacts the compiler design. In the analysis phase, new communication directives need to be developed, while in the synthesis phase, various network architectures must be considered to optimize data transmission and reception [185].

Certainly, the network's communication mechanisms and the resources the quantum task requires dictate the applicable distribution model, as depicted in Fig. 5. Three distinct categories of quantum distribution emerge: *circuit distribution*, *circuit cutting*, and *embarrassingly parallel*. It is clear, looking at Fig. 5, that all categories converge in compiling, executing, measuring, and post-processing information. Each of these distribution methods involves specific strategies for partitioning and managing the quantum circuit across different QPUs. For instance, Fig. 6 illustrates how Bell pairs can be created in the three main distribution categories. The following sections will dissect each of these categories to fully understand how the quantum distribution works in each case.

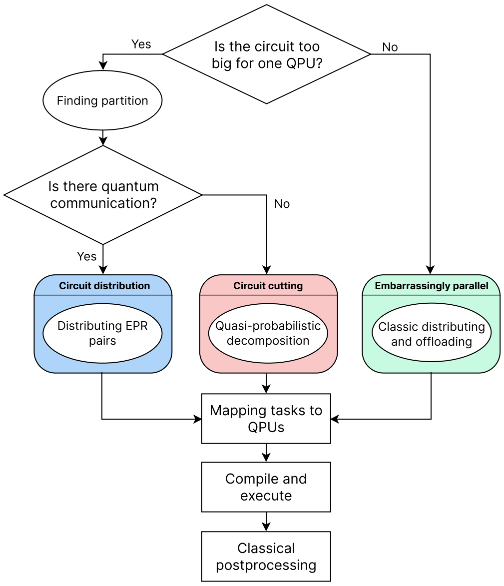


Fig. 5. Types of quantum distribution and their stages simplified.

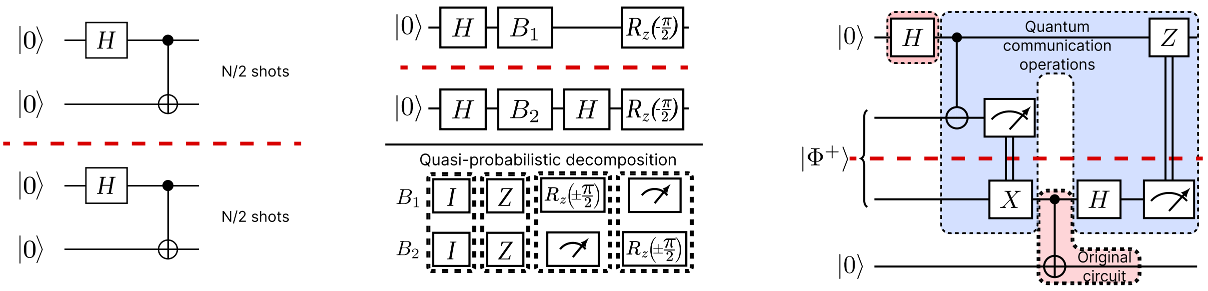
4.1.1. Circuit distribution

Circuit distribution, as shown in Fig. 5, involves three main phases: first, finding an optimal or near-optimal partition; second, distributing the partition among the available QPUs, and third, mapping this partition to each QPU. However, partitioning the circuit presents the most significant challenge and will be the primary focus of our efforts in this section. The other aspects are common to all the distribution types and will be further explained in the compilation Section 4.2.

First, for partitioning, the quantum circuit is mapped onto a graph that shows interconnections between elements. Thus, quantum circuit partitioning turns into a graph partitioning problem: given an undirected graph $G = (V, E)$ with a vertex set V and an edge set E, the aim is to partition V into two or more subsets regarding a cost function, like the number of edge cuts generated by the partition.

Graphs assume that the interaction between vertices is by pairs. However, even the most trivial phenomenon implies more than two vertices interacting concurrently. It is necessary to broaden the graph concept to gather these multilateral connections. The so-called *hypergraphs* [186] generalize the graphs to more complex situations. In short, while a graph can establish connections by pairs, a hypergraph is an object that connects more than two vertices or pins through elements called hyperedges or nets, as shown in Fig. 7. Thus, a hypergraph $H = (V, E)$ is an ensemble of pins V and nets E among those pins, and a net $e \in E$ is a subset of more than two pins.

Hence, hypergraph partitioning generalizes graph partitioning. More precisely, a k-way hypergraph partitioning groups the pins of a hypergraph into k blocks minimizing an objective function so that few nets connect pins from different blocks. The exchangeable objective functions are the cut-net and the connectivity metrics. The cut-net metric generates independent blocks of vertex sets by minimizing the nets belonging to several blocks, whereas the connectivity metric weights each net e with a factor $\lambda_e - 1$ to diminish the λ_e blocks connected by a net. The cut-net objective function sums over the nets among blocks and the connectivity metric over the λ_e blocks connected



(a) Embarrassingly parallel (b) Circuit cutting (c) Circuit distribution

Fig. 6. Example of the creation of a Bell pair for each type of distribution studied. In all types a partition of the tasks or the circuit (dotted line) is specified.

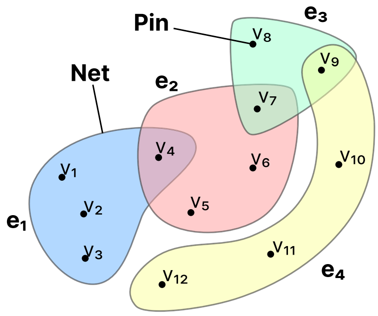


Fig. 7. Example of a hypergraph with twelve pins v_i and four nets e_j. Net e_1 has a size of four as it ensembles four pins, and pin v_4 has a degree of 2 as it belongs to two nets.

by a net. Nevertheless, both are analogue to the edge-cut problem in graph partitioning.

Underneath the goal of minimizing the cut-net and connectivity metrics lies an important consideration: while a valid partition may suffice for DQC, it may not necessarily be an optimal partition. For instance, in the circuits responsible for teledata and telegate operations – as illustrated in Fig. 3 –, these operations add up to four layers of depth to the circuit to enable operations among qubits in different QPUs. Consequently, this introduces latency to the quantum circuit, especially considering the additional synchronization required for intermediate measurements contained in both protocols between both QPUs. This latency represents a significant bottleneck in circuit distribution. Therefore, all circuit partitioning methods aim to minimize the utilization of teledata or telegate protocols. This aspect will be crucial in the circuit distribution techniques discussed in this section and beyond, summarized in Fig. 8.

Zomorodi et al. [187] introduced a first approach aiming to reduce communication between partitions, considering only two QPUs. They use the Kernighan-Lin (KL) [188] algorithm, a heuristic algorithm for graph partitioning, to divide the graph vertices into two subsets to reduce the edges across the subsets to minimize communication between the two partitions. After that, they apply a custom algorithm that aims to reduce the number of teleportations applied.

The work of Martínez and Heunen [189] was one of the most significant contributions in the field, serving as a foundational reference in many of the articles discussed here. Their method involves two key phases: a pre-processing phase, which groups equivalent gates, and a second phase, where hypergraph partitioning is performed using Karlsruhe Hypergraph Partitioning (KaHyPar) [190,191], a multilevel hypergraph partitioning framework that enhances cut-net and connectivity metrics. They evaluated their algorithm using five quantum algorithms known for their quantum speedup, such as Quantum Fourier Transform (QFT). A criticism of this work is that it did not consider optimizations such as moving gates back and forth to bring them closer together nor explore the entire search space of different partitioning options for executing global gates, which limits their ability to produce optimal solutions.

These limitations were pointed out in the work of Houshmand et al. [192], who improved on the work of Zomorodi et al. by exchanging the algorithm responsible for reducing teleportations – which had exponential cost – for a genetic one, which allowed them to significantly reduce the execution time. This work, like the Zomorodi et al. one, only considers a two QPU scheme, reason why Daei et al. [193] enhanced it by effectively mapping a quantum circuit into an appropriate number of distributed components. Moreover, Nikahd et al. [194] also go beyond, categorizing the binary gates into distinct "levels", followed by determining the optimal partitioning of qubits for each level through the solution of an integer linear program.

The work by Martínez and Heunen [189], on the other hand, was extended with an entanglement-efficient protocol [195] derived from [15] and with, among other things, a hypergraph approach to arbitrary network topologies [196]. In the first case, authors pack multiple non-local controlled unitary gates locally with one maximally entangled pair through a distributing and embedding pipeline. In the second, the authors also search for efficient entanglement within the network by reusing already available connections. In fact, this work led to many different articles employing hypergraph partitioning with KaHyPar.

Following the KaHyPar line, Sundaram et al. contribution concerns communication timing, non-local operations availability (teledata and/or telegate), and partitioning. First, engaging KaHyPar, Sundaram et al. [197] present a two-step heuristic for the distribution of quantum circuits: dividing the given qubits among the computers in the network with KaHyPar and scheduling communication operations, called migrations – equivalent to cat-entanglement operations [53]. They present a polynomial-time solution for the second step in a specific setting and a $\mathbb{O}(\log n)$-approximate solution in the general setting. Second, they amplify the available remote protocol for communications between QPUs, [198]: while Daei et al. [193] use teledata and, on the contrary, Martinez and Heunen [189] and Sundaram et al. [197] use telegate, there is upgrading in Sundaram et al. [198] applying both. For the telegate protocol, they consider a method similar to the initial two-step heuristic work, [197]. However, to partition the given qubits among QPUs, they use a Tabu-search-based heuristic regarding the heterogeneity of the network and storage limits. For the general DQC problem, they employ two heuristics: *Sequence*, a greedy approach, and *Split*, similar to the previous one, but with an iterative approach. Both contemplate the telegate solution as a subroutine. Lastly, Sundaram et al. take a step further in a recent work [199] by designing two different protocols to reduce the number of teleportations needed to perform the distributed task. The first method, termed *Local-Best*, tries to minimize the teleportation of qubits by selecting them only when necessary, with the choice of teleportation influenced by gates in the near future. The algorithm consists of two steps:

1. Find an initial assignment of qubits to computers to minimize the number of resulting non-local binary gates.

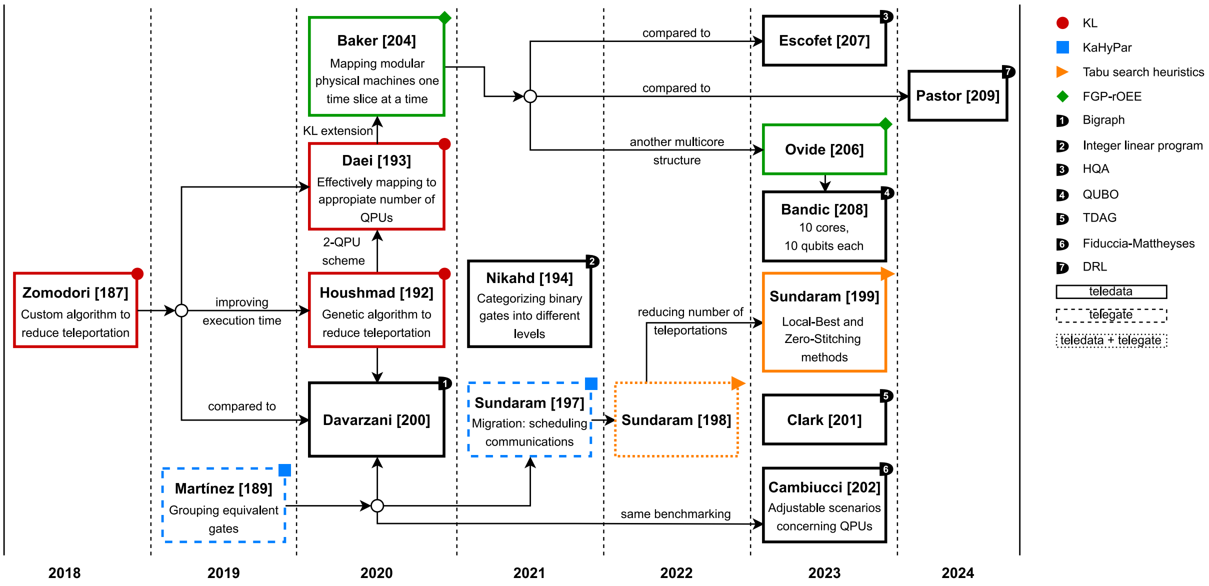


Fig. 8. Scheme of contributions to quantum circuit partitioning.

2. For each non-local binary gate G, select the teleportations to execute G locally based on the “near future” in order to minimize the total number of teleportations.

The second method to shorten the number of teleportations, *Zero-Stitching*, comprises also two steps:

1. Identify “zero-cost” subcircuits: These are contiguous subcircuits that can be executed without any teleportations.
2. Divide the given circuit into zero-cost subcircuits and “stitching” them together using teleportations.

There are also approaches employing bipartite graphs instead of hypergraphs. Davarzani et al. [200] introduce an algorithm for distributing quantum circuits to optimize the number of teleportations between qubits that consists of two steps: first, the quantum circuit is converted to a bipartite graph (bigraph), and, second, the bigraph is partitioned into K parts employing a dynamic programming approach. Finally, they compare their results with the ones yielded by works previously analyzed [187,189,192] and they claimed that the experiments gave better or equal results for benchmark circuits.

In another approach, proposed by Clark et al. [201], a different model than hypergraph is employed. They introduce the Tree-based Directed Acyclic Graph (TDAG) partitioning for quantum circuits, a novel method that views circuits as a series of binary trees and selects the tree containing the most gates for partitioning.

Besides minimizing the communication between partitions, Cambiucci et al. suggest [202] adjustable scenarios to the capabilities and constraints of the processing units involved in the distribution are considered. In this work, instead of the KL from the original hypergraphic approach, authors implement a variation of the Fiduccia–Mattheyses algorithm [203], which is a faster approximation algorithm for min-cut partitioning with a computational time that grows linearly with the network size. They use the same circuits as [189] for benchmarking.

A field-changing approach was the work developed by Baker et al. [204]. While still based on graph partitioning, this method seeks to avoid reaching a single static assignment for an entire circuit by employing near-optimal graph partitioning techniques. It leverages the inherent clustering of the DQC paradigm and the statically-known control flow of quantum programs to develop tractable partitioning heuristics. These heuristics map quantum circuits to modular physical machines one time slice at a time. Specifically, optimized mappings are created for each time slice, considering the cost to move data from the previous time slice and utilizing a tunable lookahead scheme to reduce the cost of moving to future time slices. To achieve this, a customized version of the Overall Extreme Exchange (OEE) algorithm [205] – considered a natural extension of the KL algorithm – referred to as relaxed-OEE (rOEE), is employed. Because the primary approach to map the circuit to the hardware is Fine Grained Partitioning (FGP), this method is usually referred to as FGP-rOEE. This method was further analyzed by Ovide et al. examining it under another multi-core architecture but maintaining the all-to-all qubit and cores connectivity [206]. Moreover, a Hungarian Qubit Assignment (HQA) method for partitioning is developed by Escofet et al. which also describes the assignment of qubits to cores between timeslices, and it is compared to the FGP-rOEE method [207].

A recent approach that has elevated the work of Baker et al. is the technique presented by Bandic et al. which relays on Quadratic Unconstrained Binary Optimization (QUBO) to partition the circuit at each time slice [208]. Their method’s primary strengths are rooted in the formulation of the QUBO itself. This structure enables the decoupling of the problem definition from the solver as well as surpassing the limitations of look-ahead approaches utilized in the Baker et al. solution. It is worth noting that, in this approach, two different multi-core architecture layouts composed of 10 cores with a capacity of 10 qubits each were tested, in contrast with the non-realistic all-to-all connectivity assumed by the previous approaches.

Last but not least, one of the most novel algorithms is a circuit partitioning method that employs Deep Reinforcement Learning (DRL) [209]. Once again, the FGP-rOEE is employed as a baseline to compare the results and as an inspiration due to its time-sliced graph partitioning. This work has considered three approaches: Proximal Policy Optimization (PPO), Soft Mask, and Hard Mask. The first one, the PPO, is a widely used algorithm within the DRL scheme, while the remaining two, Soft and Hard Mask, are a variant of the former PPO algorithm that introduces a masking mechanism. The Soft Mask approach adds a simple mask, which disables useless operations – such as swapping identical qubits, swapping two qubits situated on the same machine, or advancing to the subsequent time slice without establishing a valid assignment for the current one – whereas Hard Mask implements a *direct-swap* heuristic in top of the Soft Mask which solely evaluates the relocation of misplaced qubits to the respective core they need to interact with.

Now that we have explored the state-of-the-art in the circuit partitioning problem, we can understand why it poses such a significant challenge. Finding the optimal partition directly impacts performance and is a critical aspect in the later stages of compilation, where the boundaries between software and hardware become narrow. Specifically, this problem is closely related to the qubit mapping and circuit optimization stages of the distributed quantum compiler, which will be defined and explained in Section 4.2.3 as part of the compilers's synthesis phase.

4.1.2. Circuit cutting

As detailed in Section 3, on the road to fully functional DQC, one needs quantum communication in the form of a quantum network between the devices. In the absence of such of networks, there are several alternative techniques to simulate, or at least approximate, this entanglement using a classical network. In this context, circuit cutting has been suggested as a solution to partitioning a wide circuit requiring many qubits into smaller, non-entangled subcircuits. These subcircuits can then be executed (emulated) sequentially on a limited-qubit (memory) device or in parallel across multiple devices. There are several different strategies for circuit cutting, such as gate-cutting and wire-cutting (shown in Fig. 9), which will produce different subcircuits. The output of the original circuit is recovered using a combination of the results of the subcircuits, with some cost in accuracy that is overcome by increasing the number of circuit executions. This extra cost is often called sampling overhead, and it is known to grow exponentially with the number of cuts.

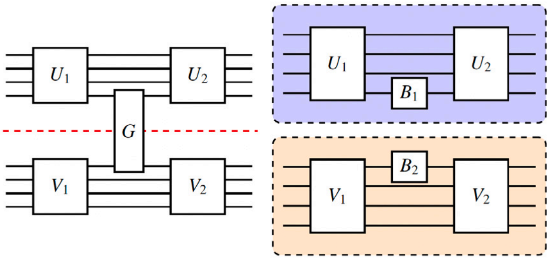
Quasi-probabilistic decomposition of quantum channels. Most circuit-cutting algorithms rely on the quasi-probabilistic simulation (QPS) of a quantum circuit, which uses the quasi-probabilistic decomposition (QPD) of the *quantum channel* of the circuit. A quantum channel $\mathcal{E}$, or *quantum operation*, is a trace-preserving, completely positive linear map between density operators. Quantum channels are typically represented through the operator-sum representation, also known as Kraus decomposition. In this representation, a channel $\mathcal{E}$ acts on a state described by a density matrix ρ as a sum of k terms $\mathcal{E}(\rho) = \sum_{j=1}^{k} E_j \rho E_j^{\dagger}$, where E_i are (Kraus) operators on the Hilbert space of ρ.

This representation is not unique, i.e., one has the freedom to choose the operators E_i of the representation and still get the same channel $\mathcal{E}$. In particular, one can choose the operators to be quantum gates that are *local* in separate sets of qubits. Consider the n-qubit bipartite system $\rho = \rho^{(1)} \otimes \rho^{(2)}$ with Hilbert space $\mathcal{H} = \mathcal{H}^{(1)} \otimes \mathcal{H}^{(2)}$, where $\mathcal{H}^{(1)}$ and $\mathcal{H}^{(2)}$ are the space of the two unconnected sets of qubits $\rho^{(1)}$ and $\rho^{(2)}$. Now consider a quantum circuit C consisting of products of arbitrary quantum gates, some of them multi-qubit gates acting on both $\mathcal{H}^{(1)}$ and $\mathcal{H}^{(2)}$ simultaneously. Our hardware may not be able to execute those non-local gates, but one can always find a decomposition such that

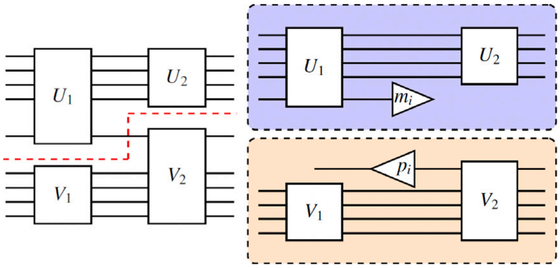
$$\begin{aligned}
\mathcal{E}(\rho) &= \sum_{i}^{m} q_i \left(V_i^{(1)} \otimes V_i^{(2)} \right) \left(\rho^{(1)} \otimes \rho^{(2)} \right) \left(V_i^{(1)\dagger} \otimes V_i^{(2)\dagger} \right) \\
&= \sum_{i}^{m} q_i \left(V_i^{(1)} \rho^{(1)} V_i^{(1)\dagger} \right) \otimes \left(V_i^{(2)} \rho^{(2)} V_i^{(2)\dagger} \right) \\
&= \sum_{i}^{m} q_i \, \mathcal{E}_i^{(1)} \left(\rho^{(1)} \right) \otimes \mathcal{E}_i^{(2)} \left(\rho^{(2)} \right) \ ,
\end{aligned}$$

with coefficients $q_i \in \mathbb{R}$ with $\sum_{i=1}^{m} q_i = 1$, and $V_i^{(1)}$ and $V_i^{(2)}$ are operations acting locally in $\mathcal{H}^{(1)}$ and $\mathcal{H}^{(2)}$ respectively, that our hardware can physically execute. The choice of q_i and the set of $V_i^{(1)}$ and $V_i^{(2)}$ is not unique, and it is known as a QPD of the quantum channel [210].

The q_i can be either positive or negative, which is why they are called quasi-probabilities. The larger the number of negative coefficients in the decomposition, the larger the 1-norm $\kappa = \sum_{i=0}^{m} |q_i|$ of the QPD becomes. Crucially, this κ quantity is related to the cost of executing the circuit C that has non-local gates, using only local operations [211,212]. Negative probabilities in the simulation of quantum circuits were already known to be related to the "quantumness" of quantum circuit. Thus they could be used as a resource to classically simulate quantum processes by separating the "hard" and the "easy" parts of the circuit [213,214], and also for performing error mitigation through a quasi-probabilistic decomposition of an ideal circuit from noisy ones [215,216].



(a) Gate cutting (spatial cut).



(b) Wire cutting (temporal cut).

Fig. 9. Two schemes for cutting a quantum circuit: gate-cutting (or spatial cut) as shown by Mitarai and Fujii [223] and wire-cutting (or temporal cut) by Peng et al. [224]. Both can be shown to be equivalent to simulating teleportation [225], wire-cutting being analogous to teledata and gate-cutting to telegate.

In practice, to calculate the expected value of an observable, we sample the outcome of the circuit measured in the appropriate basis for some number of shots N_s. We want N_s to be large enough so as to have some desired degree of accuracy ϵ. When using QPS to simulate circuits, the variance of the result increases with κ^2, and we have to compensate for increasing N_s proportionally. This effect is known as sampling overhead. This overhead is multiplicative, increasing exponentially with the number of cut gates N_c. Given a large enough number of shots, the outcome of the original circuit is recovered with arbitrary precision. However, noise sources will still introduce a bias in the computation independent of the QPS, as noise is a separate quantum channel evolving the state ρ. Quasi-probabilistic methods can also aid in error mitigation, which as mentioned above has some practical overlap with circuit cutting. Furthermore, there is experimental evidence that QPS can reduce the effect of noise sources by employing smaller circuits [217,218]. Another issue appearing when sampling a QPS appears when reconstructing the evolved ρ from the partitions. Due to finite sampling error, finding a distribution with negative terms is possible. To solve this, post-processing can be used to find the "most likely" output state [219,220], although this is not necessary for calculating expected values of observables.

Finding an efficient QPD of a general circuit C, i.e., a QPD with a small κ, is difficult. If the circuit is known to produce a state with a particular bi-partite structure, one can turn to similar techniques to execute the parts locally, such as Entanglement Forging [221,222]. However, the main direction that has been followed in the literature for circuit cutting was to perform only the QPD of specific regions of the circuit with sparse correlations, targeting non-local gates or wires.

Circuit cutting techniques: gate-cutting and wire-cutting. One preliminary work, which was later labeled as circuit cutting (and in particular, wire-cutting), was the *cluster simulation scheme* by Peng et al. [224], which decomposes the corresponding tensor network of a given quantum circuit into smaller clusters. Inter-cluster communication is then simulated classically. The authors apply these techniques for Hamiltonian simulation using the Variational Quantum Eigensolver (VQE) [226], and suggest using this hybrid variational ansatz for future modular architectures. Later, Mitarai and Fujii [223] introduce the idea of *virtual two-qubit gates*, where the action of the virtual gate is substituted with local operations. This way they only apply QPS for the non-local gates we want to get rid of. Given that most QPUs can only execute single- and two-qubit gates, it is more convenient to find an efficient QPD of the particular two-qubit gate and simulate them with local single-qubit gates. The total overhead of the QPS then scales as $\mathbb{O}(\kappa^{2N_c})$ with N_c being the number of virtual gates. Mitarai and Fujii also provide an efficient QPD for a two-qubit gate with $\kappa = 3$ at most, from which most common two-qubit gates such as $CNOT$, CZ, $RZZ(\theta)$, etc., can be derived. Fig. 9 compares the two methods, which can also be used simultaneously in the same circuit.

The main drawback of circuit cutting is the exponential overhead. This overhead has been proven to be strictly exponential [227], so it cannot be reduced to a polynomial increase and it proves a big challenge for scaling to large problems. Still, minimizing κ is an active research topic. In [225,228], the minimal sampling overhead to cut wires and two-qubit gates is derived analytically. Brenner et al. [225] show that cutting an identity gate that transported the state of the qubit before and after the cut (a wire cut) is equivalent to a teleportation protocol. As shown in Section 2.2, one needs a prepared Bell state and two bits of classical communication to teleport one qubit of data. Gate-cutting of a Bell pair between two qubits ($\kappa = 3$) is already more efficient than cutting a wire ($\kappa = 4$), although it requires ancilla qubits.

Piveteau et al. [228] suggests that this overhead can be reduced when jointly cutting multiple gates or wires, using classical communication between partitions. This is because the joint QPD of a larger unitary of N_c Bell states (also called *Bell State factory*) required for teleporting N_c gates has a lower overhead than individually cutting N_c Bell states. This overhead now scales better $\kappa = (2^{N_c+1} - 1)$, albeit using local operations and classical communication (LOCC) and ancilla qubits as requirements (one per partition and cut). While Piveteau et al. did not give the explicit QPD of this *Bell State factory*, it was later provided in [229] for $N_c = 2$ and $N_c = 3$.

Lowe et al. [230] reduced the ancilla qubit requirements for large-scale Quantum Approximate Optimization Algorithm (QAOA) simulations by combining wire-cutting and random measurement bases (inspired by classical shadow tomography [231]) and subsequent studies improved bounds for multiple wire-cuts with LOCC [232,233]. For gate-cutting [234] improved on Piveteau et al. result, finding an optimal decomposition of an arbitrary two-qubit rotation gate and reducing the ancilla requirements for cutting multiple parallel gates. Soon after, [235] achieved a similar result for clustered Hamiltonian simulation, and [236] did the same for general two-qubit unitaries.

Reducing sampling overhead can also be achieved by cutting larger unitaries. For instance, cutting a SWAP gate using QPS results in a lower overhead ($\kappa = 7$) than decomposing it into three CNOT gates and cutting each individually ($\kappa = 3^3$). This, of course, is the idea behind cutting the Bell State factory in [228] but can also be extended to higher dimension operators like Toffoli gates [236], multi-controlled CZ gates [237], and even the QFT [238]. Furthermore, in the case of Variational Quantum Algorithm (VQA) one can choose variational ansatzes designed with reduced entanglement between parts, so they are easier to partition. This can be in the form of clustered ansatzes for VQE [239,240], or more general ansatzes where the amount of entanglement is tuned so that the overhead of the QPD is always kept below a tolerance value [241].

Other strategies reduce the number of subcircuits in decompositions to lower sampling overhead. Note that, while related in their exponential scaling, the number of subcircuits in a QPD (its 0-norm) is not the same as the sampling overhead (its 1-norm). Reducing subcircuits can aid scheduling and post-processing without increasing κ. Nagai et al. realize this by introducing pre- or post-selection methods for quantum channels [242], while Chen et al. use approximate methods that directly neglect some of the elements [243,244].

Efforts to minimize quantum communication between machines focus on smart qubit assignment. A solution that minimizes the sampling overhead also minimizes the number of Bell pairs in a DQC protocol, and thus, the same compiling tools could be used for both techniques. Combining gate- and wire-cutting finds better partitions [245], which is crucial for DQC, not only for circuit cutting, as already detailed in Section 4.1.1. Some Software Development Kits (SDKs), such as Qiskit or Pennylane, incorporate these techniques in their compilation routines. Moreover, several tools such as CutQC [246], ScaleQC [247] or SuperSim [248] perform the whole circuit cutting pipeline, finding cuts, executing the subcircuit, and reconstructing the state. There is also, as we will delve in Section 4.2, a compiler named Qurzon [249] which performs all the aforementioned techniques – in fact, it uses CutQC in combination with other tools.

Herzog et al. [250] illustrated the practical application of these methods by cutting a QAOA ansatz for a combinatorial optimization problem. Their approach combined the strategies in [224,232] to reduce ancilla and classical communication requirements, while utilizing classical graph shrinking techniques to lower the overall overhead. However, the authors noted that the same classical techniques could potentially solve the problem faster through purely classical computation. Similarly, IBM's recent work [229] demonstrated the execution of a 142-qubit graph state across two 127-qubit QPUs. By implementing Piveteau et al.'s concept of a Bell State factory, they utilized real-time classical communication and parametric circuits to reduce compilation time, showcasing the practical application of circuit cutting in large-scale quantum computations.

Despite these advancements, circuit cutting occupies a challenging position in the quantum computing landscape. It is more suited for problems with sparse entanglement, which are often more easily tackled using classical methods. At the very least, circuit cutting can be useful for early DQC applications, serving as a transitional strategy until robust quantum communication networks are fully realized. This technique could thus provide a crucial stepping stone in the evolution of quantum computing infrastructure.

4.1.3. Embarrassingly parallel

The term *embarrassingly parallel* was coined within the HPC domain to describe applications that are inherently amenable to parallelization without significant effort. Notable examples include bag-of-tasks workloads – jobs devoid of dependencies that can be executed in any sequence – and parameter sweep applications, which involve numerous parallel executions with varying parameter configurations.

Similarly, in the context of quantum computing, the term *embarrassingly parallel* refers to the scenario where a problem can be divided into multiple smaller computations that can be executed independently without the need for direct communication among them. The simplest example of this in the quantum case is the *distribution of shots*, where a quantum algorithm or kernel needs to be executed multiple times without any structural changes – except for the modification needed to map the circuit to the different QPUs. Despite the quantum nature of the tasks involved, this method essentially involves classical parallelism.

A different approach comes from a distribution of the circuits needed to reconstruct the expectation value of a given observable or to support the optimization protocol. This allows several possibilities:

- *Distribution of terms in an observable*. The distribution of the expectation value terms $\langle O_i \rangle$ of a given observable $\langle O \rangle = \sum \langle O_i \rangle$ is a case of embarrassingly parallelization. An intuitive example is the VQE [226], where the function to minimize is the energy, i.e., the expectation value of a Hamiltonian $\langle H \rangle$. Depending on the specific problem, Hamiltonians can be commonly expressed using fermionic operators in second quantization formalism, as in the case

of many systems in condensed matter/chemistry, bosonic operators, or directly in Pauli operators, as in spin Hamiltonians that apply to different problems in physics, route optimization, protein folding [251], and scheduling, among others. In all cases, except the last one, the Hamiltonian has to be mapped to qubit instructions via some encoding techniques [252,253]. After that, it appears as a weighted sum of tensor products of Pauli operators, most commonly known as Pauli strings. Initially, each Pauli string can be individually sent to different QPUs. However, the scaling in the number of Pauli strings for complex problems makes this procedure inefficient. A common practice is to form groups of Pauli strings that will share the same quantum circuit to construct their expectation value. These groups are made of commuting Pauli operators that are determined using some classical routine. The simplest strategy is *qubit-wise commutativity*, where each of the commuting groups built can be measured using a single quantum circuit without difficulties [252]. An alternative is *general commutativity*, which is more efficient in reducing the number of commuting groups but entails the non-trivial task of finding the appropriate unitaries for the joint measurement of the groups [252,254].
- *Gradient and Hessian's distribution*. Just like the preparation of a parameterized trial wave function $|\psi(\theta)\rangle$ to our problem, first and second partial derivatives of the state $|\psi(\theta)\rangle$ can be analyzed with a quantum computer [255–257]. In many cases, the quantum circuits that arise from the partial derivatives can be expressed as a linear combination of circuits that use the same structure of the original circuit to prepare $|\psi(\theta)\rangle$, with a shift in their parameters, which is known as parameter shift rule [258].
- *Distribution in a gradient-free optimization*. That is a particular case of distribution that sources from the usage of gradient-free optimizers such as evolutionary ones. These optimizers overcome the need to compute gradients at the cost of using several individuals/particles that interact in a certain way to modify their parameters or generate other candidates. That is the case, for example, of Differential Evolution and the Particle Swarm Optimization algorithms [251,259,260]. Each individual is a different set of parameters that can be executed in parallel using the same quantum circuit structure. One of the possible benefits of the previously mentioned optimizers is that they can mitigate problems in the optimization landscape [259,261]. However, this would come at the cost of increasing drastically the number of circuit executions.
- *Distribution of data*. As in the case of classical Machine Learning, another possibility is to distribute the data or the model during the training. For example, [262] proposes a tool for distributing training of Quantum Machine Learning models that can also be used for VQEs. A federated approach has also been proposed [263].

There are some packages that permit the distribution of these kinds of jobs among several QPUs [262,264], based on a master-worker architecture. These packages must cope with additional issues not seen in classical Machine Learning distributed learning, such as the different architectures of the QPUs (different gate sets, different topology, or different timing for execution), the noise of each single QPU and the possible drift of these errors with the time, for counting some of the current challenges. Additionally, these techniques can also be used when circuit cutting is applied.

Another paradigm that can be considered in this context is *multi-programming* of quantum computers. The segmentation of a QPU, better known as multi-programming in quantum computing, can maximize the hardware throughput – the number of used qubits divided by the total number of qubits – and reduce the runtime. The pioneering work for multi-programming by Das et al. [265] advocated for its use to enhance the utilization and throughput of Noisy Intermediate-Scale Quantum (NISQ) computers, wherein the qubits are employed to execute multiple workloads concurrently. Other works introduce enhancements like selecting the appropriate number of circuits to execute, qubit mapping, device benchmarking, crosstalk[4] characterization, or even vulnerability analysis [266–273].

Another paradigm that may be interesting to delve into is *quantum offloading*. As mentioned in the introduction, QPUs is intended to be seamlessly integrated into classical HPC infrastructures, working along other hardware accelerators. This way of distributing the workload allows concurrent computations of classical and quantum tasks, letting CPUs proceed with calculations while QPUs accelerate specific processes in which the so-called *quantum advantage* takes part.

A profound quantum offloading analysis diverges from this work's main scope, but some relevant works can be outlined. For instance, the eXtreme-scale Accelerator programming framework (XACC) is a system-level software infrastructure for quantum-classical computing that promotes a service-oriented architecture to expose interfaces for core quantum programming, compilation, and execution tasks [8]. Strongly related is QCOR, a language extension specification of C++ that enables single-source quantum-classical programming and that employs XACC as a base [9]. Another work leveraged the OpenMP API to target quantum devices, which provides an easy-to-use and efficient interface for HPC applications to utilize quantum computing resources [274]. Similar to this were the efforts made to add QPUs to the OpenCL ecosystem of execution [7]. Even the NVIDIA company has developed the CUDA Quantum Platform for hybrid quantum–classical computation [275], enabling the aforementioned integration and programming of QPUs along with other accelerators.

4.2. Compilation

After resolving the distribution challenge, it is essential to explore the compilation process thoroughly. We will adhere to a structure akin to the classical approach, which involves an analysis phase, an intermediate representation referred to as Quantum Intermediate Representation (QIR), and a synthesis phase. This framework will aid in comprehending the compilation process for DQC and underscore the disparities between classical and quantum computing in terms of compilation.

4.2.1. Analysis phase

The analysis phase in the distributed and monolithic quantum compilation is quite similar, with the additional challenge in the distributed case of limited literature and software development compared to the monolithic counterpart. In the monolithic scenario, the underutilization of standalone languages is not because they do not exist; rather, options like Scaffold [276], Q# [277], isQ [278], Q$|SI\rangle$ [279], among others, are available. However, they are less favored due to the need for users to understand and adapt to these languages. In contrast, libraries like Qiskit [280], Cirq [281] and Qulacs [282], built on well-known classical languages such as Python (Qiskit and Cirq) and C++ (Qulacs), are more widely adopted. This situation is even more pronounced in the distributed case because there is a shortage of standalone languages specifically designed for distributed purposes. Consequently, the previously mentioned quantum monolithic libraries are often repurposed to simulate the distributed structure.

This is the case for Quantum MPI (QMPI) [283], which represents an extension of the Message Passing Interface (MPI) protocol for distributed quantum systems. We refer to this as a formal approach due to the absence of a usable library that allows for actual or simulated DQC. However, a reference implementation for QMPI has recently been published [284], although none of the code is available for use, neither in open source nor as a binary, to the best of our knowledge.

The aim of QMPI is, obviously, to add quantum functionalities to an already widely used specification such as MPI. For this purpose, it

---

[4] Crosstalk is an unwanted coupling between qubits. It is one of the noise sources in NISQ devices and can condition the hardware throughput.

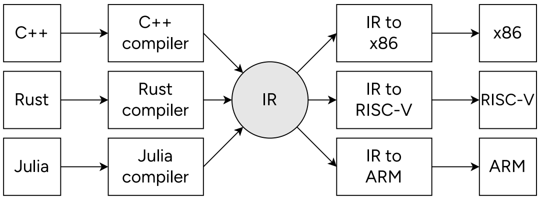


Fig. 10. The significance of intermediate representation in the compilation process - Facilitating decoupling between high-level and machine code.

defines two types of nodes: classical and quantum. The only difference between them is that classical nodes cannot be the target of quantum directives, whereas quantum nodes can manage both quantum and classical calls. The core of this difference lies in the inherent distinction between classical datatypes and quantum datatypes – bits and qubits – along with the inclusion of EPR pairs, a crucial element for the development of quantum communication protocols, as shown in Section 2. Other than that, although MPI is much more advanced than QMPI, as expected, the communication modes supported by the latter are the same: point-to-point communication and collective operations. Moreover, they define a simple performance model called SENDQ. It is worth mentioning that, contrary to almost all literature on DQC, they anticipate a relatively low logical clock speed for quantum computers due to the overhead introduced by the quantum error correction. Consequently, they do not expect classical communication to significantly affect performance, choosing to ignore classical communication in the SENDQ model. This approach contrasts significantly with all the circuit distribution methodologies discussed in Section 4.1.1, where the focus was primarily on minimizing the number of teledata and telegates, considered the main bottleneck of quantum distribution. The SENDQ model is closely associated with the NISQ era and may not be sustainable when transitioning to the fault-tolerant era.

Anyway, as it is explained in Wakizaka [285], there is a need to develop a proper quantum programming language that takes consideration of a distributed structure and extracts profit from that structure via advanced distributed computational techniques, just as it happens in classical computation.

4.2.2. Distributed quantum intermediate representation

The compilation process is complex; therefore, Intermediate Representations (IR) were introduced to establish a break in the compiler in order to obtain modularity and decoupling [286]. An IR allows to intermediate between the front-end and the back-end, improving the efficiency of compiler development and allowing abstract optimizations to the target machine. Fig. 10 shows the use of IRs as a break in the compilation process to facilitate compiler development so that programs are implemented for abstract machine code such as an IR.

An important feature of IRs is that they have to be able to represent the operations of different high-level languages to be implemented in different machine codes. Therefore, with the evolution of quantum computing, it is necessary to extend classical IRs (or create new ones) to include quantum instructions. This process has been evolving in recent years, where the number of quantum IRs has grown considerably [287–292].

For DQC, specialized IR are needed to allow the use of classical and quantum communication instructions between different PUs. This objective is what InQuIR [293], an IR specialized in DQC, aims to solve.

To exemplify the operation of this IR, we use the circuit shown in Fig. 3(b), which implements a CNOT remote gate between two separate nodes, but connected through a Bell pair $|\Phi^+\rangle$. Fig. 11(a) shows the OpenQASM code to implement this, which does not consider communication directives. The compilation of OpenQASM to InQuIR produces the code shown in Fig. 11(b) for node 0 and Fig. 11(c) for node 1. InQuIR automatically adds the necessary directives to do the remote operation using the telegate technique.

```
1  OPENQASM 2.0;
2  qreg q[2];
3  h q[0];
4  cx q[0],q[1];
```

(a) OpenQASM 2.0 code for the creation of an EPR pair.

```
1  0 {
2    world = open[0,1];
3    q0 = init();
4    _cq0 = genEnt[1](10);
5    CX q0 _cq0;
6    _m0 = measure _cq0;
7    free _cq0;
8    send[1](world, l1:_m0);
9    recv(world, l1_2:_m1);
10   Z[_m1] q0;
11 }
```

(b) InQuIR code for node 0 (qubit 0).

```
1  1 {
2    world = open[0,1];
3    q1 = init();
4    _cq1 = genEnt[0](10);
5    CX _cq1 q1;
6    H _cq1;
7    _m2 = measure _cq1;
8    free _cq1;
9    send[0](world, l1_2:_m2);
10   recv(world, l1:_m3);
11   X[_m3] q1;
12 }
```

(c) InQuIR code for node 1 (qubit 1).

Fig. 11. InQuIR representation of the creation of an EPR pair using remote gates.

The IR code extends the basic quantum operations to a distributed setting, where quantum communication and entanglement generation across different nodes (0 and 1) are involved. Lines 2 to 4 in both Figs. 11(b) and 11(c) correspond to the initialization of the communication channel between both nodes, the initialization of the local qubits, and the generation of the EPR pair, respectively. Lines 5–6 in 11(b) and 5–7 in 11(c) correspond to the gates and measurements. The measurement results are transferred between the two nodes by `send/recv` operations and used in the conditional gates.

4.2.3. Synthesis phase

In classical compilation, this corresponds to the lowest level of abstraction. At this stage, low-level, less human-readable languages—analogous to classical assembly languages—are utilized within the compilation chain. Although it is challenging to map each quantum compilation stage to distinct levels of abstraction, a parallel with classical assembly can be established through the use of Quantum Assembly Language (QASM). There are a lot of different versions, such as OpenQASM [294], cQASM [295], eQASM [296] and f-QASM [297]. But, to the best of our knowledge, only NetQASM [298] takes into account an underlying distributed structure.

In [298], Dahlberg et al. introduced an abstract model featuring a Quantum Network Processing Unit (QNPU) for end-nodes in a QN. NetQASM is proposed as an Instruction Set Architecture (ISA) designed to execute arbitrary programs on end nodes equipped with the QNPU. So, NetQASM can be seen as a low-level, assembly-like language tailored for the quantum segments of quantum network program code. It specifies the interaction between the QNPU and executes QN code, a functionality not available in other QASM languages. The language is designed to be extensible, with a core set of instructions for classical control and memory operations and a set of quantum-specific instructions grouped into "flavors". A "vanilla" flavor is introduced for universal, platform-independent quantum gates, enabling platform-independent quantum network program descriptions, with the possibility of developing platform-specific flavors for optimized quantum operations on specific hardware.

It is also worth mentioning the work of Ying and Feng [299]. They developed an algebraic language for formally specifying quantum circuits in DQC that aims to represent circuits conveniently and compactly, akin to how Boolean expressions are used for classical circuits.

Building on the classical analogy, this stage involves optimizing the code and adapting it to the target machine. The compiler performs operations such as register allocation, branch optimization, loop unrolling, and other well-known optimization techniques. Similarly, quantum compilation employs analogous optimization methods. However, unlike in classical compilation, these techniques are not always applied directly to QASM. Instead, they can be applied to the higher-level languages considered in this work. This distinction underscores the current lack of abstraction in quantum computing.

To maintain consistency with classical methodologies, the remainder of this section will elaborate on the three primary components of the synthesis phase: *optimization*, *qubit mapping*, and *verification*. First, *optimization* and *qubit mapping* will be discussed, as they are fundamental aspects of quantum compilation, particularly in the current NISQ era. Finally, the *verification* stage will be examined. Although *verification* differs in nature from the preceding two components, it serves as a crucial feature in quantum programming by providing an alternative to classical debugging techniques, ans so it will be explained at the end of the section as an important side aspect of the quantum compilation.

Optimization. The optimization phase in monolithic quantum computing encompasses a broad range of techniques aimed at minimizing various metrics, such as the number of 2-qubit gates, the circuit depth, etc. In DQC, we encounter similar optimization challenges as in the monolithic case, but with the added complexity of distributing or cutting the circuits. On the contrary, if the distribution technique performed is embarrassingly parallel, the optimization phase is, naturally, equivalent to the monolithic one, excepting the case of multi-programming where optimizations are subtle and tend to be related with crosstalk and fidelity [267,273].

Delving into circuit distribution, we have discussed in Section 4.1.1 the circuit distribution methods and efforts made to partition the circuit optimally before performing local mapping. In essence, optimization in this case mirrors that of the monolithic case but with the additional consideration of the partitioning problem, which is intricately linked to qubit mapping. Indeed, the close relationship between qubit mapping and circuit optimization is not surprising, even in the monolithic case. It is logical because an efficient mapping of qubits directly impacts circuit performance, much like how effective register management optimizes classical computing tasks. However, although we are only adding one more constraint with the circuit distribution, it is of vital importance since the teleport and telegate costs are significantly higher than those of local 2-qubit gates. As previously discussed in Section 4.1.1, this serves as justification for why circuit partitioning methods consistently aim to minimize the utilization of these remote protocols. Qiu and Chen [300] realized an interesting analysis of this topic, where the quantum cost figure of merit is employed. The quantum cost of a circuit is calculated by summing the cost of each gate present in the circuit. Any gate can be broken down into several basic gates, each with a unit cost, irrespective of their internal complexity. Using this definition of cost, they showed the expensiveness of quantum teleportation and dense coding. However, circling back to the main topic, while we have extensively covered and will further discuss partitioning in the qubit mapping section, we have deliberately chosen not to get deeply into the intricate domain of monolithic quantum optimizations, as it exceeds the scope of this work.

Regarding circuit cutting, optimizations aim at reducing the sampling overhead, or the number of subcircuits. Although both quantities are related in that both increase exponentially with the number of shots, in general, they do not need to scale the same way. The most important of the two is the sampling overhead. Still, a reduction of the number of subcircuits (without an increase in the sampling overhead) can also help in the scheduling and post-processing part of the computation. Some works reduce the sampling overhead by including LOCC, either when jointly cutting several gates [301], or in smart prepare-and-measure protocols in wire-cutting [230,232,233]. Other works attempt to cut larger unitaries [237] or constrain the overhead using parameterized gates [241]. Regarding the number of subcircuits, they can be reduced using pre- or post-selection methods [242], and some of them can be neglected in approximated methods without incurring in large errors [243,244].

Qubit mapping. When it comes to classic computing, register allocation is about finding the best way to use the limited number of registers available to store variables [302]. In the field of quantum computing, qubit mapping can be compared to register allocation in classical computing. This process involves finding an optimal mapping of logical qubits to physical qubits in a quantum device, taking into account the device's connectivity and other constraints. It is important to note the growth in complexity of this process as it moves from classical to quantum compilation. In the realm of quantum compilation, it is not only the use of the qubit's value that must be evaluated – meaning if it is thought to be a communication qubit or a computing qubit. Other factors, such as the error associated with the specific qubit and its interconnection with the remaining qubits, assume significance in the decision-making process. Qubit mapping is an NP-hard problem [303]. Therefore, exact algorithms are only computable for a reduced number of qubits, making it necessary to use techniques that are able to obtain an optimal solution even if it is not the best one. Additionally, the quantum mapping process can be separated into three processes:

- *Gate decomposition:* Refers to the stage in which gates composing the circuit are transformed into a series of native gates implementable in the actual quantum processor. This is one of the aforementioned device's constraints that have been taken into account.
- *Quantum allocation:* Refers to the process of physically assigning specific logical qubits in a quantum processor. For a correct qubit allocation, in most cases, it is necessary to add additional SWAP gates to move the qubit information [304].
- *Quantum routing:* Refers to the task of finding efficient paths for communication between qubits in a quantum processor. This is important when mapping gates of two logic qubits that are not interconnected to maximize efficiency [305,306]. For a thorough analysis of the qubit routing problem, one can check the review on the subject by Barnes [307].

Regarding DQC, it is essential to distinguish between distribution methods that require partitioning and those that do not. In the former case, where partitioning is necessary, the qubit mapping problem aligns with the classical problem. Still, it includes the additional challenge of optimizing circuit partitioning to minimize communication, as detailed in Section 4.1.1, where we already mentioned how linked those methods are with this stage of compilation.

Nevertheless, a few works that have not been mentioned in that section are of interest. The first one is the work of Mao et al. [308], who named the problem as qubit allocation problem for distributed quantum computing (QA-DQC), proved the NP-hardness of it and proposed two algorithms to deal with it: a heuristic local search algorithm and a multistage hybrid simulated annealing (MHSA) algorithm. In the latter, they combine the local search algorithm and a simulated annealing meta-heuristic algorithm, along with extensive simulations to evaluate it. The second work was also carried out by Mao et al. [309] that proposed a probability-aware qubit-to-processor mapping model, incorporating communication overhead between processor pairs determined through probabilistic analyses based on link entanglement generation rates. Additionally, they introduced a multi-flow routing protocol to enhance overall entanglement rates. Subsequently, they employed a multistage hybrid simulated annealing algorithm, which is reminiscent of the previous one, to minimize total communication overhead. As we have already mentioned, extensive simulations are conducted to demonstrate the effectiveness of these solutions across various system settings. The third work of interest in this line was the one developed by Nakai [310], which deeply developed the qubit allocation problem for DQC along with a formal definition of the problem as an optimization problem similar to how we have defined the partitioning one. Finally, the last work was developed by Chen et al. [311], where they focused on the step following the circuit partitioning, i.e., the qubit routing stage. Specifically, they focused on investigating the influence of the quantum state transmission direction during the execution of global gates on the number of transmissions and subsequent routing. It utilizes a heuristic algorithm, called Genetic Algorithm for Global Gate Direction Optimization (GAGDO), to ascertain the optimal transmission direction for all global gates in the circuit, with the goal of minimizing the overall cost of the executable circuit generated in the distributed architecture model.

Also, two works have been developed to characterize the inter-core qubit traffic in which some benchmarks arise in order to analyze mapping performance [312,313]. They employed the OpenQL compiler [314], which is not a distributed compiler *per se* but allows the embedding of a modified version of the Qmap mapper [315]. In particular, for this case, they extended it to the multi-core case employing the proposal by Baker et al. [204], i.e., the FGP-rOEE algorithm, already explained in Section 4.1.1.

In the cases of embarrassingly parallel distribution that do not require partitioning, the qubit mapping problem mirrors that of the monolithic case, with the added complexity of needing to perform mapping for each QPU. This complexity arises from the potential differences in architectures among the QPUs contained in the distributed scheme. There is just one case in the embarrassingly parallel scenario where qubit mapping differs from the monolithic case: the multi-programming scenario. This paradigm of quantum execution, which involves segmenting the QPU, imposes a series of constraints on the qubit mapping problem. One of the first approaches was the already mentioned work by Das et al. [265]. Three techniques were developed in this work:

1. Fair and Reliable Partitioning (FRP) algorithms, developed to partition qubit resources into multiple groups fairly while avoiding qubits or links with excessively high error rates.
2. Delayed Instruction Scheduling (DIS) policy, devised to mitigate interference from measurement operations of one program on the gate operations of co-running programs.
3. Adaptive Multi-Programming (AMP) design, proposed to monitor reliability impact at runtime and revert the system to isolated execution mode if the impact is high.

Different techniques were developed under the QuCloud framework by Liu and Dou [267]. In this work, they also developed three approaches. First, they utilized community detection techniques to partition physical qubits among concurrent quantum programs, mitigating resource waste. They even proposed a new technology based on these techniques called Community Detection Assistant Partitioning (CDAP). Second, they designed the X-SWAP scheme, which enables inter-program SWAPs and gives priority to SWAPs linked with critical gates to minimize SWAP overheads. Finally, they introduced a compilation task scheduler that prioritizes the compilation and execution of concurrent quantum programs based on estimated fidelity for optimal performance.

This was further extended in a subsequent work by the same authors under the QuCloud+ framework [273], in which they tried to take into consideration the crosstalk effect on real-world applications.

Verification. The verification of quantum programs is a significant side aspect of quantum compiling. Unlike in the classical world, where developers rely on debuggers to identify and fix errors, debugging quantum programs is inherently difficult due to the destructive nature of measurement. Once a quantum state is measured, it collapses irreversibly, making it impossible to observe the state at different time steps without altering it. Therefore, the verification of quantum programs becomes crucial for ensuring the correct functionality of a quantum circuit. It is essential to incorporate this verification step as a phase in the synthesis stage of compilation. This ensures that the circuit is checked immediately before execution and after optimizations have been applied to confirm that those optimizations have not altered the functionality of the quantum circuit. In the monolithic realm, several approaches have been made combining optimization and verification in what is usually referred to as *verified optimization* [291,316,317].

One way of verifying quantum programs is using quantum process algebras, which are derivations of the classical process algebras. Process algebras, also known as process calculi, are mathematically rigorous languages with well-defined semantics that allow the description and verification of properties of concurrent communicating systems, including, in this case, quantum systems.

There are some examples of these types of formal methods. For instance, Extended Quantum Process Algebra (eQPAlg) [318], which extends Quantum Process Algebra (QPAlg) [319]. More specifically, QPAlg provides a homogeneous style for formal descriptions of concurrent and distributed computations, encompassing both quantum and classical components. As authors claim, QPAlg introduces quantum variables, operations on these variables – unitary operators and measurement observables – as well as different forms of communication involving the quantum realm. The operational semantics ensure that these quantum objects, operations, and communications adhere to the postulates of quantum mechanics. Regarding eQPAlg, it extends the previous formal specification to accommodate the concept of formally specifying the quantum teleportation protocol, which has been shown in this work to be a key part of the quantum distribution model. The relationship between quantum process algebras and the algebraic language defined in the aforementioned work by Ying and Feng [299] can be compared to that between classical process algebras and Boolean algebra. In broad terms, quantum process algebras are well-suited for high-level formal specification of DQC, while the language Ying and Feng paper is mainly intended to describe low-level circuit implementation.

Regarding the verification of distributed quantum programs, the work of Feng et al. [320] introduced a distributed programming language designed for formalizing and verifying distributed quantum systems. They presented a Hoare-style logic[5] that is both sound and complete, aiding in the analysis and verification of quantum programs, including quantum teleportation and CNOT gates. Talking specifically about distributed quantum protocols, Wang's work [322] profoundly delves into the verification of several distributed quantum protocols, such as the BB84 protocol [323].

4.2.4. Available compilers

Not many full-stack tools or compilers are designed considering a distributed quantum scheme as a base. In fact, to the best of our knowledge, there is no compiler for DQC available for use, just conceptual designs and prototypes. These conceptual quantum compilers can be classified depending on which type of distribution they use from the ones described in Section 4.2, i.e., usual circuit distribution, circuit

---

[5] Hoare logic is indeed a formal system equipped with a set of logical rules used for rigorous reasoning about the correctness of computer programs [321].

Table 2
Summary of available compilers for DQC, including their authors, reference, descriptions, main focus, and categorization by distribution type.

Category	Tool/Compiler	Authors	Reference	Available	Main focus	Description
Circuit distribution	Distributed Quantum Compiler	Ferrari et al.	[324]	✓	Circuit depth minimization with specific partitioning strategies	Designed to minimize circuit depth using strategies based on data-qubit swapping and entanglement swapping. Compared against works like Martínez and Heunen.
Circuit distribution	Modular Quantum Compilation Framework for DQC	Ferrari et al.	[325]	✓	Comprehensive optimization considering network, hardware, and specific algorithms	A modular framework considering network and device constraints. Includes qubit assignment with METIS, EPR pair minimization algorithms, and optimized local routing.
Circuit distribution	Cuomo's compiler	Cuomo et al.	[326]	✗	Optimization of distributed architectures for dynamic networks	Models the compilation problem using Integer Linear Programming and time-expanded network representations. Optimized for dynamic network problems and quasi-parallelism.
Circuit cutting	Qurzon (with CutQC)	Chatterje et al.	[249]	✓	t$|ket\rangle$ for optimal qubit routing. Reconstructs original circuit results	Employs CutQC for cutting circuits into optimal subcircuits without quantum communications. Schedules execution using a greedy algorithm.
Embarrassingly parallel	palloq	Ohkura et al.	[327]	✓	High-fidelity layout synthesis for multi- programming scenarios	Manages multi-programming with layout synthesis based on noise adaptative layouts. Introduces a crosstalk detection protocol and integrates randomized benchmarking for multi-circuit allocation.
Combining techniques	Quantum Divide and Conquer Algorithm (QDCA)	Tomesh et al.	[328]	✗	Hybrid variational approaches combining cutting and distribution	Combines circuit cutting and distribution for hybrid variational applications. Uses graph partitioning techniques like METIS and KL for mapping large combinatorial optimization problems to distributed architectures.

cutting, and embarrassing parallelism. Table 2 provides a summary of the available compilers, detailing their authors, reference, descriptions, main focus, and categorization by distribution type, as discussed below.

Compilers for circuit distribution. Ferrari et al. [324] designed a distributed quantum compiler that focuses on the minimization of the depth of the circuit and, for this matter, two different techniques are tested: the *data-qubit-swapping-based* strategy and the *entanglement-swapping-based* strategy. They compared the performance of the partitioning – and, hence, of the distribution – of these two strategies with the already analyzed work by Martinez and Heunen [189]. Also, Ferrari et al. [325] designed a versatile modular quantum compilation framework for DQC, which considers both network and device constraints and characteristics. For qubit assignment, they employed METIS's multilevel k-way partitioning. Moreover, for gate scheduling, they implemented an algorithm to minimize the consumed EPR pairs and a local routing algorithm that scans the circuit and, for every gate that involves qubits not directly connected on their specific QPU, it computes the shortest sequence of necessary SWAP gates. The experimental evaluation of a quantum compiler based on this framework was demonstrated, using circuits of interest such as VQE, QFT, and graph state preparation, characterized by varying widths – ranging from 0 up to 600 qubits.

Cuomo et al. [326] modeled the compilation problem using an Integer Linear Programming formulation inspired by the extensive theory on dynamic network problems. They defined the problem as a generalization of the quickest multi-commodity flow, enabling optimization using techniques from the literature, such as a time-expanded representation of the distributed architecture. This approach, which also incorporates quasi-parallelism[6] allows for more efficient circuit operation and broader solution exploration. The work is modular, enabling adaptation to circuits with varying degrees of operation commutativity and leveraging existing network flow literature. The study aims to refine compiler efficiency and performance through an in-depth analysis of quantum circuits and focus on normal forms. Testing on square and hexagonal lattice topologies, showed that square lattices offer superior performance, attributed to their favorable edges-to-nodes ratio, indicating promising avenues for future quantum computing advancements.

Compilers for circuit cutting. As for now, the only quantum compiler considering the circuit-cutting strategy, as was explained in Section 4.1.2 is Qurzon [249]. For the first part of the compilation, an algorithm responsible for cutting the circuit into optimal parts is employed, called CutQC [246]. After the circuit is cut into several pieces, a scheduling algorithm is responsible for the execution of each of the pieces in the available quantum devices. This problem is nothing more than a classic problem of scheduling jobs, well known in the HPC environment. In this case, a greedy algorithm is employed, at least in the theoretical development of the compiler (since to obtain the results, they applied a so-called "naive" algorithm, which is not specified). For the optimal qubit routing, they reach out for the work of t|ket⟩ [329]. Then, a distributed parallel execution is performed over the whole group of subcircuits employing the different devices, and once the results are

---

[6] The authors define quasi-parallelism as a relaxed version of parallelism based on grouping logically sequenced gates within the same time step.

obtained, the CutQC work is again used to reconstruct the result of the original circuit using every result obtained in each subcircuit.

Compilers for embarrassing parallelism. Despite the absence of compilers specifically designed for embarrassingly parallel tasks in quantum computing, the inherent parallelizable nature of these tasks – primarily the distribution of shots across multiple QPUs – means that any quantum compiler or framework could be easily modified to support this mode of distribution. This adaptability is due to the fact that the distribution of computational tasks among different processors is a well-established practice in the field of HPC. Consequently, leveraging existing classical job distribution techniques allows for the straightforward parallel execution of quantum computations on multiple QPUs, highlighting a seamless integration of classical parallelism principles within quantum computing frameworks.

Nevertheless, an appreciation of the multi-programming case has to be made. Even though the already presented QuCloud and QuCloud+ [267,273] are considered mapping mechanisms, they possess a compilation task scheduler and could be naturally extended to be able to perform as compilers with a multi-programming approach. This is precisely the scope of *palloq*, presented by Ohkura et al. [327], which includes a layout synthesis for multiple quantum circuits and a job scheduler to manage efficient and high fidelity quantum multi-programming. This compiler takes multiple quantum circuits, written in OpenQASM, and the device's local gate error information as input. Their layout synthesis employs a heuristic based on noise-adaptive layout, where the device's calibration data is analyzed to search for improved allocation using a greedy approach. Additionally, they propose a software-based crosstalk detection protocol utilizing a novel combination of randomized benchmarking methods to characterize the hardware's suitability for multi-programming.

Compilers combining types of distributions. The work from Tomesh et al. [328] combines aspects of circuit distribution with the circuit-cutting technique [328]. This work introduced an algorithm called Quantum Divide and Conquer Algorithm (QDCA), a hybrid variational approach aimed at mapping large combinatorial optimization problems onto distributed quantum architectures. The QDCA specification contains several key elements: the partition of the input combinatorial optimization problem into multiple subproblems, the construction of the variational quantum circuit, and the execution of it on distributed quantum computers using quantum circuit cutting techniques. The partition of the input is where the classical techniques of graph partitioning employed for circuit distribution take place, in this case, KL and METIS. Even though it is not circuit distribution *per se*, it employs the graph partitioning techniques used in this kind of distribution to perform circuit cutting, which narrows the boundaries between these two approaches. This work presents quantum circuit cutting as a compilation tool within a hybrid, variational application. With this approach, they claimed to achieve approximate solutions to Maximum Independent Set (MIS) problems.[7]

5. Application layer

This section explores proposed quantum applications that leverage some of the methods previously outlined. Any quantum application executing at least one quantum circuit requiring multiple shots is inherently parallelizable, as the required shots can be distributed across available QPUs or the circuit can be partitioned using telegates or teledata. However, it is important to note that such parallelization does not necessarily guarantee enhanced performance; in fact, it could lead to a significant degradation.

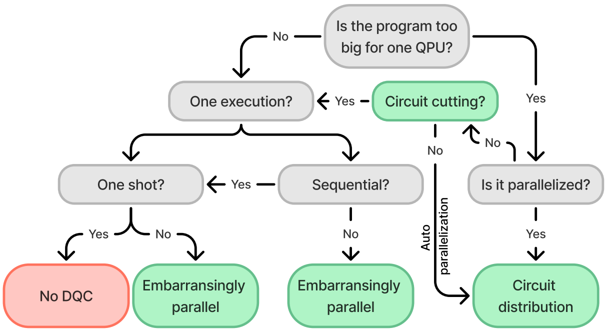


Fig. 12. Decision workflow for executing a quantum application in parallel.

Similar to classical HPC computing, the motivations for executing an application in parallel may include insufficient resources on a single QPU (e.g., a lack of qubits), stringent time constraints (where the time-to-solution fails to meet the requirements of the intended use case), distributed input data across various storage locations (it is easier to move the computation close to the data than the reverse), concerns over data security or confidentiality, among other considerations. The decision to parallelize can be made either by the user or delegated to an automatic scheduler.

For instance, focusing solely on spatial constraints (i.e., the number of qubits) and the required shots to achieve an acceptable result, Fig. 12 illustrates a decision workflow for selecting the appropriate execution method for an application. If the circuit demands more qubits than are available on a single QPU and the program is parallelized, only the required number of QPUs needs to be selected to execute it in parallel (e.g., circuit distribution, as outlined in Section 4.1). Conversely, if the application is not parallelized, a preliminary step could involve assessing the feasibility of circuit cutting — dividing the problem into multiple smaller circuits that can be executed independently without quantum communication, which implies effectively transforming the task into an embarrassingly parallel process. Should circuit cutting prove insufficiently efficient, an autoparallelization mechanism may reconfigure the circuit into a genuinely parallel program.

Conversely, if the basic circuit fits within the available QPU capacity, other options exist to use multiple QPUs in parallel to accelerate execution. For instance, if the problem involves running several circuits that can be executed independently, such as during the optimization of a variational quantum circuit, these instances can be distributed across the available QPUs. Additionally, the same circuit instance can be split among several QPUs, each handling only a fraction of the required shots. However, if these circuits are interdependent, where the execution of one depends on the results of another, only shot parallelization is feasible.

The complexity of the decision workflow of Fig. 12 increases if time constraints are included. In such cases, even if the program fits within a single QPU, parallelization may become necessary to meet the time requirements. However, selecting the number of QPUs must be approached cautiously, as parallel execution inherently introduces overhead that must be accounted for.

In the remainder of this section, we will present some selected examples of distributed quantum applications considering the division of Section 4.1 to show the possibilities of DQC. Specifically, we will discuss applications based on circuit distribution, those leveraging circuit cutting, and others that can be formulated as embarrassingly parallel. Other classifications of DQC applications also exist. For example, [330] recently analyzed the applications of distributed quantum computing, categorizing them into two main types: *resource DQC*, which addresses scenarios where a single device lacks sufficient resources, and *data DQC*, where data is distributed and QPUs can work collaboratively to get the result. Within this framework, the authors review various applications and discuss the challenges of implementing them on current

---

[7] The MIS problem is a classic NP-Complete combinatorial optimization challenge defined on a graph $G = (V, E)$. Its objective is to identify the largest feasible independent set within G, where an independent set, denoted as $S \subset V$, consists of nodes that are not adjacent to each other.

hardware. Readers are encouraged to consult this article for further examples of applications leveraging DQC.

5.1. Circuit-distribution based applications

As mentioned in the introduction of the paper, one of the first distributed algorithms was proposed by Grover [12]. In this work, he used a circuit distribution with quantum communications to estimate the mean of N numbers between -1 and 1 under ideal conditions. Later, Gupta et al. [22] presented a distributed version of the Grover search algorithm using quantum communications. Initially, the algorithm was shown using only two QPUs, where an additional qubit was needed in each QPU to handle the quantum communications using an EPR pair. The complexity analysis showed that the classical Grover requirements for operations are maintained in this distributed version since the increase in the number of operations due to the distribution scales with the number of qubits as in the original algorithm, but the number of classical communications per iteration is not increased. The paper did not show if the algorithm can scale to more than two QPUs.

One of the key quantum algorithms that present an exponential scaling is the Shor algorithm. The main drawback of this algorithm is the high number of qubits that are needed for a correct execution. Due to this requirement, it is a perfect candidate to use the circuit distribution technique. In [23], a first proposal to use several QPU was made. Firstly, they showed that the QFT could be executed in parallel, substituting each controlled operation with a remote-controlled one, and that the modular exponentiation could be parallelized using a set of QPUs. Although a communication complexity of $\mathbb{O}((log_2 N)^2)$ is needed, being N the number of bits of the number to factorize, and the total number of qubits is increased, the size of each QPU is drastically reduced.

More recently, Gidney et al. [331] analyzed the hardware resources for factoring large numbers, using the Ekerå and Håstad algorithm [332] instead of the Shor one. Applying several optimizations and considering the current methods for making logical qubits, they asserted that a number of 2048 bits can be factorized in 8 h with 20 million noisy qubits (if the operations work in the range of nanoseconds). However, due to the capabilities of the implemented additions needed to factorize the number, the qubits can be reduced to 11 million for each QPU when 2 are used and to 4 million for 8 QPUs. They require a quantum network with a low (but efficient) bandwidth of 150 qb/s. Later, Xiao et al. [333] presented a parallel algorithm that reduces the number of needed qubits, dividing the algorithm between several QPUs, each calculating one subset of the bits. Although it uses several QPUs, it is sequential because to guarantee that the correct state is used on each step, it is teletransported between them at the end of each step.

More well-known quantum algorithms have been parallelized. For example, Neumann et al. [334] studied the Quantum Phase Estimation algorithm using a remote-controlled operation. They compared two possible approaches. The first one is called standard (or automatic), where each controlled operation in the standard QFT is replaced by a remote-controlled operation. This case needs n^2 entangled pairs to execute. The second approach uses the iterative nature of the QFT, aggregating all controlled operations by a single qubit in a unique transport operation. In this case, the number of transport operations is reduced to n. They used a simulator for the experiments, introducing different noise levels to create entangled pairs. The results obtained are similar for both approaches, given the last systematically better results. This experiment showed that automatic partitioning of the problems must take care of possible optimizations and multiple usage of a single pair. One important point is that they studied only the effect of imperfect entangling in the needed pairs without considering other errors such as the measurement, controlled operations between the pairs, and the QPU qubits, etc.

Also, Van Meter et al. [335] studied some of the possible arithmetic operations using teledata and telegate methods in different distributed topologies. They found that, for these problems, the teledata outperforms the telegate method and that a linear architecture is the best choice. In [336], Tan et al. described a parallel algorithm for Simon's problem that still keeps the exponential scaling compared to the classical algorithm.

Recently, Li et al. [337] presented a family of distributed quantum algorithms for the classical Deutsch-Jozsa problem. These algorithms are based on a set of computers with remote communications. However, in the current description, the nature is still sequential, without a clear path to reduce the global depth and time. Finally, Shi et al. [284] made a first proof of concept of using QMPI for the Quantum Phase Estimation and Trotter time evolution, but without including real quantum communications.

5.2. Circuit cutting and other hybrid applications

As described in Section 4.1.2, algorithms based on circuit-cutting only need classical communications to calculate the final solution. Automatic cutting of a circuit (in space or time) is feasible when the number of control operations to cut is limited. However, it is also possible to use non-automatic clever designs to divide a single problem (usually executed using a single quantum circuit) in the execution of several independent quantum programs that later are combined classically to find the right solution.

As already mentioned in the introduction, the paper from Yepez [24] was one of the first proposals to analyze this parallel computation in a hybrid scheme. He considered the case of a system composed of quantum nodes but exclusively connected by a classical network. He named this architecture type-II quantum architecture to differentiate it from the monolithic quantum processors (or type-I), which maintain the global phase coherence. His proposal suggested that some problems need only short spatial and time entanglement, as some kinds of molecules. So they are tractable in parallel quantum computers, unlike other algorithms that need long and spatially large entanglement. For solving those problems, there are three assumptions: first, that the wave function is separable, i.e., can be expressed as a tensor product of subwave functions, each of them residing in one QPU; second, that we can apply a projection operator simultaneously on each qubit of each QPU; and, third, that this projection can be applied after each time step. Yepez proposes a quantum computer composed of many small QPUs arranged in a regular periodic lattice, where local operations are applied to the local qubits simultaneously across the lattice. He applies this proposal to solve problems with lattice gases. For small QPUs, the problems could be tractable using modern Tensor Networks techniques.

In [338,339], Zhou et al. presented distributed quantum algorithms for the Bernstein-Vazirani classical problem and the Grover search, respectively. They divide the binary functions used in the algorithms into a set of subfunctions that can be executed in parallel, getting the final result by composing the different binary parts. In the case of Grover's search, the algorithm only works when a single solution exists, while the extension is still open to multiple solutions. Similarly, Avron et al. [340] studied Deutsch-Jozsa's, Simon's, and Grover's on a distributed environment, finding that, for these algorithms, there are still advantages when comparing with the classical solutions, being the advantage reduced when compared with the fault-tolerant versions. But since these distributed algorithms require shallow circuits, they may be a short-term solution in today's NISQ era.

Several parallel versions of VQAs also use circuit-cutting techniques. For example, [224] used a circuit-cutting based VQE to calculate the ground state of BeH_2. Eddins et al. [221] presented another kind of methodology. They use the Schmidt decomposition to divide a chemical problem of $2N$ qubits in several circuits that need only N qubits, applying VQE to those and joining the results to calculate the final value of the observable. Fujii et al. [341] proposed another method to divide the problem into smaller cases that are combined hierarchically to find the final solution. The technique can be applied when the problem has some structure that aggregates the entanglement in clusters that can be linked later at a higher level. They applied the technique to a kagome lattice, using several layers of aggregation. This technique could also be used in a hybrid scheme, where part of the calculation is done by QPUs at the first steps, and later, the system is solved by a classical computer using tensor networks.

The usage of these divide-and-conquer techniques can also be applied to combinatorial optimization, where a larger problem can be solved using several computers [328,342], and to Quantum Machine Learning (QML). Marshall et al. [343] examine it for the case of classification. They found that automatic circuit cutting could avoid executing all the subcircuits because some of them do not contribute significantly to the final result and proposed a small change in the process that permits the achievement of results close to the classical Neural Networks.

5.3. Embarrassingly parallel applications

The cutting techniques presented in the previous section convert a complex problem into an example of an embarrassingly parallel application, where each smaller circuit can be executed in parallel, later combining the results classically. Other examples of these kinds of applications are [344,345], which studied the use of partial diffusion operator [346] for Grover's search algorithm. The use of this technique does not reduce the number of required qubits but presents some advantages because each circuit is smaller in depth (and, consequently, needs less time to execute in parallel), and the angles of rotations are bigger, reducing the errors in current quantum devices.

Other quantum algorithms, such as the Phase Estimation for a single phase, can be executed using this formalism [347] because it is possible to split the algorithm into several smaller circuits and combine the results classically at the end. Other classical quantum algorithms, such as the Amplitude Estimation, require large resources that can be approximated by distributing several smaller tasks and post-processing classically their results [348].

In order to get the maximum profit from the available distributed infrastructure or, in the short term, to permit the calculation of VQAs, a combination of the aforementioned techniques can be applied. For instance, DiAdamo et al. [349] proposed placing some circuits needed for calculating the expectation value on available QPUs, using the remaining free qubits to create a distributed version of the Ansatz. Alternatively, the Ansatz could be split using the circuit cutting technique.

6. Final remarks and open challenges

Distributed quantum computing emerges as a clear pathway to enhance the computational capabilities of current quantum systems. In this work, we have presented a comprehensive survey of this field's current state of the art. Using a four-layered model – physical, network, development, and application –, we have guided readers to explore its foundational principles, achievements, challenges, and promising directions for further research. Next, we conclude this work by summarizing some of the most important open challenges in the DQC field:

- **Quantum Teleportation**: It is the most fundamental mechanism required at the physical layer for distributed algorithms in DQC applications. Two types of teleportation protocols are essential: gate teleportation (telegate) and qubit state teleportation (teledata). Telegate enables the remote execution of quantum gates on entangled qubits, allowing quantum information to be manipulated without direct physical interaction. Teledata allows an unknown quantum state processed at one network node to be transmitted to a remote location.

  Open challenges: Enhancing the fidelity of these protocols is an active area of research, as high fidelity is critical for ensuring quantum-computational accuracy in future distributed quantum computers.
- **Quantum Networks**: To achieve interconnected, datacenter-scale QPUs, quantum networks must enable entanglement distribution between any two nodes in the network. Current scalable proposals suggest using quantum networking devices such as repeaters, switches, and routers. These devices support the pre-establishment of entangled qubits through transduction to flying qubits and successive entanglement distribution to end nodes, where computation occurs.

  Open challenges: Such devices must include registers of qubits and implement a limited quantum operation instruction set to execute entanglement distillation, swapping, and teleportation protocols. These advancements are essential to unlock true deterministic DQC architectures. Alternative approaches based on teledata operation with single flying qubits instead of EPR pairs could simplify network architectures. However, further research is required to match the fidelity and efficiency of current entanglement-based protocols. On the other hand, from a practical and market-oriented perspective, current quantum networking solutions are costly and lack the required performance, fidelity, and robustness. Higher-level aspects remain in the early stages of research, including developing networking protocols, scalable connectivity architectures, and robust systems. Auxiliary protocols for synchronization, resource management for entanglement distribution, network service definition, error correction, and qubit encoding must still be developed to achieve fault-tolerant, highly available, and performant quantum networks suitable for distributed quantum computing.
- **Circuit Cutting**: In the current noisy and limited QPUs scenario, circuit cutting can be a useful tool for solving large problems with small quantum computers by distributing parts of the circuit between them without requiring a fully realized quantum network.

  Open challenges: The cost associated with this technique scales exponentially with the amount of cut entanglement between the parts, and, for general quantum circuits, entanglement may have a very complex structure that is unknown beforehand. Some improvements have been proposed, which could avoid the execution of a large fraction of the subcircuits, thereby reducing the computing requirement. Nonetheless, there are criticisms about the overall utility of these techniques. Moreover, dividing circuits and executing them on different QPUs requires a better understanding of the effect of different noise profiles on each QPU. Additionally, when different architectures are employed, the execution times must be carefully managed.
- **Compilers**: Using agnostic compilers to find the best partitions for a general algorithm is similar to auto-parallelism in classical computing, which scales poorly. Designing problems that are easier to partition, such as well-designed ansatzes for variational quantum algorithms or problems tailored for modular architectures, may be more effective. In addition to automatic circuit-breaking tools, experienced programmers can develop methods for dividing and parallelizing algorithms. Tools like QMPI or frameworks for distributing programs are also necessary.

  Open challenges: Research is needed to improve agnostic compilers, develop more efficient partitioning methods, and create tools that enable programmers to parallelize quantum computations across different quantum processors efficiently.
- **Applications**: Embarrassingly parallel applications or those based on circuit knitting are the most widely used solutions in the current NISQ era.

  Open challenges: Further research is needed to develop high-level parallel programming models for distributed quantum computing that efficiently use future quantum networks.

It can be concluded from this work that distributed quantum computing offers a promising way to overcome the limitations of current quantum systems by connecting and scaling quantum processors.

While significant challenges remain – such as improving teleportation fidelity, developing scalable networks and optimizing compilers – advances in these areas will facilitate the path towards robust and fault-tolerant quantum computing, unlocking unprecedented computational capabilities.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the authors used ChatGPT in order to improve language and readability. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the content of the publication.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Tomas F. Pena reports financial support and article publishing charges were provided by European Union. Tomas F. Pena reports financial support was provided by Goverment of Spain MINECO. Tomas F. Pena reports financial support was provided by Government of Galicia. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

This work was supported by MICINN through the European Union NextGenerationEU recovery plan (PRTR-C17.I1), the Galician Regional Government through "Planes Complementarios de I+D+I con las Comunidades Autónomas" in Quantum Communication, MINECO (grants PID2019-104834GB-I00, PID2022-141623NB-I00 and PID2022-137061OB- C22), Consellería de Cultura, Educación e Ordenación Universitaria (accreditations ED431C 2022/16 and ED431G-2019/04), and the European Regional Development Fund (ERDF).

Data availability

No data was used for the research described in the article.

References

[1] R. Van Meter, W.J. Munro, K. Nemoto, Architecture of a quantum multicomputer implementing shor's algorithm, in: Theory of Quantum Computation, Communication, and Cryptography, Springer, 2008, pp. 105–114, http://dx.doi.org/10.1007/978-3-540-89304-2_10.

[2] QuEra, Benefits of modular quantum computing for business, 2023, URL https://www.quera.com/blog-posts/benefits-of-modular-quantum-computing-for-business. (Accessed 21 January 2025).

[3] Quantum news, Distributed quantum computing: Scaling quantum power with multiple processors and noise simulation, 2024, URL https://quantumzeitgeist.com/distributed-quantum-computing-scaling-quantum-power-with-multiple-processors-and-noise-simulation. (Accessed 21 January 2025).

[4] IonQ, IonQ achieves critical first step towards developing future quantum networks, 2024, URL https://ionq.com/news/ionq-achieves-critical-first-step-towards-developing-future-quantum-networks. (Accessed 21 January 2025).

[5] N. Saurabh, S. Jha, A. Luckow, A conceptual architecture for a Quantum-HPC middleware, in: Int. Conf. on Quantum Software, QSW, IEEE, 2023, pp. 116–127, http://dx.doi.org/10.1109/QSW59989.2023.00023.

[6] K. Wintersperger, H. Safi, W. Mauerer, QPU-system co-design for quantum HPC accelerators, in: Architecture of Computing Systems, ARCS 2022, in: Lecture Notes in Computer Science, vol. 13642, Springer, 2022, pp. 100–114, http://dx.doi.org/10.1007/978-3-031-21867-5_7.

[7] J. Vázquez-Pérez, C. Piñeiro, J.C. Pichel, T.F. Pena, A. Gómez, QPU integration in OpenCL for heterogeneous programming, J. Supercomput. (2024) 1–22, http://dx.doi.org/10.1007/s11227-023-05879-9.

[8] A.J. McCaskey, D.I. Lyakh, E.F. Dumitrescu, S.S. Powers, T.S. Humble, XACC: A system-level software infrastructure for heterogeneous quantum-classical computing, 2019, http://dx.doi.org/10.48550/arXiv.1911.02452, arxiv preprint.

[9] A.J. McCaskey, T. Nguyen, A. Santana, D. Claudino, T. Kharazi, H. Finkel, Extending C++ for heterogeneous quantum-classical computing, ACM Trans. Quantum Comput. 2 (2) (2021) 1–36, http://dx.doi.org/10.1145/3462670.

[10] J. Gambetta, Quantum-centric supercomputing: The next wave of computing, 2022, URL https://www.ibm.com/quantum/blog/next-wave-quantum-centric-supercomputing. (Accessed 21 January 2025).

[11] J. Gambetta, The hardware and software for the era of quantum utility is here, 2023, URL https://research.ibm.com/blog/quantum-roadmap-2033. (Accessed 21 January 2025).

[12] L.K. Grover, Quantum telecomputation, 1997, http://dx.doi.org/10.48550/arXiv.quant-ph/9704012, arXiv.

[13] R. Cleve, H. Buhrman, Substituting quantum entanglement for communication, Phys. Rev. A 56 (2) (1997) 1201–1204, http://dx.doi.org/10.1103/PhysRevA.56.1201.

[14] J.I. Cirac, A.K. Ekert, S.F. Huelga, C. Macchiavello, Distributed quantum computation over noisy channels, Phys. Rev. A 59 (1999) 4249–4254, http://dx.doi.org/10.1103/PhysRevA.59.4249.

[15] J. Eisert, K. Jacobs, P. Papadopoulos, M.B. Plenio, Optimal local implementation of nonlocal quantum gates, Phys. Rev. A 62 (5) (2000) http://dx.doi.org/10.1103/PhysRevA.62.052317.

[16] D. Collins, N. Linden, S. Popescu, Nonlocal content of quantum operations, Phys. Rev. A - At. Mol. Opt. Phys. 64 (2001) 7, http://dx.doi.org/10.1103/PhysRevA.64.032302.

[17] D.P. DiVincenzo, The physical implementation of quantum computation, Fortschr. Phys. 48 (2000) 771–783, http://dx.doi.org/10.1002/1521-3978(200009)48:9/11<771::AID-PROP771>3.0.CO;2-E.

[18] Y.L. Lim, A. Beige, L.C. Kwek, Repeat-until-success linear optics distributed quantum computing, Phys. Rev. Lett. 95 (2005) http://dx.doi.org/10.1103/PhysRevLett.95.030505.

[19] A. Serafini, S. Mancini, S. Bose, Distributed quantum computation via optical fibres, Phys. Rev. Lett. 96 (2006) http://dx.doi.org/10.1103/PhysRevLett.96.010503.

[20] L. Jiang, J.M. Taylor, A.S. Sørensen, M.D. Lukin, Distributed quantum computation based on small quantum registers, Phys. Rev. A - At. Mol. Opt. Phys. 76 (6) (2007) http://dx.doi.org/10.1103/PhysRevA.76.062323.

[21] D.K.L. Oi, S.J. Devitt, L.C.L. Hollenberg, Scalable error correction in distributed ion trap computers, Phys. Rev. A 74 (2006) http://dx.doi.org/10.1103/physreva.74.052313.

[22] M. Gupta, A. Pathak, A scheme for distributed quantum search through simultaneous state transfer mechanism, Ann. Phys., Lpz. 16 (12) (2007) 791–797, http://dx.doi.org/10.1002/andp.200710265.

[23] A. Yimsiriwattana, S.J. Lomonaco Jr., Distributed quantum computing: A distributed shor algorithm, Proc. SPIE 5436 (2004) 360–372, http://dx.doi.org/10.1117/12.546504.

[24] J. Yepez, Type-II quantum computers, Internat. J. Modern Phys. C 12 (9) (2001) 1273–1284, http://dx.doi.org/10.1142/S0129183101002668.

[25] R. Jozsa, N. Linden, On the role of entanglement in quantum-computational speed-up, Proc. R. Soc. A: Math. Phys. Eng. Sci. 459 (2003) 2011–2032, http://dx.doi.org/10.1098/rspa.2002.1097.

[26] M. Caleffi, M. Amoretti, D. Ferrari, J. Illiano, A. Manzalini, A.S. Cacciapuoti, Distributed quantum computing: a survey, Comput. Netw. 254 (2024) 110672, http://dx.doi.org/10.1016/j.comnet.2024.110672.

[27] S. Rodrigo, S. Abadal, E. Alarcon, M. Bandic, H.V. Someren, C.G. Almudever, On double full-stack communication-enabled architectures for multicore quantum computers, IEEE Micro 41 (5) (2021) 48–56, http://dx.doi.org/10.1109/MM.2021.3092706.

[28] D. Cuomo, M. Caleffi, A.S. Cacciapuoti, Towards a distributed quantum computing ecosystem, IET Quantum Commun. 1 (1) (2020) 3–8, http://dx.doi.org/10.1049/iet-qtc.2020.0002.

[29] W.K. Wootters, W.H. Zurek, A single quantum cannot be cloned, Nature 299 (5886) (1982) 802–803, http://dx.doi.org/10.1038/299802a0.

[30] R. Horodecki, P. Horodecki, M. Horodecki, K. Horodecki, Quantum entanglement, Rev. Modern Phys. 81 (2009) 865–942, http://dx.doi.org/10.1103/RevModPhys.81.865.

[31] C.C. Gerry, P.L. Knight, Introductory Quantum Optics, Cambridge Universiy Press, 2005, http://dx.doi.org/10.1017/CBO9780511791239.

[32] P. Andres-Martinez, Towards Distributed Quantum Algorithms (Master's thesis), School of Informatics, University of Edinburgh, 2018, URL https://project-archive.inf.ed.ac.uk/msc/20183076/msc_proj.pdf.

[33] C.H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres, W.K. Wootters, Teleporting an unknown quantum state via dual classical and Einstein-Podolsky-Rosen channels, Phys. Rev. Lett. 70 (13) (1993) 1895–1899, http://dx.doi.org/10.1103/PhysRevLett.70.1895.

[34] A. Acín, I. Bloch, H. Buhrman, T. Calarco, C. Eichler, J. Eisert, D. Esteve, N. Gisin, S.J. Glaser, F. Jelezko, S. Kuhr, M. Lewenstein, M.F. Riedel, P.O. Schmidt, R. Thew, A. Wallraff, I. Walmsley, F.K. Wilhelm, The quantum technologies roadmap: A European community view, New J. Phys. 20 (2018) http://dx.doi.org/10.1088/1367-2630/aad1ea.

[35] A. Karlsson, M. Bourennane, Quantum teleportation using three-particle entanglement, Phys. Rev. A 58 (1998) 4394–4400, http://dx.doi.org/10.1103/PhysRevA.58.4394.

[36] M. Żukowski, A. Zeilinger, M.A. Horne, A.K. Ekert, “Event-ready-detectors” bell experiment via entanglement swapping, Phys. Rev. Lett. 71 (26) (1993) 4287–4290, http://dx.doi.org/10.1103/PhysRevLett.71.4287.

[37] H.-J. Briegel, W. Dür, J.I. Cirac, P. Zoller, Quantum repeaters: The role of imperfect local operations in quantum communication, Phys. Rev. Lett. 81 (1998) 5932–5935, http://dx.doi.org/10.1103/PhysRevLett.81.5932.

[38] J.-G. Ren, P. Xu, H.-L. Yong, L. Zhang, S.-K. Liao, J. Yin, W.-Y. Liu, W.-Q. Cai, M. Yang, L. Li, K.-X. Yang, X. Han, Y.-Q. Yao, J. Li, H.-Y. Wu, S. Wan, L. Liu, D.-Q. Liu, Y.-W. Kuang, Z.-P. He, P. Shang, C. Guo, R.-H. Zheng, K. Tian, Z.-C. Zhu, N.-L. Liu, C.-Y. Lu, R. Shu, Y.-A. Chen, C.-Z. Peng, J.-Y. Wang, J.-W. Pan, Ground-to-satellite quantum teleportation, Nature 549 (7670) (2017) 70–73, http://dx.doi.org/10.1038/nature23675.

[39] D. Gottesman, I.L. Chuang, Demonstrating the viability of universal quantum computation using teleportation and single-qubit operations, Nature 402 (6760) (1999) 390–393, http://dx.doi.org/10.1038/46503.

[40] R. Raussendorf, H.J. Briegel, A one-way quantum computer, Phys. Rev. Lett. 86 (2001) 5188–5191, http://dx.doi.org/10.1103/PhysRevLett.86.5188.

[41] K.S. Chou, J.Z. Blumoff, C.S. Wang, P.C. Reinhold, C.J. Axline, Y.Y. Gao, L. Frunzio, M.H. Devoret, L. Jiang, R.J. Schoelkopf, Deterministic teleportation of a quantum gate between two logical qubits, Nature 561 (7723) (2018) 368–373, http://dx.doi.org/10.1038/s41586-018-0470-y.

[42] D. Bouwmeester, J.-W. Pan, K. Mattle, M. Eibl, H. Weinfurter, A. Zeilinger, Experimental quantum teleportation, Nature 390 (6660) (1997) 575–579, http://dx.doi.org/10.1038/37539.

[43] A. Furusawa, J.L. Sørensen, S.L. Braunstein, C.A. Fuchs, H.J. Kimble, E.S. Polzik, Unconditional quantum teleportation, Science 282 (5389) (1998) 706–709, http://dx.doi.org/10.1126/science.282.5389.706.

[44] J.F. Sherson, H. Krauter, R.K. Olsson, B. Julsgaard, K. Hammerer, I. Cirac, E.S. Polzik, Quantum teleportation between light and matter, Nature 443 (2006) 557–560, http://dx.doi.org/10.1038/nature05136.

[45] M. Nielsen, E. Knill, R. Laflamme, Complete quantum teleportation using nuclear magnetic resonance, Nature 396 (1998) 52–55, http://dx.doi.org/10.1038/23891.

[46] M. Riebe, H. Häffner, C.F. Roos, W. Hänsel, J. Benheim, G.P. Lancaster, T.W. Körber, C. Becher, F. Schmidt-Kaler, D.F. James, R. Blatt, Deterministic quantum teleportation with atoms, Nature 429 (2004) 734–737, http://dx.doi.org/10.1038/nature02570.

[47] M.D. Barrett, J. Chiaverini, T. Schaetz, J. Britton, W.M. Itano, J.D. Jost, E. Knill, C. Langer, D. Leibfried, R. Ozeri, D.J. Wineland, Deterministic quantum teleportation of atomic qubits, Nature 429 (2004) 737–739, http://dx.doi.org/10.1038/nature02608.

[48] L. Steffen, Y. Salathe, M. Oppliger, P. Kurpiers, M. Baur, C. Lang, C. Eichler, G. Puebla-Hellmann, A. Fedorov, A. Wallraff, Deterministic quantum teleportation with feed-forward in a solid state system, Nature 500 (2013) 319–322, http://dx.doi.org/10.1038/nature12422.

[49] X.L. Wang, X.D. Cai, Z.E. Su, M.C. Chen, D. Wu, L. Li, N.L. Liu, C.Y. Lu, J.W. Pan, Quantum teleportation of multiple degrees of freedom of a single photon, Nature 518 (2015) 516–519, http://dx.doi.org/10.1038/nature14246.

[50] X.M. Hu, C. Zhang, B.H. Liu, Y. Cai, X.J. Ye, Y. Guo, W.B. Xing, C.X. Huang, Y.F. Huang, C.F. Li, G.C. Guo, Experimental high-dimensional quantum teleportation, Phys. Rev. Lett. 125 (2020) http://dx.doi.org/10.1103/PhysRevLett.125.230501.

[51] D. Llewellyn, Y. Ding, I.I. Faruque, S. Paesani, D. Bacco, R. Santagati, Y.J. Qian, Y. Li, Y.F. Xiao, M. Huber, M. Malik, G.F. Sinclair, X. Zhou, K. Rottwitt, J.L. O’Brien, J.G. Rarity, Q. Gong, L.K. Oxenlowe, J. Wang, M.G. Thompson, Chip-to-chip quantum teleportation and multi-photon entanglement in silicon, Nat. Phys. 16 (2020) 148–153, http://dx.doi.org/10.1038/s41567-019-0727-x.

[52] J.C. Hoke, M. Ippoliti, E. Rosenberg, D. Abanin, R. Acharya, T.I. Andersen, M. Ansmann, F. Arute, K. Arya, A. Asfaw, J. Atalaya, J.C. Bardin, A. Bengtsson, G. Bortoli, A. Bourassa, J. Bovaird, L. Brill, M. Broughton, B.B. Buckley, D.A. Buell, T. Burger, B. Burkett, N. Bushnell, Z. Chen, B. Chiaro, D. Chik, J. Cogan, R. Collins, P. Conner, W. Courtney, A.L. Crook, B. Curtin, A.G. Dau, D.M. Debroy, A. Del Toro Barba, S. Demura, A. Di Paolo, I.K. Drozdov, A. Dunsworth, D. Eppens, C. Erickson, E. Farhi, R. Fatemi, V.S. Ferreira, L.F. Burgos, E. Forati, A.G. Fowler, B. Foxen, W. Giang, C. Gidney, D. Gilboa, M. Giustina, R. Gosula, J.A. Gross, S. Habegger, M.C. Hamilton, M. Hansen, M.P. Harrigan, S.D. Harrington, P. Heu, M.R. Hoffmann, S. Hong, T. Huang, A. Huff, W.J. Huggins, S.V. Isakov, J. Iveland, E. Jeffrey, Z. Jiang, C. Jones, P. Juhas, D. Kafri, K. Kechedzhi, T. Khattar, M. Khezri, M. Kieferová, S. Kim, A. Kitaev, P.V. Klimov, A.R. Klots, A.N. Korotkov, F. Kostritsa, J.M. Kreikebaum, D. Landhuis, P. Laptev, K.-M. Lau, L. Laws, J. Lee, K.W. Lee, Y.D. Lensky, B.J. Lester, A.T. Lill, W. Liu, A. Locharla, O. Martin, J.R. McClean, M. McEwen, K.C. Miao, A. Mieszala, S. Montazeri, A. Morvan, R. Movassagh, W. Mruczkiewicz, M. Neeley, C. Neill, A. Nersisyan, M. Newman, J.H. Ng, A. Nguyen, M. Nguyen, M.Y. Niu, T.E. O’Brien, S. Omonije, A. Opremcak, A. Petukhov, R. Potter, L.P. Pryadko, C. Quintana, C. Rocque, N.C. Rubin, N. Saei, D. Sank, K. Sankaragomathi, K.J. Satzinger, H.F. Schurkus, C. Schuster, M.J. Shearn, A. Shorter, N. Shutty, V. Shvarts, J. Skruzny, W.C. Smith, R. Somma, G. Sterling, D. Strain, M. Szalay, A. Torres, G. Vidal, B. Villalonga, C.V. Heidweiller, T. White, B.W.K. Woo, C. Xing, Z.J. Yao, P. Yeh, J. Yoo, G. Young, A. Zalcman, Y. Zhang, N. Zhu, N. Zobrist, H. Neven, R. Babbush, D. Bacon, S. Boixo, J. Hilton, E. Lucero, A. Megrant, J. Kelly, Y. Chen, V. Smelyanskiy, X. Mi, V. Khemani, P. Roushan, G.Q. AI, Collaborators, Measurement-induced entanglement and teleportation on a noisy quantum processor, Nature 622 (7983) (2023) 481–486, http://dx.doi.org/10.1038/s41586-023-06505-7.

[53] A. Yimsiriwattana, S.J. Lomonaco Jr., Generalized GHZ states and distributed quantum computing, 2004, http://dx.doi.org/10.48550/arXiv.quant-ph/0402148, arXiv.

[54] C.H. Bennett, D.P. DiVincenzo, P.W. Shor, J.A. Smolin, B.M. Terhal, W.K. Wootters, Remote state preparation, Phys. Rev. Lett. 87 (2001) http://dx.doi.org/10.1103/PhysRevLett.87.077902.

[55] H. Weinfurter, Experimental bell-state analysis, Europhys. Lett. 25 (1994) 559–564, http://dx.doi.org/10.1209/0295-5075/25/8/001.

[56] S. Massar, S. Popescu, Optimal extraction of information from finite quantum ensembles, Phys. Rev. Lett. 74 (1995) 1259–1263, http://dx.doi.org/10.1103/PhysRevLett.74.1259.

[57] S. Pirandola, J. Eisert, C. Weedbrook, A. Furusawa, S.L. Braunstein, Advances in quantum teleportation, Nat. Photonics 9 (2015) 641–652, http://dx.doi.org/10.1038/nphoton.2015.154.

[58] X.M. Hu, Y. Guo, B.H. Liu, C.F. Li, G.C. Guo, Progress in quantum teleportation, Nat. Rev. Phys. 5 (2023) 339–353, http://dx.doi.org/10.1038/s42254-023-00588-x.

[59] Q.C. Sun, Y.L. Mao, S.J. Chen, W. Zhang, Y.F. Jiang, Y.B. Zhang, W.J. Zhang, S. Miki, T. Yamashita, H. Terai, X. Jiang, T.Y. Chen, L.X. You, X.F. Chen, Z. Wang, J.Y. Fan, Q. Zhang, J.W. Pan, Quantum teleportation with independent sources and prior entanglement distribution over a network, Nat. Photonics 10 (2016) 671–675, http://dx.doi.org/10.1038/nphoton.2016.179.

[60] R. Valivarthi, M.G. Puigibert, Q. Zhou, G.H. Aguilar, V.B. Verma, F. Marsili, M.D. Shaw, S.W. Nam, D. Oblak, W. Tittel, Quantum teleportation across a metropolitan fibre network, Nat. Photonics 10 (2016) 676–680, http://dx.doi.org/10.1038/nphoton.2016.180.

[61] C.M. Knaut, A. Suleymanzade, Y.-C. Wei, D.R. Assumpcao, P.-J. Stas, Y.Q. Huan, B. Machielse, E.N. Knall, M. Sutula, G. Baranes, N. Sinclair, C. De-Eknamkul, D.S. Levonian, M.K. Bhaskar, H. Park, M. Lončar, M.D. Lukin, Entanglement of nanophotonic quantum memory nodes in a telecommunication network, 2023, http://dx.doi.org/10.48550/arXiv.2310.01316, arXiv eprint.

[62] V. Krutyanskiy, M. Galli, V. Krcmarsky, S. Baier, D.A. Fioretto, Y. Pu, A. Mazloom, P. Sekatski, M. Canteri, M. Teller, J. Schupp, J. Bate, M. Meraner, N. Sangouard, B.P. Lanyon, T.E. Northup, Entanglement of trapped-ion qubits separated by 230 meters, Phys. Rev. Lett. 130 (2023) http://dx.doi.org/10.1103/PhysRevLett.130.050803.

[63] J.-L. Liu, X.-Y. Luo, Y. Yu, C.-Y. Wang, B. Wang, Y. Hu, J. Li, M.-Y. Zheng, B. Yao, Z. Yan, D. Teng, J.-W. Jiang, X.-B. Liu, X.-P. Xie, J. Zhang, Q.-H. Mao, X. Jiang, Q. Zhang, X.-H. Bao, J.-W. Pan, A multinode quantum network over a metropolitan area, 2023, http://dx.doi.org/10.48550/arXiv.2309.00221, arXiv eprint.

[64] S. Daiss, S. Langenfeld, S. Welte, E. Distante, P. Thomas, L. Hartung, O. Morin, G. Rempe, A quantum-logic gate between distant quantum-network modules, Science 371 (6529) (2021) 614–617, http://dx.doi.org/10.1126/science.abe3150.

[65] S. Langenfeld, S. Welte, L. Hartung, S. Daiss, P. Thomas, O. Morin, E. Distante, G. Rempe, Quantum teleportation between remote qubit memories with only a single photon as a resource, Phys. Rev. Lett. 126 (2021) 130502, http://dx.doi.org/10.1103/PhysRevLett.126.130502.

[66] Y. Wan, D. Kienzler, S. Erickson, K. Mayer, T. Tan, J. Wu, H. Vasconcelos, S. Glancy, E. Knill, D. Wineland, A. Wilson, D. Leibfried, Quantum gate teleportation between separated qubits in a trapped-ion processor, Science 364 (2019) 875–878, http://dx.doi.org/10.1126/science.aaw9415.

[67] D. Lago-Rivera, S. Grandi, J. Rakonjac, A. Seri, H. de Riedmatten, Telecom-heralded entanglement between multimode solid-state quantum memories, Nature 594 (2021) 37–40, http://dx.doi.org/10.1038/s41586-021-03481-8.

[68] C.H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J.A. Smolin, W.K. Wootters, Purification of noisy entanglement and faithful teleportation via noisy channels, Phys. Rev. Lett. 76 (1996) 722–725, http://dx.doi.org/10.1103/PhysRevLett.76.722.

[69] D.E. Browne, T. Rudolph, Resource-efficient linear optical quantum computation, Phys. Rev. Lett. 95 (2005) http://dx.doi.org/10.1103/PhysRevLett.95.010501.

[70] S. Bartolucci, P. Birchall, H. Bombín, H. Cable, C. Dawson, M. Gimeno-Segovia, E. Johnston, K. Kieling, N. Nickerson, M. Pant, F. Pastawski, T. Rudolph, C. Sparrow, Fusion-based quantum computation, Nat. Commun. 14 (2023) http://dx.doi.org/10.1038/s41467-023-36493-1.

[71] J.-W. Pan, D. Bouwmeester, H. Weinfurter, A. Zeilinger, Experimental entanglement swapping: Entangling photons that never interacted, Phys. Rev. Lett. 80 (1998) 3892–3894, http://dx.doi.org/10.1103/PhysRevLett.80.3891.

[72] S.L.N. Hermans, M. Pompili, H.K.C. Beukers, S. Baier, J. Borregaard, R. Hanson, Qubit teleportation between non-neighbouring nodes in a quantum network, Nature 605 (7911) (2022) 663–668, http://dx.doi.org/10.1038/s41586-022-04697-y.

[73] Y.F. Huang, X.F. Ren, Y.S. Zhang, L.M. Duan, G.C. Guo, Experimental teleportation of a quantum controlled-NOT gate, Phys. Rev. Lett. 93 (2004) http://dx.doi.org/10.1103/PhysRevLett.93.240501.

[74] J.-W. Pan, D. Bouwmeester, M. Daniel, H. Weinfurter, A. Zeilinger, Experimental test of quantum nonlocality in three-photon Greenberger-Horne-Zeilinger entanglement, Nature 403 (2000) 515–519, http://dx.doi.org/10.1038/35000514.

[75] M. Murao, D. Jonathan, M.B. Plenio, V. Vedral, Quantum telecloning and multiparticle entanglement, Phys. Rev. A 59 (1999) 156–161, http://dx.doi.org/10.1103/PhysRevA.59.156.

[76] Z. Zhao, Y.-A. Chen, A.-N. Zhang, T. Yang, H.J. Briegel, J.-W. Pan, Experimental demonstration of five-photon entanglement and open-destination teleportation, Nature 430 (6995) (2004) 54–58, http://dx.doi.org/10.1038/nature02643.

[77] S.M. Lee, S.W. Lee, H. Jeong, H.S. Park, Quantum teleportation of shared quantum secret, Phys. Rev. Lett. 124 (2020) http://dx.doi.org/10.1103/PhysRevLett.124.060501.

[78] Z. Zhao, A.N. Zhang, X.Q. Zhou, Y.A. Chen, C.Y. Lu, A. Karlsson, J.W. Pan, Experimental realization of optimal asymmetric cloning and telecloning via partial teleportation, Phys. Rev. Lett. 95 (2005) http://dx.doi.org/10.1103/PhysRevLett.95.030502.

[79] L.C. Peng, D. Wu, H.S. Zhong, Y.H. Luo, Y. Li, Y. Hu, X. Jiang, M.C. Chen, L. Li, N.L. Liu, K. Nemoto, W.J. Munro, B.C. Sanders, C.Y. Lu, J.W. Pan, Cloning of quantum entanglement, Phys. Rev. Lett. 125 (2020) http://dx.doi.org/10.1103/PhysRevLett.125.210502.

[80] Q. Wang, Y. Wang, X. Sun, Y. Tian, W. Li, L. Tian, X. Yu, J. Zhang, Y. Zheng, Controllable continuous variable quantum state distributor, Opt. Lett. 46 (2021) 1844, http://dx.doi.org/10.1364/ol.419261.

[81] D. Awschalom, K.K. Berggren, et al., Development of quantum interconnects (QuICs) for next-generation information technologies, PRX Quantum 2 (2021) http://dx.doi.org/10.1103/PRXQuantum.2.017002.

[82] H. Edlbauer, J. Wang, T. Crozes, P. Perrier, S. Ouacel, C. Geffroy, G. Georgiou, E. Chatzikyriakou, A. Lacerda-Santos, X. Waintal, D.C. Glattli, P. Roulleau, J. Nath, M. Kataoka, J. Splettstoesser, M. Acciai, M.C. da Silva Figueira, K. Öztas, A. Trellakis, T. Grange, O.M. Yevtushenko, S. Birner, C. Bäuerle, Semiconductor-based electron flying qubits: Review on recent progress accelerated by numerical modelling, EPJ Quantum Technol. 9 (1) (2022) 21, http://dx.doi.org/10.1140/epjqt/s40507-022-00139-w.

[83] M. Zhong, M.P. Hedges, R.L. Ahlefeldt, J.G. Bartholomew, S.E. Beavan, S.M. Wittig, J.J. Longdell, M.J. Sellars, Optically addressable nuclear spins in a solid with a six-hour coherence time, Nature 517 (7533) (2015) 177–180, http://dx.doi.org/10.1038/nature14025.

[84] P. Hurst, A. Miller, Trends in undersea fiber optic systems, in: OCEANS 2000 MTS/IEEE Conference and Exhibition. Conference Proceedings (Cat. No.00CH37158), Vol. 1, 2000, pp. 479–488, http://dx.doi.org/10.1109/OCEANS.2000.881303, vol.1.

[85] K. Grobe, M. Eiselt, Wavelength Division Multiplexing: A Practical Engineering Guide, in: Wiley Series in Pure and Applied Optics, Wiley, 2013, http://dx.doi.org/10.1002/9781118755068.

[86] R.W. Munn, C.N. Ironside, Principles and Applications of Nonlinear Optical Materials, Springer, 1993, http://dx.doi.org/10.1007/978-94-011-2158-3.

[87] M. Barbieri, C. Cinelli, P. Mataloni, F. De Martini, Polarization-momentum hyperentangled states: Realization and characterization, Phys. Rev. A 72 (2005) 052110, http://dx.doi.org/10.1103/PhysRevA.72.052110.

[88] D. Hucul, I.V. Inlek, G. Vittorini, C. Crocker, S. Debnath, S.M. Clark, C. Monroe, Modular entanglement of atomic qubits using photons and phonons, Nat. Phys. 11 (1) (2015) 37–42, http://dx.doi.org/10.1038/nphys3150.

[89] D.L. Moehring, P. Maunz, S. Olmschenk, K.C. Younge, D.N. Matsukevich, L.-M. Duan, C. Monroe, Entanglement of single-atom quantum bits at a distance, Nature 449 (7158) (2007) 68–71, http://dx.doi.org/10.1038/nature06118.

[90] P. Maunz, D.L. Moehring, S. Olmschenk, K.C. Younge, D.N. Matsukevich, C. Monroe, Quantum interference of photon pairs from two remote trapped atomic ions, Nat. Phys. 3 (8) (2007) 538–541, http://dx.doi.org/10.1038/nphys644.

[91] J. Hofmann, M. Krug, N. Ortegel, L. Gérard, M. Weber, W. Rosenfeld, H. Weinfurter, Heralded entanglement between widely separated atoms, Science 337 (6090) (2012) 72–75, http://dx.doi.org/10.1126/science.1221856.

[92] E. Togan, Y. Chu, A.S. Trifonov, L. Jiang, J. Maze, L. Childress, M.V.G. Dutt, A.S. Sørensen, P.R. Hemmer, A.S. Zibrov, M.D. Lukin, Quantum entanglement between an optical photon and a solid-state spin qubit, Nature 466 (7307) (2010) 730–734, http://dx.doi.org/10.1038/nature09256.

[93] S. Castelletto, A. Edmonds, T. Gaebel, J. Rabeau, Production of multiple diamond-based single-photon sources, IEEE J. Sel. Top. Quantum Electron. 18 (6) (2012) 1792–1798, http://dx.doi.org/10.1109/JSTQE.2012.2199283.

[94] N. Kalb, A.A. Reiserer, P.C. Humphreys, J.J.W. Bakermans, S.J. Kamerling, N.H. Nickerson, S.C. Benjamin, D.J. Twitchen, M. Markham, R. Hanson, Entanglement distillation between solid-state quantum network nodes, Science 356 (6341) (2017) 928–932, http://dx.doi.org/10.1126/science.aan0070.

[95] P.C. Humphreys, N. Kalb, J.P.J. Morits, R.N. Schouten, R.F.L. Vermeulen, D.J. Twitchen, M. Markham, R. Hanson, Deterministic delivery of remote entanglement on a quantum network, Nature 558 (7709) (2018) 268–273, http://dx.doi.org/10.1038/s41586-018-0200-5.

[96] C.T. Nguyen, D.D. Sukachev, M.K. Bhaskar, B. Machielse, D.S. Levonian, E.N. Knall, P. Stroganov, R. Riedinger, H. Park, M. Lončar, M.D. Lukin, Quantum network nodes based on diamond qubits with an efficient nanophotonic interface, Phys. Rev. Lett. 123 (2019) http://dx.doi.org/10.1103/PhysRevLett.123.183602.

[97] V. Krutyanskiy, M. Galli, V. Krcmarsky, S. Baier, D.A. Fioretto, Y. Pu, A. Mazloom, P. Sekatski, M. Canteri, M. Teller, J. Schupp, J. Bate, M. Meraner, N. Sangouard, B.P. Lanyon, T.E. Northup, Entanglement of trapped-ion qubits separated by 230 meters, Phys. Rev. Lett. 130 (2023) 050803, http://dx.doi.org/10.1103/PhysRevLett.130.050803.

[98] A.J. Stolk, K.L. van der Enden, M.-C. Slater, I. te Raa-Derckx, P. Botma, J. van Rantwijk, J.J.B. Biemond, R.A.J. Hagen, R.W. Herfst, W.D. Koek, A.J.H. Meskers, R. Vollmer, E.J. van Zwet, M. Markham, A.M. Edmonds, J.F. Geus, F. Elsen, B. Jungbluth, C. Haefner, C. Tresp, J. Stuhler, S. Ritter, R. Hanson, Metropolitan-scale heralded entanglement of solid-state qubits, Sci. Adv. 10 (44) (2024) eadp6442, http://dx.doi.org/10.1126/sciadv.adp6442.

[99] J. Ramette, J. Sinclair, Z. Vendeiro, A. Rudelis, M. Cetina, V. Vuletić, Any-to-any connected cavity-mediated architecture for quantum computing with trapped ions or rydberg arrays, PRX Quantum 3 (2022) http://dx.doi.org/10.1103/PRXQuantum.3.010344.

[100] Y. Arakawa, M.J. Holmes, Progress in quantum-dot single photon sources for quantum information technologies: A broad spectrum overview, Appl. Phys. Rev. 7 (2) (2020) http://dx.doi.org/10.1063/5.0010193.

[101] S. Castelletto, A. Boretti, Perspective on solid-state single-photon sources in the infrared for quantum technology, Adv. Quantum Technol. 6 (10) (2023) 2300145, http://dx.doi.org/10.1002/qute.202300145.

[102] T. Zhong, P. Goldner, Emerging rare-earth doped material platforms for quantum nanophotonics, Nanophotonics 8 (11) (2019) 2003–2015, http://dx.doi.org/10.1515/nanoph-2019-0185.

[103] N. Samkharadze, G. Zheng, N. Kalhor, D. Brousse, A. Sammak, U.C. Mendes, A. Blais, G. Scappucci, L.M.K. Vandersypen, Strong spin-photon coupling in silicon, Science 359 (6380) (2018) 1123–1127, http://dx.doi.org/10.1126/science.aar4054.

[104] P. Kurpiers, P. Magnard, T. Walter, B. Royer, M. Pechal, J. Heinsoo, Y. Salathé, A. Akin, S. Storz, J.-C. Besse, S. Gasparinetti, A. Blais, A. Wallraff, Deterministic quantum state transfer and remote entanglement using microwave photons, Nature 558 (7709) (2018) 264–267, http://dx.doi.org/10.1038/s41586-018-0195-y.

[105] P. Magnard, S. Storz, P. Kurpiers, J. Schär, F. Marxer, J. Lütolf, T. Walter, J.-C. Besse, M. Gabureac, K. Reuer, A. Akin, B. Royer, A. Blais, A. Wallraff, Microwave quantum link between superconducting circuits housed in spatially separated cryogenic systems, Phys. Rev. Lett. 125 (2020) 260502, http://dx.doi.org/10.1103/PhysRevLett.125.260502.

[106] M. Renger, S. Gandorfer, W. Yam, F. Fesquet, M. Handschuh, K.E. Honasoge, F. Kronowetter, Y. Nojiri, M. Partanen, M. Pfeiffer, H. van der Vliet, A.J. Matthews, J. Govenius, R.N. Jabdaraghi, M. Prunnila, A. Marx, F. Deppe, R. Gross, K.G. Fedorov, Cryogenic microwave link for quantum local area networks, 2023, http://dx.doi.org/10.48550/arXiv.2308.12398, arXiv eprint.

[107] D. Lago-Rivera, J.V. Rakonjac, S. Grandi, H.d. Riedmatten, Long distance multiplexed quantum teleportation from a telecom photon to a solid-state qubit, Nat. Commun. 14 (1) (2023) 1889, http://dx.doi.org/10.1038/s41467-023-37518-5.

[108] P. Zhao, M.-Y. Yang, S. Zhu, L. Zhou, W. Zhong, M.-M. Du, Y.-B. Sheng, Generation of hyperentangled state encoded in three degrees of freedom, Sci. China Phys. Mech. Astron. 66 (10) (2023) http://dx.doi.org/10.1007/s11433-023-2164-7.

[109] W. Li, L. Zhang, H. Tan, Y. Lu, S.-K. Liao, J. Huang, H. Li, Z. Wang, H.-K. Mao, B. Yan, Q. Li, Y. Liu, Q. Zhang, C.-Z. Peng, L. You, F. Xu, J.-W. Pan, High-rate quantum key distribution exceeding 110 Mb s–1, Nat. Photonics 17 (5) (2023) 416–421, http://dx.doi.org/10.1038/s41566-023-01166-4.

[110] J. Zhang, E. Zallo, B. Höfer, Y. Chen, R. Keil, M. Zopf, S. Böttner, F. Ding, O.G. Schmidt, Electric-field-induced energy tuning of on-demand entangled-photon emission from self-assembled quantum dots, Nano Lett. 17 (1) (2017) 501–507, http://dx.doi.org/10.1021/acs.nanolett.6b04539.

[111] W. Ou, X. Wang, W. Wei, T. Jin, Y. Zhu, T. Wang, J. Zhang, X. Ou, J. Zhang, Strain tuning self-assembled quantum dots for energy-tunable entangled-photon sources using a photolithographically fabricated microelectromechanical system, ACS Photonics 9 (10) (2022) 3421–3428, http://dx.doi.org/10.1021/acsphotonics.2c01033.

[112] C. Hopfmann, W. Nie, N.L. Sharma, C. Weigelt, F. Ding, O.G. Schmidt, Maximally entangled and gigahertz-clocked on-demand photon pair source, Phys. Rev. B 103 (2021) 075413, http://dx.doi.org/10.1103/PhysRevB.103.075413.

[113] P. Aumann, M. Prilmüller, F. Kappe, L. Ostermann, D. Dalacu, P.J. Poole, H. Ritsch, W. Lechner, G. Weihs, Demonstration and modeling of time-bin entangled photons from a quantum dot in a nanowire, AIP Adv. 12 (5) (2022) 055115, http://dx.doi.org/10.1063/5.0081874.

[114] L. Zhou, Y.-B. Sheng, Purification of logic-qubit entanglement, Sci. Rep. 6 (1) (2016) http://dx.doi.org/10.1038/srep28813.

[115] F. Kaiser, P. Vergyris, A. Martin, D. Aktas, M.P.D. Micheli, O. Alibart, S. Tanzilli, Quantum optical frequency up-conversion for polarisation entangled qubits: Towards interconnected quantum information devices, Opt. Express 27 (18) (2019) 25603–25610, http://dx.doi.org/10.1364/OE.27.025603.

[116] S. Murakami, R. Fujimoto, T. Kobayashi, R. Ikuta, A. Inoue, T. Umeki, S. Miki, F. China, H. Terai, R. Kasahara, T. Mukai, N. Imoto, T. Yamamoto, Quantum frequency conversion using 4-port fiber-pigtailed PPLN module, Opt. Express 31 (18) (2023) 29271–29279, http://dx.doi.org/10.1364/OE.494313.

[117] M.J. Weaver, P. Duivestein, A.C. Bernasconi, S. Scharmer, M. Lemang, T.C.v. Thiel, F. Hijazi, B. Hensen, S. Gröblacher, R. Stockill, An integrated microwave-to-optics interface for scalable quantum computing, Nature Nanotechnology 19 (2) (2023) 166–172, http://dx.doi.org/10.1038/s41565-023-01515-y.

[118] R. Sahu, L. Qiu, W. Hease, G. Arnold, Y. Minoguchi, P. Rabl, J.M. Fink, Entangling microwaves with light, Science 380 (6646) (2023) 718–721, http://dx.doi.org/10.1126/science.adg3812.

[119] T.B. Pittman, B.C. Jacobs, J.D. Franson, Single photons on pseudodemand from stored parametric down-conversion, Phys. Rev. A 66 (2002) 042303, http://dx.doi.org/10.1103/PhysRevA.66.042303.

[120] O. Landry, J.A.W. van Houwelingen, A. Beveratos, H. Zbinden, N. Gisin, Quantum teleportation over the Swisscom telecommunication network, J. Opt. Soc. Am. B 24 (2) (2007) 398–403, http://dx.doi.org/10.1364/JOSAB.24.000398.

[121] P.M. Leung, T.C. Ralph, Quantum memory scheme based on optical fibers and cavities, Phys. Rev. A 74 (2006) http://dx.doi.org/10.1103/PhysRevA.74.022311.

[122] T. Tanabe, M. Notomi, E. Kuramochi, A. Shinya, H. Taniyama, Trapping and delaying photons for one nanosecond in an ultrasmall high-Q photonic-crystal nanocavity, Nat. Photonics 1 (1) (2007) 49–52, http://dx.doi.org/10.1038/nphoton.2006.51.

[123] S.A. Moiseev, B.S. Ham, Photon-echo quantum memory with efficient multipulse readings, Phys. Rev. A 70 (2004) http://dx.doi.org/10.1103/PhysRevA.70.063809.

[124] A.I. Lvovsky, B.C. Sanders, W. Tittel, Optical quantum memory, Nat. Photonics 3 (12) (2009) 706–714, http://dx.doi.org/10.1038/nphoton.2009.231.

[125] M. Guo, S. Liu, W. Sun, M. Ren, F. Wang, M. Zhong, Rare-earth quantum memories: The experimental status quo, Front. Phys. 18 (2) (2023) 21303, http://dx.doi.org/10.1007/s11467-022-1240-8.

[126] A.V. Gorshkov, A. André, M. Fleischhauer, A.S. Sørensen, M.D. Lukin, Universal approach to optimal photon storage in atomic media, Phys. Rev. Lett. 98 (2007) 123601, http://dx.doi.org/10.1103/PhysRevLett.98.123601.

[127] W. Tittel, M. Afzelius, T. Chaneliére, R. Cone, S. Kröll, S. Moiseev, M. Sellars, Photon-echo quantum memory in solid state systems, Laser Photonics Rev. 4 (2) (2010) 244–267, http://dx.doi.org/10.1002/lpor.200810056.

[128] M. Afzelius, C. Simon, H. de Riedmatten, N. Gisin, Multimode quantum memory based on atomic frequency combs, Phys. Rev. A 79 (2009) http://dx.doi.org/10.1103/PhysRevA.79.052329.

[129] A. Ortu, A. Holzäpfel, J. Etesse, M. Afzelius, Storage of photonic time-bin qubits for up to 20 ms in a rare-earth doped crystal, Npj Quantum Inf. 8 (1) (2022) 29, http://dx.doi.org/10.1038/s41534-022-00541-3.

[130] E. Knill, R. Laflamme, G.J. Milburn, A scheme for efficient quantum computation with linear optics, Nature 409 (6816) (2001) 46–52, http://dx.doi.org/10.1038/35051009.

[131] P. Kok, W.J. Munro, K. Nemoto, T.C. Ralph, J.P. Dowling, G.J. Milburn, Linear optical quantum computing with photonic qubits, Rev. Modern Phys. 79 (2007) 135–174, http://dx.doi.org/10.1103/RevModPhys.79.135.

[132] E. Knill, R. Laflamme, W.H. Zurek, Resilient quantum computation, Science 279 (5349) (1998) 342–345, http://dx.doi.org/10.1126/science.279.5349.342.

[133] A.Y. Kitaev, Fault-tolerant quantum computation by anyons, Ann. Physics 303 (1) (2003) 2–30, http://dx.doi.org/10.1016/S0003-4916(02)00018-0.

[134] T.P. Harty, D.T.C. Allcock, C.J. Ballance, L. Guidoni, H.A. Janacek, N.M. Linke, D.N. Stacey, D.M. Lucas, High-fidelity preparation, gates, memory, and readout of a trapped-ion quantum bit, Phys. Rev. Lett. 113 (2014) http://dx.doi.org/10.1103/PhysRevLett.113.220501.

[135] N. Schlosser, G. Reymond, I. Protsenko, P. Grangier, Sub-poissonian loading of single atoms in a microscopic dipole trap, Nature 411 (6841) (2001) 1024–1027, http://dx.doi.org/10.1038/35082512.

[136] M.A. Norcia, A.W. Young, A.M. Kaufman, Microscopic control and detection of ultracold strontium in optical-tweezer arrays, Phys. Rev. X 8 (2018) 041054, http://dx.doi.org/10.1103/PhysRevX.8.041054.

[137] K. Barnes, P. Battaglino, B.J. Bloom, K. Cassella, R. Coxe, N. Crisosto, J.P. King, S.S. Kondov, K. Kotru, S.C. Larsen, J. Lauigan, B.J. Lester, M. McDonald, E. Megidish, S. Narayanaswami, C. Nishiguchi, R. Notermans, L.S. Peng, A. Ryou, T.-Y. Wu, M. Yarwood, Assembly and coherent control of a register of nuclear spin qubits, Nat. Commun. 13 (1) (2022) 2779, http://dx.doi.org/10.1038/s41467-022-29977-z.

[138] L. Isenhower, E. Urban, X.L. Zhang, A.T. Gill, T. Henage, T.A. Johnson, T.G. Walker, M. Saffman, Demonstration of a neutral atom controlled-NOT quantum gate, Phys. Rev. Lett. 104 (2010) http://dx.doi.org/10.1103/PhysRevLett.104.010503.

[139] M.W. Doherty, N.B. Manson, P. Delaney, F. Jelezko, J. Wrachtrup, L.C. Hollenberg, The nitrogen-vacancy colour centre in diamond, Phys. Rep. 528 (1) (2013) 1–45, http://dx.doi.org/10.1016/j.physrep.2013.02.001.

[140] Y. Wang, M. Um, J. Zhang, S. An, M. Lyu, J.-N. Zhang, L.-M. Duan, D. Yum, K. Kim, Single-qubit quantum memory exceeding ten-minute coherence time, Nat. Photonics 11 (10) (2017) 646–650, http://dx.doi.org/10.1038/s41566-017-0007-1.

[141] ITU-T, Transmission Media and Optical Systems Characteristics – Optical Fibre Cables, Standard, Telecomm. Standardization Sector of ITU, 2016.

[142] Y. Tamura, H. Sakuma, K. Morita, M. Suzuki, Y. Yamamoto, K. Shimada, Y. Honma, K. Sohma, T. Fujii, T. Hasegawa, The first 0.14-dB/km loss optical fiber and its impact on submarine transmission, J. Lightwave Technol. 36 (1) (2018) 44–49, http://dx.doi.org/10.1109/JLT.2018.2796647.

[143] E. Sugita, R. Nagase, K. Kanayama, T. Shintaku, SC-type single-mode optical fiber connectors, J. Lightwave Technol. 7 (11) (1989) 1689–1696, http://dx.doi.org/10.1109/50.45890.

[144] Y. Yamamoto, The quantum optical repeater, Science 263 (5152) (1994) 1394–1395, http://dx.doi.org/10.1126/science.263.5152.1394.

[145] M. Takeoka, S. Guha, M.M. Wilde, Fundamental rate-loss tradeoff for optical quantum key distribution, Nat. Commun. 5 (1) (2014) 5235, http://dx.doi.org/10.1038/ncomms6235.

[146] J. Miguel-Ramiro, W. Dür, Efficient entanglement purification protocols for d-level systems, Phys. Rev. A 98 (2018) http://dx.doi.org/10.1103/PhysRevA.98.042309.

[147] X.-M. Hu, C.-X. Huang, Y.-B. Sheng, L. Zhou, B.-H. Liu, Y. Guo, C. Zhang, W.-B. Xing, Y.-F. Huang, C.-F. Li, G.-C. Guo, Long-distance entanglement purification for quantum communication, Phys. Rev. Lett. 126 (2021) http://dx.doi.org/10.1103/PhysRevLett.126.010503.

[148] F.-F. Du, G. Fan, Y.-M. Wu, B.-C. Ren, Faithful and efficient hyperentanglement purification for spatial-polarization-time-bin photon system, Chin. Phys. B 32 (6) (2023) http://dx.doi.org/10.1088/1674-1056/aca395.

[149] C.-C. Luo, L. Zhou, W. Zhong, Y.-B. Sheng, Purification for hybrid logical qubit entanglement, Quantum Inf. Process. 21 (8) (2022) 300, http://dx.doi.org/10.1007/s11128-022-03646-y.

[150] T. van Leent, M. Bock, F. Fertig, R. Garthoff, S. Eppelt, Y. Zhou, P. Malik, M. Seubert, T. Bauer, W. Rosenfeld, W. Zhang, C. Becher, H. Weinfurter, Entangling single atoms over 33 km telecom fibre, Nature 607 (7917) (2022) 69–73, http://dx.doi.org/10.1038/s41586-022-04764-4.

[151] S. Ourari, Ł. Dusanowski, S.P. Horvath, M.T. Uysal, C.M. Phenicie, P. Stevenson, M. Raha, S. Chen, R.J. Cava, N.P. de Leon, J.D. Thompson, Indistinguishable telecom band photons from a single Er ion in the solid state, Nature 620 (7976) (2023) 977–981, http://dx.doi.org/10.1038/s41586-023-06281-4.

[152] J. Laurat, On-demand entanglement could lead to scalable quantum networks, Nature 558 (7709) (2018) 192–193, http://dx.doi.org/10.1038/d41586-018-05336-1.

[153] Q. Zhu, Y. Zhao, H. Xu, L. Huang, C. Qiao, Integrating all-optical switching and entangled photon source placement for entanglement routing, in: IEEE/ACM 31st Int. Symp. on Quality of Service, IWQoS, 2023, pp. 1–10, http://dx.doi.org/10.1109/IWQoS57198.2023.10188776.

[154] A.S. Cacciapuoti, M. Viscardi, J. Illiano, M. Caleffi, Entanglement distribution in the quantum internet: Knowing when to stop!, 2023, http://dx.doi.org/10.48550/arXiv.2307.05123, arxiv eprint.

[155] Y. Zeng, J. Zhang, J. Liu, Z. Liu, Y. Yang, Entanglement routing design over quantum networks, IEEE/ACM Trans. Netw. 32 (1) (2023) 352–367, http://dx.doi.org/10.1109/TNET.2023.3282560.

[156] L. Gyongyosi, S. Imre, Advances in the quantum internet, Commun. ACM 65 (8) (2022) 52–63, http://dx.doi.org/10.1145/3524455.

[157] H.J. Kimble, The quantum internet, Nature 453 (7198) (2008) 1023–1030, http://dx.doi.org/10.1038/nature07127.

[158] S. Wehner, D. Elkouss, R. Hanson, Quantum internet: A vision for the road ahead, Science 362 (6412) (2018) eaam9288, http://dx.doi.org/10.1126/science.aam9288.

[159] A.S. Cacciapuoti, M. Caleffi, F. Tafuri, F.S. Cataliotti, S. Gherardini, G. Bianchi, Quantum internet: Networking challenges in distributed quantum computing, IEEE Netw. 34 (1) (2020) 137–143, http://dx.doi.org/10.1109/MNET.001.1900092.

[160] M. Pant, H. Krovi, D. Towsley, L. Tassiulas, L. Jiang, P. Basu, D. Englund, S. Guha, Routing entanglement in the quantum internet, Npj Quantum Inf. 5 (1) (2019) 25, http://dx.doi.org/10.1038/s41534-019-0139-x.

[161] A. Reiserer, G. Rempe, Cavity-based quantum networks with single atoms and optical photons, Rev. Modern Phys. 87 (2015) 1379–1418, http://dx.doi.org/10.1103/RevModPhys.87.1379.

[162] Z. Li, K. Xue, J. Li, L. Chen, R. Li, Z. Wang, N. Yu, D.S. Wei, Q. Sun, J. Lu, Entanglement-assisted quantum networks: Mechanics, enabling technologies, challenges, and research directions, IEEE Commun. Surv. Tutor. 25 (4) (2023) 2133–2189, http://dx.doi.org/10.1109/COMST.2023.3294240.

[163] J. Miguel-Ramiro, A. Pirker, W. Dür, Genuine quantum networks with superposed tasks and addressing, Npj Quantum Inf. 7 (2021) http://dx.doi.org/10.1038/s41534-021-00472-5.

[164] J. Illiano, M. Caleffi, A. Manzalini, A.S. Cacciapuoti, Quantum internet protocol stack: A comprehensive survey, Comput. Netw. 213 (2022) 109092, http://dx.doi.org/10.1016/j.comnet.2022.109092.

[165] W. Kozlowski, S. Wehner, R.V. Meter, B. Rijsman, A.S. Cacciapuoti, M. Caleffi, S. Nagayama, RFC 9340: Architectural principles for a quantum internet, 2023, http://dx.doi.org/10.17487/RFC9340.

[166] M. Azari, P. Polakos, K.P. Seshadreesan, Quantum switches for Gottesman–Kitaev–Preskill qubit-based all-photonic quantum networks, 2024, http://dx.doi.org/10.48550/arXiv.2402.02721, arxiv preprint.

[167] I. Tillman, T. Vasantam, K.P. Seshadreesan, A continuous variable quantum switch, in: Proc. of Int. Conf. on Quantum Computing and Engineering, QCE, IEEE, 2022, pp. 365–371, http://dx.doi.org/10.1109/QCE53715.2022.00057.

[168] T.C. Ralph, A.P. Lund, Nondeterministic Noiseless Linear Amplification of Quantum Systems, Vol. 1110, 2009, pp. 155–160, http://dx.doi.org/10.1063/1.3131295.

[169] A. Pirker, W. Dür, A quantum network stack and protocols for reliable entanglement-based networks, New J. Phys. 21 (3) (2019) http://dx.doi.org/10.1088/1367-2630/ab05f7.

[170] R.V. Meter, J. Touch, Designing quantum repeater networks, IEEE Commun. Mag. 51 (8) (2013) 64–71, http://dx.doi.org/10.1109/MCOM.2013.6576340.

[171] R. Van Meter, R. Satoh, N. Benchasattabuse, K. Teramoto, T. Matsuo, M. Hajdušek, T. Satoh, S. Nagayama, S. Suzuki, A quantum internet architecture, in: 2022 IEEE Int. Conf. on Quantum Computing and Engineering, QCE, 2022, pp. 341–352, http://dx.doi.org/10.1109/QCE53715.2022.00055.

[172] Z. Li, K. Xue, J. Li, N. Yu, J. Liu, D.S.L. Wei, Q. Sun, J. Lu, Building a large-scale and wide-area quantum internet based on an OSI-alike model, China Commun. 18 (10) (2021) 1–14, http://dx.doi.org/10.23919/JCC.2021.10.001.

[173] A. Dahlberg, M. Skrzypczyk, T. Coopmans, L. Wubben, F. Rozpundefineddek, M. Pompili, A. Stolk, P. Pawełczak, R. Knegjens, J. de Oliveira Filho, R. Hanson, S. Wehner, A link layer protocol for quantum networks, in: Proceedings of the ACM Special Interest Group on Data Communication, ACM, New York, NY, USA, 2019, pp. 159–173, http://dx.doi.org/10.1145/3341302.3342070.

[174] A.S. Cacciapuoti, J. Illiano, M. Caleffi, Quantum internet addressing, IEEE Netw. 38 (1) (2024) 104–111, http://dx.doi.org/10.1109/MNET.2023.3328393.

[175] J. Illiano, M. Caleffi, M. Viscardi, A.S. Cacciapuoti, Quantum MAC: Genuine entanglement access control via many-body dicke states, IEEE Trans. Commun. 72 (4) (2024) 2090–2105, http://dx.doi.org/10.1109/TCOMM.2023.3344140.

[176] M. Caleffi, Optimal routing for quantum networks, IEEE Access 5 (2017) 22299–22312, http://dx.doi.org/10.1109/ACCESS.2017.2763325.

[177] N. Yu, C.-Y. Lai, L. Zhou, Protocols for packet quantum network intercommunication, IEEE Trans. Quantum Eng. 2 (2021) 1–9, http://dx.doi.org/10.1109/TQE.2021.3112594.

[178] Defense Advanced Research Projects Agency, Transmission control protocol, 1981, URL https://www.ietf.org/rfc/rfc793.txt. (Accessed 21 January 2025).

[179] R. Cleve, D. Gottesman, H.-K. Lo, How to share a quantum secret, Phys. Rev. Lett. 83 (1999) 648–651, http://dx.doi.org/10.1103/PhysRevLett.83.648.

[180] C.D. Donne, M. Iuliano, B. van der Vecht, G.M. Ferreira, H. Jirovská, T. van der Steenhoven, A. Dahlberg, M. Skrzypczyk, D. Fioretto, M. Teller, P. Filippov, A.R.-P. Montblanch, J. Fischer, B. van Ommen, N. Demetriou, D. Leichtle, L. Music, H. Ollivier, I. te Raa, W. Kozlowski, T. Taminiau, P. Pawełczak, T. Northup, R. Hanson, S. Wehner, Design and demonstration of an operating system for executing applications on quantum network nodes, 2024, http://dx.doi.org/10.48550/arXiv.2407.18306.

[181] T. Satoh, R.V. Meter, Quantum sockets for the NISQ era quantum internet, 2018, URL https://www.ngc.is.ritsumei.ac.jp/~ger/static/AQIS18/OnlineBooklet/158.pdf. (Accessed 21 January 2025).

[182] S. Gauthier, G. Vardoyan, S. Wehner, A control architecture for entanglement generation switches in quantum networks, in: Proc. of the 1st Workshop on Quantum Networks and Distributed Quantum Computing, QuNet, ACM, 2023, pp. 38–44, http://dx.doi.org/10.1145/3610251.3610552.

[183] Z. Xiao, J. Li, K. Xue, Z. Li, N. Yu, Q. Sun, J. Lu, A connectionless entanglement distribution protocol design in quantum networks, IEEE Netw. 38 (1) (2024) 131–139, http://dx.doi.org/10.1109/MNET.2023.3321044.

[184] C. Amza, A.L. Cox, S. Dwarkadas, P. Keleher, H. Lu, R. Rajamony, W. Yu, W. Zwaenepoel, TreadMarks: Shared memory computing on networks of workstations, Computer 29 (1996) 18–28, http://dx.doi.org/10.1109/2.485843.

[185] M. Van Steen, A.S. Tanenbaum, Distributed Systems, Maarten van Steen Leiden, The Netherlands, 2017, URL https://www.distributed-systems.net/index.php/books/ds3/.

[186] C. Berge, Hypergraphs: Combinatorics of Finite Sets, in: North-Holland Mathematical Library, Elsevier Science, 1984, URL https://shop.elsevier.com/books/hypergraphs/berge/978-0-444-87489-4.

[187] M. Zomorodi-Moghadam, M. Houshmand, M. Houshmand, Optimizing teleportation cost in distributed quantum circuits, Int. J. Theor. Phys. 57 (3) (2018) 848–861, http://dx.doi.org/10.1007/S10773-017-3618-X.

[188] B.W. Kernighan, S. Lin, An efficient heuristic procedure for partitioning graphs, Bell Syst. Tech. J. 49 (2) (1970) 291–307, http://dx.doi.org/10.1002/j.1538-7305.1970.tb01770.x.

[189] P. Andrés-Martínez, C. Heunen, Automated distribution of quantum circuits via hypergraph partitioning, Phys. Rev. A 100 (3) (2019) 032308, http://dx.doi.org/10.1103/PhysRevA.100.032308.

[190] Y. Akhremtsev, T. Heuer, P. Sanders, S. Schlag, Engineering a direct k-way hypergraph partitioning algorithm, in: Proc. of ALENEX, SIAM, 2017, pp. 28–42, http://dx.doi.org/10.1137/1.9781611974768.3.

[191] S. Schlag, T. Heuer, L. Gottesbüren, Y. Akhremtsev, C. Schulz, P. Sanders, High-quality hypergraph partitioning, ACM J. Exp. Algorithmics 27 (2023) 1–39, http://dx.doi.org/10.1145/3529090.

[192] M. Houshmand, Z. Mohammadi, M. Zomorodi-Moghadam, M. Houshmand, An evolutionary approach to optimizing teleportation cost in distributed quantum computation, Int. J. Theor. Phys. 59 (4) (2020) 1315–1329, http://dx.doi.org/10.1007/S10773-020-04409-0.

[193] O. Daei, K. Navi, M. Zomorodi-Moghadam, Optimized quantum circuit partitioning, Internat. J. Theoret. Phys. 59 (12) (2020) 3804–3820, http://dx.doi.org/10.1007/s10773-020-04633-8.

[194] E. Nikahd, N. Mohammadzadeh, M. Sedighi, M.S. Zamani, Automated window-based partitioning of quantum circuits, Phys. Scr. 96 (3) (2021) 035102, http://dx.doi.org/10.1088/1402-4896/ABD57C.

[195] J.-Y. Wu, K. Matsui, T. Forrer, A. Soeda, P. Andrés-Martínez, D. Mills, L. Henaut, M. Murao, Entanglement-efficient bipartite-distributed quantum computing, Quantum 7 (2023) 1196, http://dx.doi.org/10.22331/q-2023-12-05-1196.

[196] P. Andrés-Martínez, T. Forrer, D. Mills, J.-Y. Wu, L. Henaut, K. Yamamoto, M. Murao, R. Duncan, Distributing circuits over heterogeneous, modular quantum computing network architectures, 2023, http://dx.doi.org/10.48550/arXiv.2305.14148, arxiv preprint.

[197] R. G. Sundaram, H. Gupta, C.R. Ramakrishnan, Efficient distribution of quantum circuits, in: 35th Int. Symp. on Distributed Computing, DISC, in: Leibniz International Proceedings in Informatics (LIPIcs), vol. 209, 2021, pp. 1–20, http://dx.doi.org/10.4230/LIPIcs.DISC.2021.41.

[198] R.G. Sundaram, H. Gupta, C.R. Ramakrishnan, Distribution of quantum circuits over general quantum networks, in: 2022 IEEE Int. Conf. on Quantum Computing and Engineering, QCE, 2022, pp. 415–425, http://dx.doi.org/10.1109/QCE53715.2022.00063.

[199] R.G. Sundaram, H. Gupta, Distributing quantum circuits using teleportations, in: 2023 IEEE Int. Conf. on Quantum Software, QSW, 2023, pp. 186–192, http://dx.doi.org/10.1109/QSW59989.2023.00030.

[200] Z. Davarzani, M. Zomorodi-Moghadam, M. Houshmand, M. Nouri-Baygi, A dynamic programming approach for distributing quantum circuits by bipartite graphs, Quantum Inf. Process. 19 (2020) http://dx.doi.org/10.1007/s11128-020-02871-7.

[201] J. Clark, T. Humble, H. Thapliyal, TDAG: Tree-based directed acyclic graph partitioning for quantum circuits, in: Proc. of the Great Lakes Symposium on VLSI, ACM, 2023, pp. 587–592, http://dx.doi.org/10.1145/3583781.3590234.

[202] W. Cambiucci, R. Silveira, W. Ruggiero, Hypergraphic partitioning of quantum circuits for distributed quantum computing, in: 2023 IEEE Int. Conf. on Quantum Computing and Engineering, QCE, IEEE Computer Society, Los Alamitos, CA, USA, 2023, pp. 268–269, http://dx.doi.org/10.1109/QCE57702.2023.10237.

[203] C.M. Fiduccia, R.M. Mattheyses, A linear-time heuristic for improving network partitions, in: Papers on Twenty-Five Years of Electronic Design Automation, 1982, pp. 175–181, http://dx.doi.org/10.1109/DAC.1982.1585498.

[204] J.M. Baker, C. Duckering, A. Hoover, F.T. Chong, Time-sliced quantum circuit partitioning for modular architectures, in: Proc. of the 17th ACM Int. Conf. on Computing Frontiers, ACM, 2020, pp. 98–107, http://dx.doi.org/10.1145/3387902.3392617.

[205] T. Park, C.Y. Lee, Algorithms for partitioning a graph, Comput. Ind. Eng. 28 (4) (1995) 899–909, http://dx.doi.org/10.1016/0360-8352(95)00003-J.

[206] A. Ovide, S. Rodrigo, M. Bandic, H. Van Someren, S. Feld, S. Abadal, E. Alarcon, C.G. Almudever, Mapping quantum algorithms to multi-core quantum computing architectures, in: IEEE Int. Symp. on Circuits and Systems, ISCAS, IEEE, 2023, pp. 1–5, http://dx.doi.org/10.1109/ISCAS46773.2023.10181589.

[207] P. Escofet, A. Ovide, C.G. Almudever, E. Alarcón, S. Abadal, Hungarian qubit assignment for optimized mapping of quantum circuits on multi-core architectures, IEEE Comput. Archit. Lett. (2023) http://dx.doi.org/10.1109/LCA.2023.3318857.

[208] M. Bandic, L. Prielinger, J. Nüßlein, A. Ovide, S. Rodrigo, S. Abadal, H. van Someren, G. Vardoyan, E. Alarcon, C.G. Almudever, et al., Mapping quantum circuits to modular architectures with QUBO, in: 2023 IEEE Int. Conf. on Quantum Computing and Engineering, QCE, Vol. 1, IEEE, 2023, pp. 790–801, http://dx.doi.org/10.1109/QCE57702.2023.00094.

[209] A. Pastor, P. Escofet, S.B. Rached, E. Alarcón, P. Barlet-Ros, S. Abadal, Circuit partitioning for multi-core quantum architectures with deep reinforcement learning, 2024, http://dx.doi.org/10.48550/arXiv.2401.17976, arxiv preprint.

[210] H. Pashayan, J.J. Wallman, S.D. Bartlett, Estimating outcome probabilities of quantum circuits using quasiprobabilities, Phys. Rev. Lett. 115 (7) (2015) http://dx.doi.org/10.1103/PhysRevLett.115.070501.

[211] K. Mitarai, K. Fujii, Overhead for simulating a non-local channel with local channels by quasiprobability sampling, Quantum 5 (2021) 388, http://dx.doi.org/10.22331/q-2021-01-28-388.

[212] C. Piveteau, D. Sutter, S. Woerner, Quasiprobability decompositions with reduced sampling overhead, Npj Quantum Inf. 8 (1) (2022) 1–9, http://dx.doi.org/10.1038/s41534-022-00517-3.

[213] H.F. Hofmann, How to simulate a universal quantum computer using negative probabilities, J. Phys. A 42 (27) (2009) 275304, http://dx.doi.org/10.1088/1751-8113/42/27/275304.

[214] S. Bravyi, G. Smith, J.A. Smolin, Trading classical and quantum computational resources, Phys. Rev. X 6 (2) (2016) http://dx.doi.org/10.1103/PhysRevX.6.021043.

[215] K. Temme, S. Bravyi, J.M. Gambetta, Error mitigation for short-depth quantum circuits, Phys. Rev. Lett. 119 (18) (2017) http://dx.doi.org/10.1103/PhysRevLett.119.180509.

[216] S. Endo, S.C. Benjamin, Y. Li, Practical quantum error mitigation for near-future applications, Phys. Rev. X 8 (3) (2018) http://dx.doi.org/10.1103/PhysRevX.8.031027.

[217] C. Ying, B. Cheng, Y. Zhao, H.-L. Huang, Y.-N. Zhang, M. Gong, Y. Wu, S. Wang, F. Liang, J. Lin, Y. Xu, H. Deng, H. Rong, C.-Z. Peng, M.-H. Yung, X. Zhu, J.-W. Pan, Experimental simulation of larger quantum circuits with fewer superconducting qubits, Phys. Rev. Lett. 130 (11) (2023) 110601, http://dx.doi.org/10.1103/PhysRevLett.130.110601.

[218] A.P. Singh, K. Mitarai, Y. Suzuki, K. Heya, Y. Tabuchi, K. Fujii, Y. Nakamura, Experimental demonstration of a high-fidelity virtual two-qubit gate, Phys. Rev. Res. 6 (2024) http://dx.doi.org/10.1103/PhysRevResearch.6.013235.

[219] J.A. Smolin, J.M. Gambetta, G. Smith, Efficient method for computing the maximum-likelihood quantum state from measurements with additive Gaussian noise, Phys. Rev. Lett. 108 (7) (2012) 070502, http://dx.doi.org/10.1103/PhysRevLett.108.070502.

[220] M.A. Perlin, Z.H. Saleem, M. Suchara, J.C. Osborn, Quantum circuit cutting with maximum-likelihood tomography, Npj Quantum Inf. 7 (64) (2021) 1–8, http://dx.doi.org/10.1038/s41534-021-00390-6.

[221] A. Eddins, M. Motta, T.P. Gujarati, S. Bravyi, A. Mezzacapo, C. Hadfield, S. Sheldon, Doubling the size of quantum simulators by entanglement forging, PRX Quantum 3 (1) (2022) 010309, http://dx.doi.org/10.1103/PRXQuantum.3.010309.

[222] D. Faílde, J. Santos-Suárez, D.A. Herrera-Martí, J. Mas, Hamiltonian forging of a thermofield double, 2023, http://dx.doi.org/10.48550/arXiv.2311.10566, arxiv preprint.

[223] K. Mitarai, K. Fujii, Constructing a virtual two-qubit gate by sampling single-qubit operations, New J. Phys. 23 (2) (2021) 023021, http://dx.doi.org/10.1088/1367-2630/abd7bc.

[224] T. Peng, A.W. Harrow, M. Ozols, X. Wu, Simulating large quantum circuits on a small quantum computer, Phys. Rev. Lett. 125 (15) (2020) http://dx.doi.org/10.1103/PhysRevLett.125.150504.

[225] L. Brenner, C. Piveteau, D. Sutter, Optimal wire cutting with classical communication, 2023, http://dx.doi.org/10.48550/arXiv.2302.03366, arxiv preprint.

[226] A. Peruzzo, J. McClean, P. Shadbolt, M.-H. Yung, X.-Q. Zhou, P.J. Love, A. Aspuru-Guzik, J.L. O'Brien, A variational eigenvalue solver on a photonic quantum processor, Nat. Commun. 5 (1) (2014) 4213, http://dx.doi.org/10.1038/ncomms5213.

[227] S.C. Marshall, J. Tura, V. Dunjko, All this for one qubit? Bounds on local circuit cutting schemes, 2023, http://dx.doi.org/10.48550/arXiv.2303.13422, arxiv preprint, arXiv:2303.13422.

[228] C. Piveteau, D. Sutter, Circuit knitting with classical communication, IEEE Trans. Inform. Theory (2023) http://dx.doi.org/10.1109/TIT.2023.3310797, 1–1.

[229] A. Carrera Vazquez, C. Tornow, D. Ristè, S. Woerner, M. Takita, D.J. Egger, Combining quantum processors with real-time classical communication, Nature (2024) 1–5, http://dx.doi.org/10.1038/s41586-024-08178-2.

[230] A. Lowe, M. Medvidović, A. Hayes, L.J. O'Riordan, T.R. Bromley, J.M. Arrazola, N. Killoran, Fast quantum circuit cutting with randomized measurements, Quantum 7 (2023) 934, http://dx.doi.org/10.22331/q-2023-03-02-934.

[231] H.-Y. Huang, R. Kueng, J. Preskill, Predicting many properties of a quantum system from very few measurements, Nat. Phys. 16 (2020) 1050–1057, http://dx.doi.org/10.1038/s41567-020-0932-7.

[232] H. Harada, K. Wada, N. Yamamoto, Doubly optimal parallel wire cutting without ancilla qubits, PRX Quantum 5 (4) (2024) 040308, http://dx.doi.org/10.1103/PRXQuantum.5.040308.

[233] E. Pednault, An alternative approach to optimal wire cutting without ancilla qubits, 2023, http://dx.doi.org/10.48550/arXiv.2303.08287, arxiv preprint.

[234] C. Ufrecht, L.S. Herzog, D.D. Scherer, M. Periyasamy, S. Rietsch, A. Plinge, C. Mutschler, Optimal joint cutting of two-qubit rotation gates, Phys. Rev. A 109 (5) (2024) 052440, http://dx.doi.org/10.1103/PhysRevA.109.052440.

[235] A.W. Harrow, A. Lowe, Optimal quantum circuit cuts with application to clustered Hamiltonian simulation, 2024, http://dx.doi.org/10.48550/arXiv.2403.01018, arXiv.

[236] L. Schmitt, C. Piveteau, D. Sutter, Cutting circuits with multiple two-qubit unitaries, 2023, http://dx.doi.org/10.48550/arXiv.2312.11638, arxiv preprint.

[237] C. Ufrecht, M. Periyasamy, S. Rietsch, D.D. Scherer, A. Plinge, C. Mutschler, Cutting multi-control quantum gates with ZX calculus, Quantum 7 (2023) 1147, http://dx.doi.org/10.22331/q-2023-10-23-1147.

[238] J. Chen, E.M. Stoudenmire, S.R. White, Quantum Fourier transform has small entanglement, PRX Quantum 4 (4) (2023) http://dx.doi.org/10.1103/PRXQuantum.4.040318.

[239] Y. Zhang, L. Cincio, C.F.A. Negre, P. Czarnik, P.J. Coles, P.M. Anisimov, S.M. Mniszewski, S. Tretiak, P.A. Dub, Variational quantum eigensolver with reduced circuit complexity, Npj Quantum Inf. 8 (96) (2022) 1–10, http://dx.doi.org/10.1038/s41534-022-00599-z.

[240] S. Zhang, Z. Qin, Y. Zhou, R. Li, C. Du, Z. Xiao, Single entanglement connection architecture between multi-layer HEA for distributed VQE, 2023, http://dx.doi.org/10.48550/arXiv.2307.12323, arxiv preprint.

[241] G. Gentinetta, F. Metz, G. Carleo, Overhead-constrained circuit knitting for variational quantum dynamics, Quantum 8 (2024) 1296, http://dx.doi.org/10.22331/q-2024-03-21-1296.

[242] R. Nagai, S. Kanno, Y. Sato, N. Yamamoto, Quantum channel decomposition with preselection and postselection, Phys. Rev. A 108 (2) (2023) http://dx.doi.org/10.1103/PhysRevA.108.022615.

[243] D. Chen, B. Baheri, V. Chaudhary, Q. Guan, N. Xie, S. Xu, Approximate quantum circuit cutting, 2022, http://dx.doi.org/10.48550/arXiv.2212.01270, arxiv preprint.

[244] D.T. Chen, E.H. Hansen, X. Li, V. Kulkarni, V. Chaudhary, B. Ren, Q. Guan, S. Kuppannagari, J. Liu, S. Xu, Efficient quantum circuit cutting by neglecting basis elements, in: IEEE IPDPS Workshops, IPDPSW, 2023, pp. 517–523, http://dx.doi.org/10.1109/IPDPSW59300.2023.00091.

[245] S. Brandhofer, I. Polian, K. Krsulich, Optimal partitioning of quantum circuits using gate cuts and wire cuts, IEEE Trans. Quantum Eng. 5 (2024) 1–10, http://dx.doi.org/10.1109/TQE.2023.3347106.

[246] W. Tang, T. Tomesh, M. Suchara, J. Larson, M. Martonosi, CutQC: Using small quantum computers for large quantum circuit evaluations, in: Proc. of the 26th ACM Int. Conf. on Architectural Support for Programming Languages and Operating Systems, ACM, 2021, pp. 473–486, http://dx.doi.org/10.1145/3445814.3446758.

[247] W. Tang, M. Martonosi, ScaleQC: A scalable framework for hybrid computation on quantum and classical processors, 2022, http://dx.doi.org/10.48550/arXiv.2207.00933, arxiv preprint.

[248] K.N. Smith, M.A. Perlin, P. Gokhale, P. Frederick, D. Owusu-Antwi, R. Rines, V. Omole, F. Chong, Clifford-based circuit cutting for quantum simulation, in: Proc. of the 50th Int. Symp. on Computer Architecture, ACM, 2023, pp. 1–13, http://dx.doi.org/10.1145/3579371.3589352.

[249] T. Chatterjee, A. Das, S.I. Mohtashim, A. Saha, A. Chakrabarti, Qurzon: A prototype for a divide and conquer-based quantum compiler for distributed quantum systems, SN Comput. Sci. 3 (4) (2022) http://dx.doi.org/10.1007/s42979-022-01207-9.

[250] L.S. Herzog, F. Wagner, C. Ufrecht, L. Palackal, A. Plinge, C. Mutschler, D.D. Scherer, Improving Quantum and Classical Decomposition Methods for Vehicle Routing, 2024, http://dx.doi.org/10.48550/arXiv.2404.05551, arXiv.

[251] A. Robert, P.K. Barkoutsos, S. Woerner, I. Tavernelli, Resource-efficient quantum algorithm for protein folding, Npj Quantum Inf. 7 (1) (2021) 38, http://dx.doi.org/10.1038/s41534-021-00368-4.

[252] J. Tilly, H. Chen, S. Cao, D. Picozzi, K. Setia, Y. Li, E. Grant, L. Wossnig, I. Rungger, G.H. Booth, J. Tennyson, The variational quantum eigensolver: A review of methods and best practices, Phys. Rep. 986 (2022) 1–128, http://dx.doi.org/10.1016/j.physrep.2022.08.003, The Variational Quantum Eigensolver: A Review of Methods and Best Practices.

[253] Y. Wang, L.M. Sager-Smith, D.A. Mazziotti, Quantum simulation of bosons with the contracted quantum eigensolver, New J. Phys. 25 (10) (2023) http://dx.doi.org/10.1088/1367-2630/acf9c3.

[254] P. Gokhale, O. Angiuli, Y. Ding, K. Gui, T. Tomesh, M. Suchara, M. Martonosi, F.T. Chong, Minimizing state preparations in variational quantum eigensolver by partitioning into commuting families, 2019, http://dx.doi.org/10.48550/arXiv.1907.13623, arxiv preprint.

[255] M. Schuld, V. Bergholm, C. Gogolin, J. Izaac, N. Killoran, Evaluating analytic gradients on quantum hardware, Phys. Rev. A 99 (2019) 032331, http://dx.doi.org/10.1103/PhysRevA.99.032331.

[256] J. Stokes, J. Izaac, N. Killoran, G. Carleo, Quantum natural gradient, Quantum 4 (2020) 269, http://dx.doi.org/10.22331/q-2020-05-25-269.

[257] J.D. Viqueira, D. Faílde, M.M. Juane, A. Gómez, D. Mera, Density matrix emulation of quantum recurrent neural networks for multivariate time series prediction, 2023, http://dx.doi.org/10.48550/arXiv.2310.20671, arXiv.

[258] D. Wierichs, J. Izaac, C. Wang, C.Y.-Y. Lin, General parameter-shift rules for quantum gradients, Quantum 6 (2022) 677, http://dx.doi.org/10.22331/q-2022-03-30-677.

[259] D. Faílde, J.D. Viqueira, M. Mussa Juane, A. Gómez, Using differential evolution to avoid local minima in variational quantum algorithms, Sci. Rep. 13 (1) (2023) 16230, http://dx.doi.org/10.1038/s41598-023-43404-3.

[260] M.S. Alvarez-Alvarado, F.E. Alban-Chacón, E.A. Lamilla-Rubio, C.D. Rodríguez-Gallegos, W. Velásquez, Three novel quantum-inspired swarm optimization algorithms using different bounded potential fields, Sci. Rep. 11 (1) (2021) 11655, http://dx.doi.org/10.1038/s41598-021-90847-7.

[261] E.R. Anschuetz, B.T. Kiani, Quantum variational algorithms are swamped with traps, Nat. Commun. 13 (1) (2022) 7760, http://dx.doi.org/10.1038/s41467-022-35364-5.

[262] Y. Du, Y. Qian, X. Wu, D. Tao, A distributed learning scheme for variational quantum algorithms, IEEE Trans. Quantum Eng. 3 (2022) http://dx.doi.org/10.1109/TQE.2022.3175267.

[263] S.Y.C. Chen, S. Yoo, Federated quantum machine learning, Entropy 23 (4) (2021) http://dx.doi.org/10.3390/e23040460.

[264] S. Stein, N. Wiebe, Y. Ding, P. Bo, K. Kowalski, N. Baker, J. Ang, A. Li, EQC: Ensembled quantum computing for variational quantum algorithms, in: Proc. of the 49th Int. Symp. on Computer Architecture, ACM, 2022, pp. 59–71, http://dx.doi.org/10.1145/3470496.3527434.

[265] P. Das, S.S. Tannu, P.J. Nair, M. Qureshi, A case for multi-programming quantum computers, in: Proceedings of the 52nd Annual IEEE/ACM Int. Symp. on Microarchitecture, 2019, pp. 291–303, http://dx.doi.org/10.1145/3352460.3358287.

[266] A. Ash-Saki, M. Alam, S. Ghosh, Analysis of crosstalk in NISQ devices and security implications in multi-programming regime, in: Proc. of the ACM/IEEE Int. Symp. on Low Power Electronics and Design, 2020, pp. 25–30, http://dx.doi.org/10.1145/3370748.3406570.

[267] L. Liu, X. Dou, QuCloud: A new qubit mapping mechanism for multi-programming quantum computing in cloud environment, in: IEEE Int. Symp. on High-Performance Computer Architecture, HPCA, IEEE, 2021, pp. 167–178, http://dx.doi.org/10.1109/HPCA51647.2021.00024.

[268] A.A. Saki, S. Ghosh, Qubit sensing: A new attack model for multi-programming quantum computing, 2021, http://dx.doi.org/10.48550/arXiv.2104.05899, arxiv preprint.

[269] S. Niu, A. Todri-Sanial, Multi-programming cross platform benchmarking for quantum computing hardware, 2022, http://dx.doi.org/10.48550/arXiv.2206.03144, arxiv preprint.

[270] S. Niu, A. Todri-Sanial, How parallel circuit execution can be useful for NISQ computing? in: 2022 Design, Automation & Test in Europe Conference & Exhibition, DATE, 2022, pp. 1065–1070, http://dx.doi.org/10.23919/DATE54114.2022.9774512.

[271] S. Niu, A. Todri-Sanial, Enabling multi-programming mechanism for quantum computing in the NISQ era, Quantum 7 (2023) 925, http://dx.doi.org/10.22331/q-2023-02-16-925.

[272] S. Niu, A. Todri-Sanial, Multi-programming mechanism on near-term quantum computing, in: Quantum Computing: Circuits, Systems, Automation and Applications, Springer, 2023, pp. 19–54, http://dx.doi.org/10.1007/978-3-031-37966-6_2.

[273] L. Liu, X. Dou, QuCloud+: A holistic qubit mapping scheme for single/multi-programming on 2D/3D NISQ quantum computers, ACM Trans. Archit. Code Optim. 21 (1) (2024) 1–27, http://dx.doi.org/10.1145/3631525.

[274] J.K. Lee, O.T. Brown, M. Bull, M. Ruefenacht, J. Doerfert, M. Klemm, M. Schulz, Quantum task offloading with the openmp API, 2023, http://dx.doi.org/10.48550/arXiv.2311.03210, arxiv preprint.

[275] The CUDA Quantum development team, CUDA Quantum, 2025, URL https://github.com/NVIDIA/cuda-quantum. (Accessed 21 January 2025).

[276] A.J. Abhari, A. Faruque, M.J. Dousti, L. Svec, O. Catu, A. Chakrabati, C.-F. Chiang, S. Vanderwilt, J. Black, F. Chong, M. Martonosi, M. Suchara, K. Brown, M. Pedram, T. Brun, Scaffold: Quantum Programming Language, Tech. Rep., Princeton University, Dept. of Computer Science, 2012, URL https://www.cs.princeton.edu/techreports/2012/934.pdf. (Accessed 21 January 2025).

[277] K. Svore, A. Geller, M. Troyer, J. Azariah, C. Granade, B. Heim, V. Kliuchnikov, M. Mykhailova, A. Paz, M. Roetteler, Q#: Enabling scalable quantum computing and development with a high-level DSL, in: Proc. of the Real World Domain Specific Languages Workshop, ACM, 2018, pp. 1–10, http://dx.doi.org/10.1145/3183895.3183901.

[278] J. Guo, H. Lou, J. Yu, R. Li, W. Fang, J. Liu, P. Long, S. Ying, M. Ying, isQ: An integrated software stack for quantum programming, IEEE Trans. Quantum Eng. 4 (2023) http://dx.doi.org/10.1109/TQE.2023.3275868.

[279] S. Liu, X. Wang, L. Zhou, J. Guan, Y. Li, Y. He, R. Duan, M. Ying, Q|SI⟩: A quantum programming environment, in: Symposium on Real-Time and Hybrid Systems: Essays Dedicated to Professor Chaochen Zhou on the Occasion of His 80th Birthday, Springer International Publishing, 2018, pp. 133–164, http://dx.doi.org/10.1007/978-3-030-01461-2_8, Chapter 8.

[280] Qiskit contributors, Qiskit: An open-source framework for quantum computing, 2023, http://dx.doi.org/10.5281/zenodo.2573505.

[281] Cirq Developers, Cirq, 2023, http://dx.doi.org/10.5281/zenodo.4062499.

[282] Y. Suzuki, Y. Kawase, Y. Masumura, Y. Hiraga, M. Nakadai, J. Chen, K.M. Nakanishi, K. Mitarai, R. Imai, S. Tamiya, T. Yamamoto, T. Yan, T. Kawakubo, Y.O. Nakagawa, Y. Ibe, Y. Zhang, H. Yamashita, H. Yoshimura, A. Hayashi, K. Fujii, Qulacs: A fast and versatile quantum circuit simulator for research purpose, Quantum 5 (2021) http://dx.doi.org/10.22331/Q-2021-10-06-559.

[283] T. Häner, D.S. Steiger, T. Hoefler, M. Troyer, Distributed quantum computing with QMPI, in: Int. Conf. for High Perf. Computing, Networking, Storage and Analysis, IEEE, 2021, http://dx.doi.org/10.1145/3458817.3476172.

[284] Y. Shi, T. Nguyen, S. Stein, T. Stavenger, M. Warner, M. Roetteler, T. Hoefler, A. Li, A reference implementation for a quantum message passing interface, in: Proc. of the SC'23 Workshops, ACM, 2023, pp. 1420–1425, http://dx.doi.org/10.1145/3624062.3624212.

[285] R. Wakizaka, Towards reliable distributed quantum computing on quantum interconnects, ACM Int. Conf. Proc. Ser. (2023) 114–116, http://dx.doi.org/10.1145/3594671.3594691.

[286] F. Chow, Intermediate representation, Queue 11 (2013) 30–37, http://dx.doi.org/10.1145/2542661.2544374.

[287] A. McCaskey, T. Nguyen, A MLIR dialect for quantum assembly languages, in: Int. Conf, on Quantum Computing and Engineering, QCE, IEEE, 2021, pp. 255–264, http://dx.doi.org/10.1109/QCE52317.2021.00043.

[288] T. Lubinski, C. Granade, A. Anderson, A. Geller, M. Roetteler, A. Petrenko, B. Heim, Advancing hybrid quantum–classical computation with real-time execution, Front. Phys. 10 (2022) 940293, http://dx.doi.org/10.3389/FPHY.2022.940293.

[289] A. Peduri, S. Bhat, T. Grosser, QSSA: An SSA-based IR for quantum computing, in: Proc. of the 31st ACM SIGPLAN Int. Conf. on Compiler Construction, ACM, New York, NY, USA, 2022, pp. 2–14, http://dx.doi.org/10.1145/3497776.3517772.

[290] D. Ittah, T. Häner, V. Kliuchnikov, T. Hoefler, QIRO: A static single assignment-based quantum program representation for optimization, ACM Trans. Quantum Comput. 3 (3) (2022) http://dx.doi.org/10.1145/3491247.

[291] K. Hietala, R. Rand, S.-H. Hung, X. Wu, M. Hicks, Verified optimization in a quantum intermediate representation, 2019, http://dx.doi.org/10.48550/arXiv.1904.06319, arxiv preprint.

[292] X.-Z. Luo, J.-G. Liu, P. Zhang, L. Wang, Yao.jl: Extensible, efficient framework for quantum algorithm design, Quantum 4 (2020) 341, http://dx.doi.org/10.22331/q-2020-10-11-341.

[293] S. Nishio, R. Wakizaka, InQuIR: Intermediate representation for interconnected quantum computers, 2023, http://dx.doi.org/10.48550/arXiv.2302.00267, arxiv preprint.

[294] A.W. Cross, L.S. Bishop, J.A. Smolin, J.M. Gambetta, Open quantum assembly language, 2017, http://dx.doi.org/10.48550/arXiv.1707.03429, arxiv preprint.

[295] N. Khammassi, G.G. Guerreschi, I. Ashraf, J.W. Hogaboam, C.G. Almudever, K. Bertels, cQASM v1.0: Towards a common quantum assembly language, 2018, http://dx.doi.org/10.48550/arXiv.1805.09607, arxiv preprint.

[296] X. Fu, L. Riesebos, M.A. Rol, J. Van Straten, J. Van Someren, N. Khammassi, I. Ashraf, R.F. Vermeulen, V. Newsum, K.K. Loh, J.C. De Sterke, W.J. Vlothuizen, R.N. Schouten, C.G. Almudever, L. Dicarlo, K. Bertels, EQASM: An executable quantum instruction set architecture, in: Proc. 25th IEEE Int. Symp. on High Performance Computer Architecture, HPCA, IEEE, 2019, pp. 224–237, http://dx.doi.org/10.1109/HPCA.2019.00040.

[297] K.M. Svore, A.V. Aho, A.W. Cross, I. Chuang, I.L. Markov, A layered software architecture for quantum computing design tools, Computer 39 (1) (2006) 74–83, http://dx.doi.org/10.1109/MC.2006.4.

[298] A. Dahlberg, B. van der Vecht, C.D. Donne, M. Skrzypczyk, I.t. Raa, W. Kozlowski, S. Wehner, NetQASM - a low-level instruction set architecture for hybrid quantum-classical programs in a quantum internet, Quantum Sci. Technol. 7 (3) (2022) 035023, http://dx.doi.org/10.1088/2058-9565/ac753f.

[299] M. Ying, Y. Feng, An algebraic language for distributed quantum computing, IEEE Trans. Comput. 58 (6) (2009) 728–743, http://dx.doi.org/10.1109/TC.2009.13.

[300] X. Qiu, L. Chen, Quantum cost of dense coding and teleportation, Phys. Rev. A 105 (6) (2022) 062451, http://dx.doi.org/10.1103/PhysRevA.105.062451.

[301] C. Piveteau, D. Sutter, Circuit knitting with classical communication, IEEE Trans. Inform. Theory 70 (4) (2024) 2734–2745, http://dx.doi.org/10.1109/TIT.2023.3310797.

[302] D.W. Wall, Global register allocation at link time, ACM SIGPLAN Not. 21 (1986) 264–275, http://dx.doi.org/10.1145/13310.13338.

[303] T. Ito, N. Kakimura, N. Kamiyama, Y. Kobayashi, Y. Okamoto, Algorithmic theory of qubit routing, in: Algorithms and Data Structures, Springer Nature, 2023, pp. 533–546, http://dx.doi.org/10.1007/978-3-031-38906-1_35.

[304] A. Paler, On the influence of initial qubit placement during NISQ circuit compilation, in: Quantum Technology and Optimization Problems, Springer, 2019, pp. 207–217, http://dx.doi.org/10.1007/978-3-030-14082-3_18.

[305] M. Bandic, C.G. Almudever, S. Feld, Interaction graph-based characterization of quantum benchmarks for improving quantum circuit mapping techniques, Quantum Mach. Intell. 5 (2023) 1–30, http://dx.doi.org/10.1007/S42484-023-00124-1.

[306] A. Cowtan, S. Dilkes, R. Duncan, A. Krajenbrink, W. Simmons, S. Sivarajah, On the qubit routing problem, in: 14th Conf. on the Theory of Quantum Computation, Communication and Cryptography, TQC, Vol. 135, 2019, pp. 5:1–5:32, http://dx.doi.org/10.4230/LIPIcs.TQC.2019.5.

[307] H. Barnes, A Survey of Qubit Routing Algorithms (Ph.D. thesis), Rochester Institute of Technology, 2023, URL https://repository.rit.edu/theses/11428.

[308] Y. Mao, Y. Liu, Y. Yang, Qubit allocation for distributed quantum computing, in: Conf. on Computer Communications, IEEE, 2023, pp. 1–10, http://dx.doi.org/10.1109/INFOCOM53939.2023.10228915.

[309] Y. Mao, Y. Liu, Y. Yang, Probability-aware qubit-to-processor mapping in distributed quantum computing, in: 1st Workshop on Quantum Networks and Distributed Quantum Computing, ACM, 2023, pp. 51–56, http://dx.doi.org/10.1145/3610251.3610554.

[310] M. Nakai, Qubit Allocation For Distributed Quantum Computing (Bachelor's thesis), Keio University, 2021, URL https://aqua.sfc.wide.ad.jp/publications/dave_bthesis.pdf.

[311] Z. Chen, X. Chen, Y. Jiang, X. Cheng, Z. Guan, Routing strategy for distributed quantum circuit based on optimized gate transmission direction, Internat. J. Theoret. Phys. 62 (12) (2023) 255, http://dx.doi.org/10.1007/s10773-023-05489-4.

[312] S. Rodrigo, D. Spanò, M. Bandic, S. Abadal, H. Van Someren, A. Ovide, S. Feld, C.G. Almudéver, E. Alarcón, Characterizing the spatio-temporal qubit traffic of a quantum intranet aiming at modular quantum computer architectures, in: 9th ACM Int. Conf. on Nanoscale Computing and Communication, 2022, pp. 1–7, http://dx.doi.org/10.1145/3558583.3558846.

[313] S.B. Rached, I.L. Agudo, S. Rodrigo, M. Bandic, S. Feld, H. van Someren, E. Alarcón, C.G. Almudéver, S. Abadal, Characterizing the inter-core qubit traffic in large-scale quantum modular architectures, 2023, http://dx.doi.org/10.48550/arXiv.2310.01921, arxiv preprint.

[314] N. Khammassi, I. Ashraf, J. Someren, R. Nane, A. Krol, M.A. Rol, L. Lao, K. Bertels, C.G. Almudever, OpenQL: A portable quantum programming framework for quantum accelerators, ACM J. Emerg. Technol. Comput. Syst. 18 (1) (2021) 1–24, http://dx.doi.org/10.1145/3474222.

[315] L. Lao, H. Van Someren, I. Ashraf, C.G. Almudever, Timing and resource-aware mapping of quantum circuits to superconducting processors, IEEE Trans. Comput.- Aided Des. Integr. Circuits Syst. 41 (2) (2021) 359–371, http://dx.doi.org/10.1109/TCAD.2021.3057583.

[316] A. Fagan, R. Duncan, Optimising clifford circuits with quantomatic, Electron. Proc. Theor. Comput. Sci. 287 (2019) 85–105, http://dx.doi.org/10.4204/eptcs.287.5.

[317] M. Xu, Z. Li, O. Padon, S. Lin, J. Pointing, A. Hirth, H. Ma, J. Palsberg, A. Aiken, U.A. Acar, Z. Jia, Quartz: superoptimization of quantum circuits, in: Proc. of the Conf. on Programming Language Design and Implementation, PLDI, ACM, 2022, pp. 625–640, http://dx.doi.org/10.1145/3519939.3523433.

[318] S. Haider, S.A.R. Kazmi, An extended quantum process algebra (eQPAlg) approach for distributed quantum systems, 2020, http://dx.doi.org/10.48550/arXiv.2001.04249, arxiv preprint.

[319] M. Lalire, P. Jorrand, A process algebraic approach to concurrent and distributed quantum computation: Operational semantics, 2004, http://dx.doi.org/10.48550/arXiv.quant-ph/0407005, arXiv.

[320] Y. Feng, S. Li, M. Ying, Verification of distributed quantum programs, ACM Trans. Comput. Log. 23 (3) (2022) http://dx.doi.org/10.1145/3517145.

[321] C.A.R. Hoare, An axiomatic basis for computer programming, Commun. ACM 12 (1969) 576–580, http://dx.doi.org/10.1145/363235.363259.

[322] Y. Wang, Verification of distributed quantum protocols, 2021, http://dx.doi.org/10.48550/arXiv.2110.11416, arxiv preprint.

[323] C.H. Bennett, G. Brassard, Quantum cryptography: Public key distribution and coin tossing, Theoret. Comput. Sci. 560 (2014) 7–11, http://dx.doi.org/10.1016/j.tcs.2014.05.025.

[324] D. Ferrari, S. Nasturzio, M. Amoretti, Poster: A software tool for mapping and executing distributed quantum computations on a network simulator, in: QCrypt, 2021, URL https://2021.qcrypt.net/posters/QCrypt2021Poster185Ferrari.pdf. (Accessed 21 January 2025).

[325] D. Ferrari, S. Carretta, M. Amoretti, A modular quantum compilation framework for distributed quantum computing, IEEE Trans. Quantum Eng. 4 (2023) 1–13, http://dx.doi.org/10.1109/TQE.2023.3303935.

[326] D. Cuomo, Architectures and Circuits for Distributed Quantum Computing (Ph.D. thesis), Univ. of Naples Federico II, Univ. of Camerino, National Research Council of Italy, 2023, URL http://www.fedoatd.unina.it/961/1/THESIS_DQC.pdf.

[327] Y. Ohkura, T. Satoh, R. Van Meter, Simultaneous execution of quantum circuits on current and near-future NISQ systems, IEEE Trans. Quantum Eng. 3 (2022) 1–10, http://dx.doi.org/10.1109/TQE.2022.3164716.

[328] T. Tomesh, Z.H. Saleem, M.A. Perlin, P. Gokhale, M. Suchara, M. Martonosi, Divide and conquer for combinatorial optimization and distributed quantum computation, in: Int. Conf. on Quantum Computing and Engineering, QCE, Vol. 1, IEEE, 2023, pp. 1–12, http://dx.doi.org/10.1109/QCE57702.2023.00009.

[329] S. Sivarajah, S. Dilkes, A. Cowtan, W. Simmons, A. Edgington, R. Duncan, $\mathrm{T}|\mathrm{ket}\rangle$: A retargetable compiler for NISQ devices, Quantum Sci. Technol. 6 (1) (2020) 014003, http://dx.doi.org/10.1088/2058-9565/ab8e92.

[330] J.C. Boschero, N.M.P. Neumann, W. van der Schoot, T. Sijpesteijn, R. Wezeman, Distributed quantum computing: Applications and challenges, 2024, http://dx.doi.org/10.48550/arXiv.2410.00609.

[331] C. Gidney, M. Ekerå, How to factor 2048 bit RSA integers in 8 hours using 20 million noisy qubits, Quantum 5 (2021) http://dx.doi.org/10.22331/q-2021-04-15-433.

[332] M. Ekerå, J. Håstad, Quantum algorithms for computing short discrete logarithms and factoring RSA integers, in: T. Lange, T. Takagi (Eds.), Post-Quantum Cryptography, in: LNCS, no. 10346, Springer, 2017, pp. 347–363, http://dx.doi.org/10.1007/978-3-319-59879-6_20.

[333] L. Xiao, D. Qiu, L. Luo, P. Mateus, Distributed quantum-classical hybrid shor's algorithm, 2023, http://dx.doi.org/10.48550/arXiv.2304.12100, arxiv preprint.

[334] N.M. Neumann, R. van Houte, T. Attema, Imperfect distributed quantum phase estimation, in: ICCS, in: LNCS, vol. 12142, Springer, 2020, pp. 605–615, http://dx.doi.org/10.1007/978-3-030-50433-5_46.

[335] R. Van Meter, W.J. Munro, K. Nemoto, K.M. Itoh, Arithmetic on a distributed-memory quantum multicomputer, ACM J. Emerg. Technol. Comput. Syst. 3 (4) (2008) http://dx.doi.org/10.1145/1324177.1324179.

[336] J. Tan, L. Xiao, D. Qiu, L. Luo, P. Mateus, Distributed quantum algorithm for Simon's problem, Phys. Rev. A 106 (3) (2022) http://dx.doi.org/10.1103/PhysRevA.106.032417.

[337] H. Li, D. Qiu, L. Luo, Distributed exact quantum algorithms for Deutsch–Jozsa problem, 2023, http://dx.doi.org/10.48550/arXiv.2303.10663, arxiv preprint.

[338] X. Zhou, D. Qiu, L. Luo, Distributed Bernstein–Vazirani algorithm, Phys. A 629 (2023) 129209, http://dx.doi.org/10.1016/j.physa.2023.129209.

[339] X. Zhou, D. Qiu, L. Luo, Distributed exact Grover's algorithm, Front. Phys. 18 (5) (2023) 1–25, http://dx.doi.org/10.1007/s11467-023-1327-x.

[340] J. Avron, O. Casper, I. Rozen, Quantum advantage and noise reduction in distributed quantum computing, Phys. Rev. A 104 (2021) http://dx.doi.org/10.1103/PhysRevA.104.052404.

[341] K. Fujii, K. Mizuta, H. Ueda, K. Mitarai, W. Mizukami, Y.O. Nakagawa, Deep variational quantum eigensolver: A divide-and-conquer method for solving a larger problem with smaller size quantum computers, PRX Quantum 3 (2022) http://dx.doi.org/10.1103/PRXQuantum.3.010346.

[342] Z. Zhou, Y. Du, X. Tian, D. Tao, QAOA-in-QAOA: Solving large-scale MaxCut problems on small quantum machines, Phys. Rev. Appl. 19 (2) (2023) 024027, http://dx.doi.org/10.1103/PhysRevApplied.19.024027.

[343] S.C. Marshall, C. Gyurik, V. Dunjko, High dimensional quantum machine learning with small quantum computers, Quantum 7 (2022) http://dx.doi.org/10.22331/q-2023-08-09-1078.

[344] K. Zhang, P. Rao, K. Yu, H. Lim, V. Korepin, Implementation of efficient quantum search algorithms on NISQ computers, Quantum Inf. Process. 20 (2021) 233, http://dx.doi.org/10.1007/s11128-021-03165-2.

[345] G. Park, K. Zhang, K. Yu, V. Korepin, Quantum multi-programming for Grover's search, Quantum Inf. Process. 22 (1) (2023) http://dx.doi.org/10.1007/s11128-022-03793-2.

[346] L.K. Grover, Trade-offs in the quantum search algorithm, Phys. Rev. A 66 (2002) http://dx.doi.org/10.1103/PhysRevA.66.052314.

[347] K. Li, D. Qiu, L. Li, S. Zheng, Z. Rong, Application of distributed semi-quantum computing model in phase estimation, Inform. Process. Lett. 120 (2017) 23–29, http://dx.doi.org/10.1016/j.ipl.2016.12.002.

[348] T. Tanaka, Y. Suzuki, S. Uno, R. Raymond, T. Onodera, N. Yamamoto, Amplitude estimation via maximum likelihood on noisy quantum computer, Quantum Inf. Process. 20 (2021) 1–29, http://dx.doi.org/10.1007/s11128-021-03215-9.

[349] S. DiAdamo, M. Ghibaudi, J. Cruise, Distributed quantum computing and network control for accelerated VQE, IEEE Trans. Quantum Eng. 2 (2021) 1–21, http://dx.doi.org/10.1109/TQE.2021.3057908.