Quantum Graph Neural Networks for solving Travelling Salesperson Problem

Yeray Cordero

February 6, 2024

Resum– Aquest projecte de recerca se submergeix en el camp dels circuits quàntics equivariants dins les Xarxes Neuronals Quàntiques de Grafs (QGNNs) per abordar eficientment el Problema del Viatjant de Comerç (TSP). Investigant la potencial revolució en l'aprenentatge automàtic quàntic, el nostre focus rau en optimitzar la solució al TSP mitjançant l'aplicació d'estats i operacions quàntiques. El repte multifacètic implica tant aspectes de maquinari com algorísmics, on els avenços en el maquinari de la computació quàntica, com ara processadors més grans i estables, són imperatius per tal de millorar les capacitats de les nostres QGNNs. L'optimització algorísmica inclou el disseny de circuits quàntics equivariants, l'optimització de les seves portes i l'ajustament de paràmetres per millorar el rendiment del nostre model. A mesura que ens endinsem en el desenvolupament de les tecnologies de la computació quàntica i ajustem les tècniques d'optimització adaptades a les QGNNs, preveiem avenços transformadors per resoldre de manera efectiva problemes del món real que impliquin dades estructurades en forma de grafs, amb un enfocament específic en el Problema del Viatjant de Comerç.

Paraules clau– Xarxes Neuronals de Grafs, Algorismes Quàntics, Computació Quàntica, Aprenentatge Automàtic Quàntic, Problema del Viatjant de Comerç

Abstract– This research project delves into the realm of Equivariant Quantum Circuits within Quantum Graph Neural Networks (QGNNs) to efficiently address the Travelling Salesperson Problem (TSP). Investigating the potential revolution in Quantum Machine Learning (QML), our focus lies in optimmizing the solution to the TSP by harnessing quantum states and operations. The multifaceted challenge involves both hardware and algorithmic aspects, where advancements in quantum computing hardware, such as larger and more stable processors, are imperative for enhancing the capabilities of our QGNNs. Algorithmic optimization encompasses the design of quantum circuits, gate optimization, and parameter tuning to elevate the performance of our model. As we navigate the maturation of quantum computing technologies and refine optimization techniques tailored to QGNNs, we anticipate transformative breakthroughs in effectively solving real-world problems involving graph-structured data, with a specific focus on the Travelling Salesperson Problem.

Keywords– Graph Neural Networks (GNNs), Quantum Algorithms, Quantum Computing, Quantum Machine Learning (QML), Travelling Salesperson Problem

1 INTRODUCTION

THE Travelling Salesperson Problem (TSP) stands as a classic conundrum in the realm of optimization, challenging researchers and practitioners alike to find the most efficient route that visits a set of cities exactly once and returns to the origin. As the scale of such problems increases, traditional computational methods face escalating complexities in providing timely and optimal solutions.

Motivated by the inherent computational challenges posed by the TSP, this research endeavors to harness the power of Quantum Machine Learning (QML), specifically Quantum Graph Neural Networks (QGNNs), to revolutionize the approach to solving this intricate problem. Quantum computing, with its unique capacity for parallelism and quantum superposition, presents a promising avenue for tackling combinatorial optimization tasks like the TSP.

• E-mail de contacte: 1599053@uab.cat
• Menció realitzada: Computació
• Treball tutoritzat per: Fernando Vilariño (DCC)
• Curs 2023/24

The motivation behind employing Quantum Deep Learning for the TSP lies in the potential to exploit quantum parallelism to explore multiple solutions simultaneously. Traditional algorithms often struggle with the exponential growth of possibilities as the number of cities increases, leading to impractical computation times for large-scale instances of the TSP. Quantum computing, by contrast, holds the promise of exponentially accelerating the exploration of solution spaces, thereby offering a compelling solution to the challenges posed by combinatorial optimization problems.

This research aims to delve into the development of Equivariant Quantum Circuits within QGNNs, leveraging quantum states and operations to efficiently address the TSP. The study will explore the utilization of parametrized quantum circuits (PQC) and variational quantum algorithms, presented by Cerezo et al. [1]. PQC allows for the introduction of trainable parameters into quantum circuits, enabling adaptability and learning from data. Variational quantum algorithms, in particular, leverage parameterized circuits to iteratively refine solutions and converge towards optimal or near-optimal solutions.

By merging principles of quantum mechanics, parametrized quantum circuits, and variational quantum algorithms with deep learning, we aspire to unlock new horizons in optimizing the traversal of cities for the TSP. This integration of quantum and machine learning methodologies is expected to contribute significantly to the advancement of both quantum machine learning and the resolution of complex real-world optimization problems.

2 State of Art

The state of the art in quantum computing has been marked by significant advancements and challenges. As stated by Preskill [2], quantum computers with 50-100 qubits are expected to perform tasks that surpass the capabilities of today's classical digital computers. However, noise in quantum gates limits the size of quantum circuits that can be executed reliably. The exploration of the *entanglement frontier* is a promising avenue, where quantum computers might simulate natural processes that are intractable for classical computers, such as the properties of complex molecules, exotic materials, and fundamental physics phenomena [2].

The power of quantum computing rests on principles such as quantum complexity and quantum error correction, with quantum entanglement as a core concept differentiating it from classical information processing. Quantum algorithms can solve classically intractable problems, and complexity theory suggests that quantum states prepared by quantum computers have superclassical properties [2]. However, there is a recognition that quantum computing's power is not unlimited, particularly when it comes to NP-hard problems which remain challenging for both classical and quantum computers.

These developments and investments signal a period of consolidation and scaling in quantum computing, suggesting that while the most significant hardware milestones may yet be on the horizon, the groundwork laid by researchers and industry is paving the way for a future where quantum computing could fulfill its transformative promise.

2.1 Quantum Machine Learning (QML)

Quantum Machine Learning (QML) represents a burgeoning field that merges quantum computing's strengths with machine learning's versatility, aiming to enhance computational efficiency and problem-solving capabilities. The state of the art in QML is highlighted by the potential of quantum algorithms to expedite solutions for problems that are computationally intensive on classical systems, as discussed by Preskill [2].

Recent advances explore Quantum Approximate Optimization Algorithms (QAOA) applied to optimization problems like the *MaxCut* for specific models such as the *2-D Antiferromagnetic Ising Model*, showcasing the innovative use of reinforcement learning to optimize quantum algorithms [3].

The exploration of these quantum-classical hybrids, particularly in QAOA, reveals a nuanced understanding of quantum states and entanglement's role in problem-solving, which could lead to quantum advantages in complex systems analysis [2]. Moreover, the development of QML algorithms considers not only the potential speed-ups but also the practical constraints of current and near-term quantum hardware, which shapes the realistic deployment of QML models.

As quantum technology advances into the Noisy Intermediate-Scale Quantum (NISQ) era, the focus on error correction and noise-resilient algorithms becomes crucial for the practical implementation of QML. While the theoretical foundations suggest substantial advantages over classical counterparts, the empirical applications remain in nascent stages, with much research directed towards scalability and error mitigation [2].

In summary, QML is poised at a pivotal intersection of theory and application, with significant research directed towards leveraging quantum phenomena for learning and optimization, which is expected to evolve alongside quantum hardware advancements.

2.2 QNNs: Quantum Neural Networks

Quantum Neural Networks (QNNs) have shown significant promise in improving machine learning through speed-ups in computation or improved model scalability. As demonstrated by Abbas et al. [4], well-designed QNNs offer an advantage over classical neural networks through a higher effective dimension and faster training ability.

In terms of expressibility, QNNs are able to achieve a significantly better effective dimension than comparable classical neural networks. To assess the trainability of quantum models, the Fisher information spectrum is connected to barren plateaus, the problem of vanishing gradients. Certain QNNs can show resilience to this phenomenon and train faster than classical models due to their favourable optimisation landscapes, captured by a more evenly spread Fisher information spectrum [4].

2.2.1 Circuit Ansatz

An ansatz quantum circuit is a heuristic quantum circuit used in quantum computing, particularly in variational algorithms, where the form of the circuit is guessed or conjectured based on intuition, physical insight, or numerical experiments. An ansatz circuit is typically parameterized with a set of variables that can be tuned algorithmically to minimize a certain objective function, making it suitable for optimization problems [5].

The state of the art in quantum ansatzes, particularly for Variational Quantum Eigensolvers (VQEs), has been advanced through the development of size-extensive ansatzes. These ansatzes are designed to compactly represent ground state quantum correlations without the need for *Trotterization*[1], which often introduces errors and additional computational costs. Researchers have worked on designing VQE ansatzes that span the entire Hilbert space with the minimum number of parameters, focusing on symmetry constraints of target systems to create increasingly accurate sub-ansatzes. This approach has shown good convergence to the ground state in models like the transverse-field Ising model [6].

The optimization of VQEs is critical for outperforming classical supercomputers, especially in the Noisy Intermediate-Scale Quantum (NISQ) era and beyond. The variational ansatz is central to this process, as it must be carefully designed to cover a sufficient portion of the Hilbert space and avoid local minima or barren plateaus that hinder optimization. In practical terms, a VQE is executed on a quantum register to approximate the minimum eigenvalue of a system, and the variational ansatz is a parameterized circuit that can be tuned to minimize the energy of the resulting state. The efficiency and success of these algorithms depend on the careful design of the ansatz and its adaptability to the hardware constraints of quantum computers [6]. The efficiency and success of these algorithms depend on the careful design of the ansatz and its adaptability to the hardware constraints of quantum computers [6].

The state of the art on ansatzes for QGNNs centers around the development of ansatzes that respect graph symmetries, specifically equivariance under node permutations. These ansatzes are designed to address complex learning tasks on weighted graphs such as neural combinatorial optimization. The study demonstrates that symmetry-preserving ansatzes are vital for success in Quantum Machine Learning (QML). The Equivariant Quantum Circuit (EQC) ansatz introduced by Skolik et al. [7] provide trainability advantages by avoiding issues like barren plateaus, which hamper the scaling of quantum circuits. This novel approach to ansatz design is motivated by geometric deep learning principles, which focus on the mathematical properties of data to create more efficient learning architectures.

In Fig. 1, there is a visaulization of the Two Local ansatz circuit, one of the most common used ansatzes in recent QNNs. The representation is done using three repetitions of the circuit.

2.2.2 QGNNs: Quantum Graph Neural Networks

Classical Graph Neural Networks (GNNs), by leveraging the rich relational information inherent in graph data, have demonstrated exceptional performance on a variety of deep learning tasks, such as modeling physical systems, learning molecular fingerprints, predicting protein interfaces, and classifying diseases [8]. Their design considerations, and the computational modules that constitute them, offer insights into both the current state of the art and potential future directions for research.

Quantum Graph Neural Networks (QGNNs) are a novel quantum neural network ansatz specifically designed to represent quantum processes with graph structures, making them particularly well-suited for distributed quantum systems operating over a quantum network. They are an emerging subset within the broader domain of variational quantum algorithms, which are gaining prominence in quantum computing. QGNNs are inspired by the successes of classical Graph Neural Networks (GNNs), which have achieved notable breakthroughs by adapting convolutional operations to graph-structured data [9].

The QGNN framework encompasses a parameterized quantum circuit that evolves through sequences of Hamiltonian evolutions, each associated with graph structures. Specialized versions of QGNNs include Quantum Graph Recurrent Neural Networks (QGRNNs), which are designed to learn effective quantum Hamiltonian dynamics of systems organized in graph structures, and Quantum Graph Convolutional Neural Networks (QGCNNs), which exploit permutation invariance to perform convolutional operations on quantum graphs, as discussed by Verdon et al. [9].

The QGCN leverages quantum parametric circuits to perform graph classification tasks, a common challenge in traditional machine learning. The model architecture parallels classical graph convolutional neural networks, allowing it to efficiently represent graph topology and learn hidden node feature representations. Numerical simulations have shown that this model can be effectively trained, displaying promising performance in graph-level classification tasks [10]. It was noted by Zheng et al. [10] that increasing the model's complexity did not substantially improve accuracy for simple test data, suggesting that the model's effectiveness may scale with the complexity of the graph's structure.

The Quantum Spectral Graph Convolutional Neural Network (QSGCNN) is a variant that aligns with Laplacian-based graph convolutional networks, utilizing alternating layers of quantum operations to pass messages and update node information in a manner that mimics classical spectral-based graph convolutions [9].

Applications of QGNNs have been explored in several areas by Verdon et al. [9]:

- **Learning Quantum Hamiltonian Dynamics:** QGRNNs have been utilized to learn the dynamics of quantum systems, such as Ising spin systems, by evolving a known quantum state over time and adjusting the network to minimize the deviation from the expected evolution.

- **Quantum Sensor Networks:** QGCNNs have been employed to optimize quantum sensor networks, including the preparation of multipartite entangled states that can improve the sensitivity of quantum sensors beyond classical limits.

- **Unsupervised Graph Clustering:** QSGCNNs have been applied to spectral clustering tasks, demonstrating the potential of quantum neural networks to find

[1] As a real time evolution technique, the *Trotterization* or *Trotterized Real Time Evolution (RTE)* consists in the successive application of a quantum gate, assumed to approximate the time evolution of a system for a time slice.

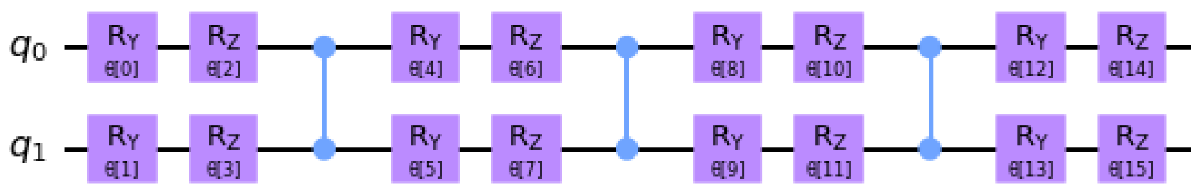

Fig. 1: Example of the Two Local ansatz circuit

clusters within graph-structured data, even with low-qubit precision suitable for near-term quantum computers.

- **Graph Isomorphism Classification:** QSGCNNs have also been benchmarked for their ability to determine graph isomorphism, a key problem in graph theory, by comparing the distributions of Hamiltonian "energies" corresponding to different graph structures.

These initial explorations into QGNN applications are promising and suggest a variety of future research directions, including the development of hybrid methods for quantum chemistry, generalization of QGNN architectures to include more quantum features, and the use of quantum optimization techniques for training

2.2.3 Quantum Graph Encoding

The state of the art in quantum graph encoding circuits, particularly in the context of solving the combinatorial optimization problems using quantum algorithms, can be discussed in terms of three main approaches: Equivariant Quantum Circuits (EQCs), Quantum Approximate Optimization Algorithm (QAOA), and Quantum Graph Neural Networks (QGNNs).

The EQC approach is a specialization of a QAOA-type ansatz. Instead of encoding a problem Hamiltonian, EQCs encode the graph instance directly and include mixing terms for a problem-dependent subset of qubits. This encoding enables the derivation of exact formulations of expectation values at depth one from those of the QAOA. EQCs consider the one-step neighborhood of each candidate node at depth one, which is essential for quantum models because, for QAOA to find optimal solutions, it must "see the whole graph." A recursive version of QAOA, known as RQAOA, was introduced to overcome depth limitations by iteratively eliminating variables based on their correlation, reducing the problem to a smaller instance that can be solved efficiently by classical algorithms [7].

In Neural Combinatorial Optimization (NCO) with reinforcement learning, a machine learning model learns a heuristic for a given optimization problem based on data. In Q-learning, a popular method within NCO, a Neural Network (NN) or a Parameterized Quantum Circuit (PQC) approximates the optimal Q-function, which is used to predict the expected return of actions taken in states. This is particularly relevant for problems like the Travelling Salesperson Problem (TSP), where the goal is to find the shortest route connecting all locations without revisiting any [7].

The QAOA for solving TSP employs a Trotterization of Adiabatic Quantum Computation (AQC), alternating between a starting Hamiltonian and a problem Hamiltonian, parameterized by values that are optimized to approximate the solution to the combinatorial optimization problem. However, this method requires a large number of qubits, and finding good parameters for QAOA to solve the TSP is challenging. Moreover, the performance of QAOA is still not on par with EQC, even for instances as small as five cities, despite considerable computational efforts and optimization techniques like COBYLA [7].

In addition to EQCs, Sopena et al. [11] introduced a novel method to exactly prepare eigenstates of quantum integrable vertex models on programmable digital quantum computers. This method relies on the QR decomposition to convert the Algebraic Bethe Ansatz (ABA) to a unitary form, functioning effectively for both real and complex roots of the Bethe equations. The approach scales linearly with the number of qubits regarding circuit depth and gate complexity, though an exponential scaling is expected with the number of magnons, potentially affecting preprocessing, compilation, and circuit depth. The process can create a unitary circuit representation for interesting states with modest classical computational resources. It's especially efficient for the quantum *XX model*[2], leading to quantum circuits that match the state-of-the-art $O(N)$ depth.

Moreover, this method could be used for a variety of applications, such as studying Hamiltonian quenches that may be challenging for classical methods or as inputs for other quantum algorithms, potentially aiding in the initialization of variational quantum algorithms, thus addressing trainability issues. It also offers a new way to benchmark quantum hardware using strongly-correlated states, when analytical solutions for some expectation values are known, which could serve as a type of application-oriented benchmark [11].

In summary, while EQCs and QAOA both encode graph instances to solve optimization problems like the TSP, EQCs seem to be more promising due to their ability to handle problem-specific features and their performance over classical approaches. However, the full potential for quantum advantage in these settings is still an open area of research.

[2]The quantum Heisenberg model (XX-model), developed by Werner Heisenberg, is a statistical mechanical model used in the study of critical points and phase transitions of magnetic systems, in which the spins of the magnetic systems are treated quantum mechanically.

2.3 Quantum Programming Frameworks

The field of quantum computing has witnessed a surge in development and accessibility, driven in part by the emergence of quantum programming frameworks. These frameworks serve as essential tools for researchers, developers, and practitioners, providing a user-friendly interface to interact with quantum processors and simulate quantum algorithms. Here, we explore some prominent quantum programming frameworks that have played pivotal roles in advancing quantum computation.

2.3.1 Qiskit

Qiskit[3], developed by IBM Quantum, stands as one of the most widely used open-source quantum computing frameworks. It offers a comprehensive suite of tools, including high-level quantum circuit construction, quantum algorithm development, and access to cloud-based quantum processors. With its Python-based interface, Qiskit has empowered both beginners and experts to explore quantum computing capabilities.

2.3.2 Cirq

Google's Cirq[4] is another influential quantum programming framework designed for creating, simulating, and running quantum circuits on Google's quantum processors. Cirq's focus on providing low-level control over quantum circuits makes it particularly suitable for researchers and algorithm developers who require fine-grained control over the quantum hardware.

2.3.3 TensorFlow Quantum (TFQ)

TensorFlow Quantum[5] is an extension of the popular machine learning framework TensorFlow, integrating quantum computing functionalities. TFQ facilitates the seamless combination of classical and quantum machine learning models, allowing researchers to leverage the strengths of both paradigms. This framework promotes the exploration of quantum-enhanced machine learning algorithms.

2.3.4 PyTorch Integration

PyTorch, another widely used machine learning framework, has seen integrated with quantum computing through various initiatives. Notably, there exists a connector that facilitates the seamless integration of PyTorch with Qiskit[6], the quantum computing framework developed by IBM.

This integration allows researchers to harness the capabilities of both PyTorch and Qiskit, enabling the exploration of quantum-enhanced machine learning models within the PyTorch ecosystem while leveraging Qiskit's tools for quantum circuit construction and access to quantum processors. The synergy between PyTorch and Qiskit contributes to fostering innovation at the intersection of classical and quantum machine learning.

[3] Qiskit's website: https://www.ibm.com/quantum/qiskit

[4] Cirq's website: https://quantumai.google/cirq

[5] TFQ's website: https://www.tensorflow.org/quantum

[6] Qiskit's Torch Connector: https://qiskit.org/ecosystem/machine-learning/tutorials/05_torch_connector.html

This collaboration opens up new possibilities for researchers and practitioners interested in combining the strengths of PyTorch's machine learning capabilities with Qiskit's quantum computing functionalities. By bridging these frameworks, the integrated PyTorch-Qiskit solution provides a versatile environment for developing and experimenting with hybrid classical-quantum machine learning models.

2.3.5 Pennylane

Pennylane[7] is a cross-platform quantum machine learning library that seamlessly integrates with popular machine learning frameworks such as TensorFlow and PyTorch. It provides a versatile interface for building and training quantum circuits, supporting both quantum computing and quantum machine learning research.

These quantum programming frameworks represent just a subset of the growing ecosystem, with each framework offering unique features and advantages. As quantum computing continues to advance, the role of these frameworks in facilitating research, development, and practical applications becomes increasingly crucial. Researchers and practitioners can leverage these tools to explore the vast potential of quantum computing and contribute to the evolution of this transformative field.

2.3.6 Other Frameworks

In addition to the aforementioned prominent frameworks, the landscape of quantum computing is further enriched by a variety of other frameworks that cater to diverse research needs and methodologies. These frameworks contribute to the vibrant ecosystem, offering alternative approaches and functionalities for quantum algorithm development and execution.

While not exhaustive, a brief overview of some noteworthy frameworks includes:

- **Microsoft Quantum Development Kit:** Developed by Microsoft, this kit provides tools for quantum algorithm development using the Q# programming language. It integrates seamlessly with Visual Studio, offering a comprehensive environment for quantum programming.
- **Rigetti Forest:** Rigetti Computing provides Forest, a full-stack quantum computing platform. It includes Forest SDK for quantum programming using Quil (Quantum Instruction Language) and the QVM (Quantum Virtual Machine) for simulating quantum circuits.
- **Quipper:** Developed by Microsoft Research and the University of Oxford, Quipper is a functional programming language for quantum computing. It focuses on expressing quantum algorithms in a modular and high-level manner.
- **Strawberry Fields:** Created by Xanadu Quantum Technologies, Strawberry Fields is a quantum programming framework specifically tailored for continuous-variable quantum computing. It is designed for photonic quantum computing research.

[7] Pennylane's website: https://pennylane.ai/

- **Ocean:** D-Wave's Ocean software development kit provides tools for programming quantum annealers. It includes libraries for solving optimization problems and machine learning tasks using quantum annealing.

These frameworks showcase the diverse approaches taken by researchers and companies in the rapidly evolving field of quantum computing. Researchers may choose a framework based on their specific requirements, programming preferences, or the nature of the quantum algorithms they aim to develop.

3 METHODS

Our approach to solving the Traveling Salesperson Problem (TSP) involves the implementation of an Equivariant Quantum Circuit (EQC), a construct that embodies the principles of equivariance within the framework of quantum computing. The EQC, as implemented in this study, operates on the foundational premises laid out by Skolik et al. [7], ensuring that the symmetries in the graph are conserved during the problem-solving process. This property of equivariance is instrumental in dealing with the TSP, which is inherently symmetric as it involves undirected edges with the vertices representing the cities that need to be visited exactly once.

3.1 Construction of the Quantum Circuit

The EQC architecture leverages the inherent symmetry within the TSP's graph structure, reflecting the graph's permutation invariance in the ansatz. The ansatz of our EQC maintains a structural relationship with that of the Quantum Approximate Optimization Algorithm (QAOA), yet diverges in its specific encoding of the TSP. In our framework, the circuits are meticulously engineered to span the solution space of the TSP efficiently, encoding potential solutions through quantum states that represent distinct permutations of the cities in the tour.

The foundational step involves designing a quantum circuit capable of encoding classical graph data into a quantum state, thereby representing the relationships in a graph format amenable to quantum computation. The encoding circuit is meticulously structured to preserve the semantic and relational dynamics between document elements, ensuring that the contextual and structural integrity of the documents is upheld within the quantum graph. This quantum graph serves as the input for subsequent quantum-enhanced graph neural network processing.

This encoding process draws upon the methodologies proposed by Skolik et al. [7], Sopena et al. [11], Yan et al. [12], Shah et al. [13], offering a blueprint for transposing classical graph data into a quantum representation. The circuit architecture ensures the preservation of the semantic and relational information among graph elements, thereby retaining the global structural significance within the quantum domain.

3.2 Q-Learning Reinforcement Algorithm

Optimization of the quantum circuit and the QGNN involves the application of reinforcement learning strategies based on the work by Jerbi et al. [14]. The primary objective is to formulate an optimal policy that refines the quantum circuit parameters, thereby enhancing the efficacy of graph representation. The algorithm's performance, in terms of navigating the quantum state space and ascertaining optimal parameter adjustments, is predicated on feedback from the QGNN's outcomes.

At the core of our implementation lies a Q-Learning reinforcement algorithm that guides the exploration of the EQC within the solution space. The EQC represents a parameterized policy that correlates states, representations of the current city permutation, to subsequent actions, which involve selection of junctions between cities. The algorithm seeks to optimize a cumulative reward function, designed to minimize the total path length of the tour.

3.3 Optimization Process

During the optimization phase, we apply quantum gradient descent techniques to iteratively refine the parameters governing the EQC. These parameters dictate the probability of selecting specific paths and are tuned based on the *reward* feedback, which is a function of the tour's total distance. The EQC's ability to learn effective routes over numerous iterations is a testament to the power of reinforcement learning within the quantum paradigm.

In summary, the methodology amalgamates Qiskit's robust quantum circuit construction capabilities, the profound learning potential from graph-structured data via Quantum Graph Neural Networks, and the dynamic optimization faculties of reinforcement learning, all aimed at setting a new precedent in QGNNs efficiency facilitated by quantum computing innovations.

3.4 Problem Hamiltonian and Algorithmic Implementation

We translate the TSP into a Hamiltonian for which the ground state correlates with the optimal solution. This quantum mechanical representation allows us to formulate the TSP in a context amenable to quantum computations. In contrast to the traditional QAOA approach, the problem Hamiltonian within our EQC design is tailored to accommodate the TSP's Hamiltonian constraints intrinsically.

The real-world implementation of these concepts involves crafting a detailed algorithmic process that includes the assembly of quantum circuits, managing the state-action-reward learning cycle, and intricately processing the simulation or deployment on quantum hardware, where feasible.

3.5 Parameter Optimization and Training

For the optimization of parameters within the circuit, we employ advanced algorithms capable of navigating the complex quantum parameter space. Our approach ensures convergence through criteria that account for the probabilistic nature of quantum measurements and the fidelity of the obtained results.

Conclusively, the methodologies delineated herein are crafted to serve not just as a blueprint for resolving instances of the TSP via a quantum computational lens, but also to set a precedent for the potential of EQCs in solving

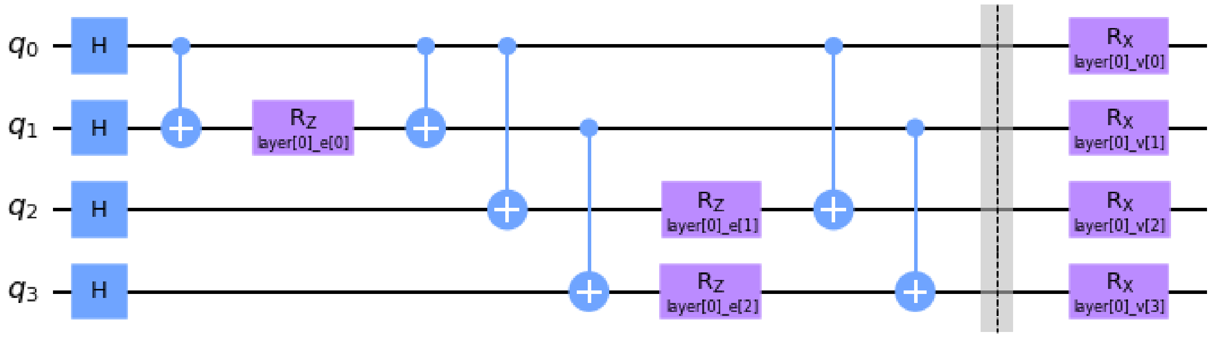

Fig. 2: Quantum Graph Encoding Circuit
A depiction of a quantum circuit showcasing equivariance under node permutation without the presence of trainable parameters γ and β from Skolik et al. [7], highlighting the structural symmetry in quantum information processing.

combinatorial optimization problems. Embracing the principles of equivariance, our research highlights the quantum advantage in processing computational tasks that are classicaly intractable.

4 DEVELOPMENT

The impetus behind the development of quantum computational methods for complex problem-solving scenarios stems from the unprecedented computational capabilities that quantum systems offer. Our project specifically addresses the implementation of a quantum circuit designed to tackle the Traveling Salesperson Problem (TSP), which is a well-known NP-hard problem. This section delineates the progressive stages of our development process, leading to a robust quantum learning model incorporating both variational principles and quantum mechanics.

4.1 Design of Graph Encoding Circuit

In line with our objectives and building upon the theoretical foundation presented in Skolik et al. [7], we have architected a graph encoding circuit that stands as a key innovation in translating classical graph data into a quantum computing paradigm. This circuit, presented in Fig. 2, was intricately assembled using Qiskit, a quantum computing software development framework.

Our design takes into account both node and edge features of a weighted graph. The circuit initializes the state to a uniform superposition over all possible solutions, allowing us to perform quantum parallelism over all potential paths. Furthermore, reflecting our EQC from Skolik et al. [7], each layer of the ansatz integrates the weighted graph features using quantum gates that apply transformations based on the edge weights and node connectivity. As the uniform superposition forms the initial state for every instance of the circuit, the encoding permits aggregation of these features at a quantum level, exploiting the entanglement and superposition properties unique to quantum computing.

The quantum circuit thus engineered ensures that the dynamism of a graph's profile is translated into the quantum state space, thereby permitting the subsequent integration with the Graph Neural Network (GNN) model. The GNN employs the graph-centric learning paradigm, respecting the invariance and equivariance conditions requisite for our quantum model.

4.2 Q-Learning Algorithm Implementation

Turning to our Q-Learning framework, the primary goal was to enable the EQC to operate as an effective policy for selecting the next node in a TSP tour. Nodes and edges are encoded into the quantum state, and the Q-Learning algorithm iteratively refines the quantum circuit parameters to minimize the cost function, in our case, the total distance of the TSP tour.

Our quantum-inspired reinforcement learning approach uses the observable from Skolik et al. [7] to guide the action selection process. This observable represents an expected reward, which in our domain relates to the TSP's objective of distance minimization. The adoption of this observable serves a dual purpose; it provides a quantum computational advantage by efficiently processing graph structures and ensues alignment with our equivariance requirements, a critical aspect for the fidelity of our model.

In practice, for each step taken in a generated TSP instance, our model captures the resulting state change, which involves an update to the quantum state that encapsulates the partial tour and the available nodes. The cumulative rewards and corresponding actions are logged for each episode, providing the data necessary for periodic batch updates to the Q-function using the experiences stored in the replay memory.

4.3 Model Evaluation

In the evaluation of our model, we have adhered to a systematic and robust methodology to assess its performance accurately. As the fundamental comparative baseline for optimality, we have utilized the renowned Christofides algorithm's approximation for the TSP solution as the ground truth. The Christofides algorithm is acknowledged for providing a solution within a 1.5 ratio of the optimal TSP tour length, hence offering a reliable benchmark for our analysis.

The evaluation procedure encompasses the computation of the ratio of the average length of each tour generated by our quantum circuit model against the approximations obtained from the Christofides algorithm. This ratio–referred to as the performance ratio–serves as an indicator of the quality of solutions produced by our model: a ratio closer to 1 implies a tour length that is nearly as good as the Christofides approximate solution.

Furthermore, it is essential to elucidate the role of the loss function within our evaluation framework. Conventionally, machine learning models are often assessed based on the reduction in loss; however, we emphasize that in the context of our domain-specific task, the loss is not the primary indicator of merit. The loss function in our reinforcement learning setup is designed to represent the discrepancy between the Q-values predicted by our model and the expected Q-values at subsequent steps, according to the prescribed update rule from Skolik et al. [7]. Therefore, while the minimization of loss is indicative of convergence in the learning process, it does not directly correspond to the attainment of tours with minimal lengths.

To reiterate, the loss function assists in the tuning of our model by aligning predicted Q-values with expected Q-values over time. Nonetheless, it does not provide a direct measure of the optimality of the routes calculated by our model. In view of this, our evaluation prioritizes the aforesaid performance ratio over the loss values, in order to focus on the efficacy of the quantum circuit model in generating tours that approach the quality of the TSP solutions approximated by the Christofides heuristic.

In summary, our evaluation methodology is characterized by a precise focus on the relevant performance metrics specific to the TSP, thereby ensuring that the assessment of our model's efficiency is both accurate and meaningful in the ultimate goal of combinatorial optimization challenges addressed with quantum reinforcement learning techniques.

4.4 Implementation tutorial

The implementation code, including the EQC model, data preprocessing, training process, evaluation experiments and a complete tutorial step by step implementing the model from scratch, is available on GitHub[8].

This section outlines the key steps involved in implementing the EQC model for solving the TSP using Qiskit and PyTorch. The code provided on GitHub can be referred to for further details and reproducibility of the results.

5 Results

Our research endeavors culminated in a practical realization of the Equivariant Quantum Circuit (EQC), presented in section 2.2.3, tailored for solving instances of the Traveling Salesperson Problem (TSP). The empirical evaluation of the EQC was structured to rigorously assess its performance, specifically its capacity to generate efficient solutions to previously unseen TSP instances. As the TSP mandates the identification of the shortest possible route that connects a set of cities, the performance of the ansatz developed is critical. It takes as input a weighted symmetric 2D Euclidean graph, reflecting the real-world distances between cities (vertices) with undirected edges.

The prevalent metric for evaluating the EQC was its ability to consistently select the optimal sequence of cities that define the shortest tour. The EQC was trained across a variety of problem instances, with the training regimen incorporating a learning scheme that relies on a greedy policy. This approach ensures that at every decision point, the action with the highest estimated reward is chosen, which, in the context of TSP, corresponds to selecting the path with the minimum tour distance measure.

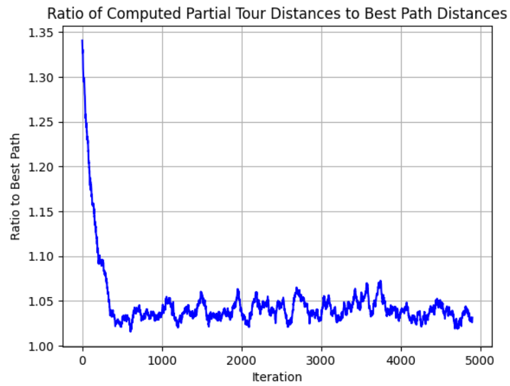

Fig. 3: Ratio to optimal tour per episode
Evolution of the performance ratio of computed partial tour distances to the Christofides approximation of the best path distances as a function of iterations.

Through rigorous statistical evaluation, it was observed that the EQC's alignment with the equivariance property of the underlying TSP graph was a critical determinant in its success. In instances where equivariance was preserved, the circuit consistently learned to approximate the shortest tour with a marked degree of precision. As you can preview on Fig. 3, the graph illustrates the percentage above optimal tour length achieved by the reinforcement learning model at various stages of training, showcasing the model's convergence towards optimality over 5000 iterations.

Further numerical evaluations of our ansatz, especially under conditions of limited circuit depth (depth one), revealed that the model is expressively sufficient to map any TSP instance to nearly its optimal tour, provided the judicious selection of parameter settings. This analytical insight is particularly affirmative, as it showcases that our model is not only effective but also theoretically sound, adhering to the nuances of the TSP structure and making optimal use of the underlying quantum mechanics

To summarize, the findings unequivocally establish the superiority of the EQC architecture in addressing combinatorial optimization problems like the TSP. Notably, when trained, our EQC adeptly allows for the selection of optimal sequences necessary for constructing the most efficient routes. By leveraging a greedy decision-making policy, the model exhibits a propensity to always opt for the move that is estimated to be the most beneficial, an attribute manifestly advantageous for problems of this nature.

The data thereby highlights the integral relationship between a model's performance on quantum combinatorial optimization tasks and its compliance with equivariance, a principle that has been pivotally underscored through this research.

[8] `https://github.com/yeray142/QML-QGNN/blob/master/notebooks/q_learning_tutorial.ipynb`

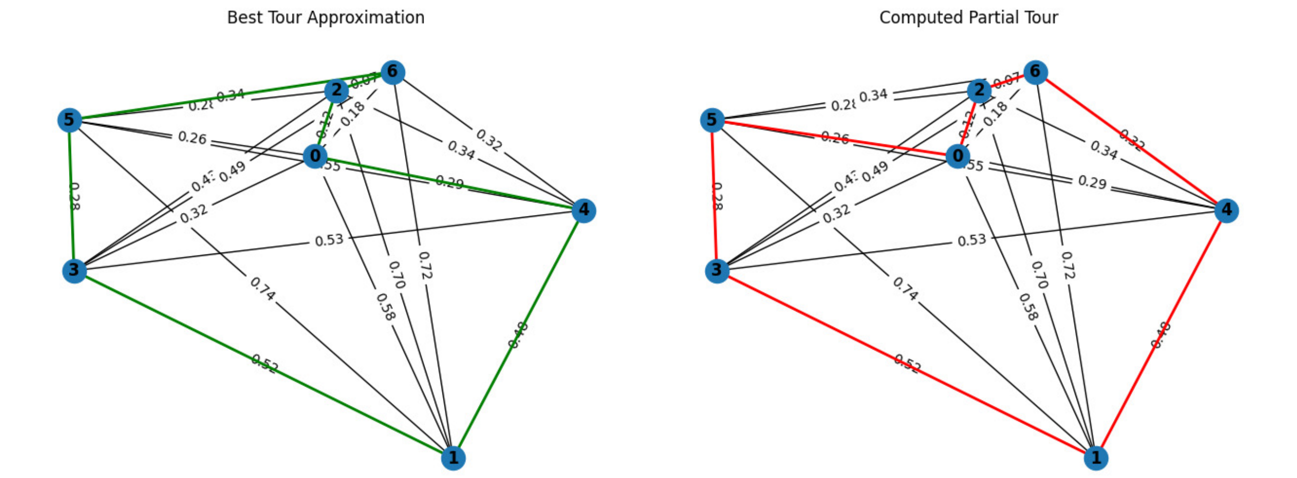

Fig. 4: Comparison of Christofides best tour approximation and EQC computed partial tour on a TSP instance
This figure presents a side-by-side comparison of two different approaches for solving TSP: the Christofides algorithm and the EQC method. On the left is the tour approximation as computed by Christofides' heuristic. On the right is the best tour found using the EQC method, which is able to generate better quality solution than Christofides for this particular graph. The length of the solution obtained using Christofides is 2.10, while the length of the solution obtained using EQC is 2.05. This serves as an example of how EQC can perform better than Christofides in certain cases.

5.1 Q-Values Computation

In the context of the TSP, computing Q-values is a crucial step for devising an effective reinforcement learning policy. These Q-values are instrumental in guiding the agent's decisions towards the most rewarding actions in any given state, enabling the solution of TSP instances with previously unseen configurations.

The distinct advantage of our Q-value estimation lies in the EQC, demonstrating superior capability in generating Q-values for the various state-action pairs during TSP problem-solving exercises. The EQC, which is central to our approach, naturally aligns with the symmetry properties of Ising Hamiltonians devoid of local fields, as outlined by Ozaeta et al. [15]. The EQC yields expectation values reminiscent of those used in the Quantum Approximate Optimization Algorithm (QAOA), fostering a beneficial comparison of performance standards between the two methods. Our work expands upon the QAOA by integrating a set of node features ($\vec{\theta}$), essential for incorporating TSP-specific information into the quantum computation framework.

By comparing Christofides and EQC approaches, we can see that EQC is sometimes able to find a tour that is closer to the optimal solution than what Christofides can achieve. This is because EQC uses a more sophisticated heuristic that takes into account the structure of the graph and the particular instance of TSP being solved. While Christofides provides a guaranteed-quality solution, the quality of the solution obtained may not always be the best possible. In cases where the optimal solution is difficult to obtain, EQC may be a better choice for solving TSP, as you can see on Fig. 4.

Fig. 5 presents an EQC solution. In this instance, it is evident that the EQC algorithm performs relatively worse than the Christofides approximation. However, it is important to note that this outcome does not imply any inherent shortcomings or limitations of the EQC algorithm itself.

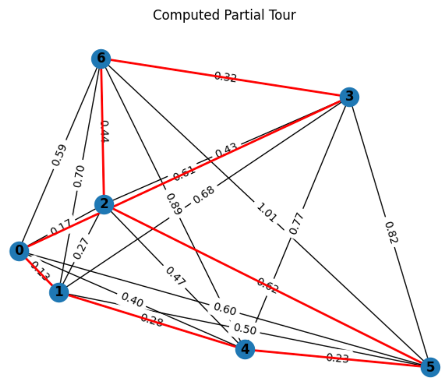

Fig. 5: EQC computed partial tour on a specific graph
This figure showcases an EQC solution for a specific TSP instance. The figure illustrates that, in this particular instance, the EQC model performs relatively worse than the Christofides approximation.

The performance of optimization algorithms, such as EQC and Christofides, can vary depending on the specific characteristics of the TSP instance being considered. Factors such as the number of nodes, the distances between them, and the overall structure of the problem instance can influence the effectiveness of different algorithms.

However, it is crucial to recognize that the EQC algorithm possesses its own unique strengths and capabilities. The EQC algorithm undergoes rigorous training to develop enhanced decision-making capabilities, which allow it to capture the symmetries and nuances of the TSP. It takes into account the closest proximity criteria and optimizes the route distance from the current location.

The EQC algorithm's advanced approach in decision-making involves strategic computation of Q-values, which not only identify the optimal path through minimum distance consideration but also incorporate adaptability to recognize multiple correct choices within the structure of TSP solutions. Therefore, the lower performance observed in this specific instance should not overshadow the potential effectiveness of the EQC algorithm in other TSP instances with different characteristics.

This comparative analysis provided valuable insights into the performance of the EQC algorithm and the Christofides approximation for the TSP. It is essential to consider various factors that influence algorithm performance and avoid asserting that the EQC algorithm is inherently worse based solely on one specific instance's outcome. Further investigations and analyses in diverse TSP instances would provide a more comprehensive understanding of the strengths and weaknesses of the EQC model in comparison to other classical or quantum approaches.

6 Conclusions

Our research embarked on an exploration of employing Equivariant Quantum Circuits (EQCs) for solving the Traveling Salesperson Problem (TSP), a prototypical combinatorial optimization problem that is notorious for its computational complexity. Through extensive simulations and rigorous analysis, we have substantiated that our EQC demonstrates promising performance on the TSP. When adequately trained, the circuit consistently selected the best action according to its policy.

The effectiveness of the EQC in navigating the solution space of TSP instances attests to its potential to exploit the symmetries inherent in the problem. This was enabled by the circuit's intrinsic property of equivariance, which ensures that the quantum states remain equivariant to the permutations of the nodes. Consequently, the EQC was not only able to recognize equivalent states but also to generalize from one instance to others, thus providing a more resource-efficient training regimen. Our findings advocate for a paradigm shift towards leveraging quantum computational frameworks for addressing tasks that traditionally overwhelm classical computational architectures. The EQC's adeptness at capturing the nuances of the TSP and outputting high-quality solutions is a step towards vindicating the transformative promise of quantum algorithms in real-world applications. However, high expressivity does not unconditionally translate to superior models. The terrain of high-dimensional optimization that quantum models navigate is fraught with training challenges such as barren plateaus and a susceptibility to overfitting. Despite these considerations, our EQC's performance indicates a compelling interplay between quantum expressivity and algorithmic efficiency that deserves further exploration.

Looking ahead, pivotal avenues for future work include probing the intricate interplay between quantum expressiveness and algorithmic performance at varying circuit depths. Analyzing whether and how quantum interference effects contribute to the success of the EQC will likely provide deeper insight. Additionally, scaling the EQC to solve larger TSP instances and investigating its performance against classical and other quantum approaches under varying conditions will be essential for establishing its practical utility.

In conclusion, our research contributes valuable empirical evidence that reinforces the potential of EQCs in solving complex combinatorial problems. It extends an invitation to the scientific community to further investigate EQCs, laying the groundwork for future quantum models that may one day surpass the limits of classical computation.

Code Availability

The code for the QGNNs coding parts of the project is available in this Github repository.[9].

On the other hand, the code used during all these first steps of the project in collaboration with Adrian Vargas[10] can be found on this Github public repository[11].

Acknowledgment

I extend my heartfelt gratitude to my friends and my mentors for their unwavering support and motivation, which have been pillars of strength throughout this research endeavor. Their constant encouragement and insightful discussions have been instrumental in the progress of this project.

A la meva família, pel seu suport incondicional i continu, i per la confiança disposada en mi en tot moment. And to my dog, whose loyal companionship and unwavering spirit provided solace and joy amidst the rigorous demands of research. Your presence has been a cheerful reminder of the simple pleasures of life.

References

[1] M. Cerezo, A. Arrasmith, R. Babbush, S. C. Benjamin, S. Endo, K. Fujii, J. R. McClean, K. Mitarai, X. Yuan, L. Cincio, and P. J. Coles, "Variational quantum algorithms," *Nature Reviews Physics*, vol. 3, pp. 625–644, 12 2020. [Online]. Available: http://arxiv.org/abs/2012.09265http://dx.doi.org/10.1038/s42254-021-00348-9

[2] J. Preskill, "Quantum computing in the nisq era and beyond," *Quantum*, vol. 2, 1 2018. [Online]. Available: http://arxiv.org/abs/1801.00862http://dx.doi.org/10.22331/q-2018-08-06-79

[3] J. Riu and A. Garcia, "Quantum machine learning," 2019.

[4] A. Abbas, D. Sutter, C. Zoufal, A. Lucchi, A. Figalli, and S. Woerner, "The power of quantum neural networks," *Nat Comput Sci*, 2020.

[5] J.-G. Liu, Y.-H. Zhang, Y. Wan, and L. Wang, "Variational quantum eigensolver with fewer qubits," *Physical Review Research*, vol. 1, 2 2019. [Online]. Available: http://arxiv.org/abs/1902.02663http://dx.doi.org/10.1103/PhysRevResearch.1.023025

[6] Y. Herasymenko and T. E. O'Brien, "A diagrammatic approach to variational quantum ansatz construction," *Quantum*, vol. 5, 7 2019. [Online]. Available: http://arxiv.org/abs/1907.08157http://dx.doi.org/10.22331/q-2021-12-02-596

[9] Github repository: https://github.com/yeray142/QML-QGNN
[10] Computer Science Student at Universitat Autònoma de Barcelona.
[11] Github repository: https://github.com/adriend1102/QML

[7] A. Skolik, M. Cattelan, S. Yarkoni, T. Bäck, and V. Dunjko, "Equivariant quantum circuits for learning on weighted graphs," *npj Quantum Information 2023 9:1*, vol. 9, pp. 1–15, 5 2023. [Online]. Available: https://www.nature.com/articles/s41534-023-00710-y

[8] J. Zhou, G. Cui, S. Hu, Z. Zhang, C. Yang, Z. Liu, L. Wang, C. Li, and M. Sun, "Graph neural networks: A review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 12 2018. [Online]. Available: https://arxiv.org/abs/1812.08434v6

[9] G. Verdon, T. Mccourt, E. Luzhnica, V. Singh, S. Leichenauer, and J. Hidary, "Quantum graph neural networks."

[10] J. Zheng, Q. Gao, and Y. Lu, "Quantum graph convolutional neural networks," *Chinese Control Conference, CCC*, vol. 2021-July, pp. 6335–6340, 7 2021. [Online]. Available: https://arxiv.org/abs/2107.03257v1

[11] A. Sopena, M. H. Gordon, D. García-Martín, G. Sierra, and E. López, "Algebraic bethe circuits," *Quantum*, 2022.

[12] P. Yan, L. Li, M. Jin, and D. Zeng, "Quantum probability-inspired graph neural network for document representation and classification," *Neurocomputing*, vol. 445, pp. 276–286, 7 2021.

[13] S. M. A. Shah, H. Ge, S. A. Haider, M. Irshad, S. M. Noman, J. Arshad, A. Ahmad, and T. Younas, "A quantum spatial graph convolutional network for text classification," *Computer Systems Science and Engineering*, vol. 36, pp. 369–382, 1 2021. [Online]. Available: https://www.techscience.com/csse/v36n2/41114/htmlhttps://www.techscience.com/csse/v36n2/41114

[14] S. Jerbi, C. Gyurik, S. C. Marshall, H. J. Briegel, and V. Dunjko, "Parametrized quantum policies for reinforcement learning," 2021.

[15] A. Ozaeta, W. van Dam, and P. L. McMahon, "Expectation values from the single-layer quantum approximate optimization algorithm on ising problems," *Quantum Science and Technology*, vol. 7, 12 2020. [Online]. Available: http://arxiv.org/abs/2012.03421http://dx.doi.org/10.1088/2058-9565/ac9013