QUARMEN

Erasmus Mundus Joint Master

QUANTUM SCIENCE & TECHNOLOGY

Quantum Reinforcement Learning using Quantum Optical Projective Simulation

Master Thesis

Presented by

Alexandre BERGERAULT

Internship carried out at

Quandela

Applications Engineers team

Vassilis APOSTOLOU

Master degree of Physics

Erasmus Mundus QUARMEN

Massy, France - August 2024

'I dream of a day when egoism no longer reigns in the sciences,
when we join forces to study, and instead of sending sealed envelopes to academics,
we hasten to publish our slightest observations if they are new, and add
"I don't know the rest".'

Evariste Galois (1811-1832)
Nitens lux, horrenda procella, tenebris aeternis involuta

Abstract

Projective Simulation (**PS**) is a pioneering model within the domain of *Reinforcement Learning* (**RL**) designed with the aim of conserving the interpretability of the learning process. It is characterized by its utilization of random excitation walks on a graph structure to direct the decision-making processes of an autonomous agent [1]. This model has been extended into the quantum domain through *Quantum Optical Projective Simulation* (**QOPS**) [2], which operates based on the principles of photon quantum walks within a specifically designed optical circuit. The QOPS framework distinguishes itself from classical PS by exploiting the intrinsic quantum mechanical properties of state superposition and entanglement, leading to significant improvements in the computational efficiency and problem-solving precision. It is noteworthy that QOPS has demonstrated its efficacy through successful applications in a variety of tasks that require single photon inputs [2]. Following on these advancements, the introduction of *multi-excitation projective simulation* (**mePS**) presents a very promising approach to overcome certain complex challenges that have not been addressed previously [3], [4].

The primary objective of this internship was to delve into the exploration of applications of QOPS that necessitate the use of both single and multi-photon inputs.

In the absence of a pre-established *multi-excitation QOPS* (**meQOPS**) framework, the development of specific applications required a thorough investigation into the distinctive characteristics and interaction dynamics of photons within optical circuits.

The initial phase of the research focuses on the implementation of a quantum game, specifically modeled on the well-known Prisoner's Dilemma paradigm [5]. The intention is to adapt an existing version that employs photon quantum walks for compatibility with Quandela's Quantum Processing Unit (QPU Altair). This project present a unique set of challenges, primarily stemming from the physical limitations and operational constraints inherent to the quantum computer.

Subsequent to this, the research led us to evaluate the feasibility of establishing a meQOPS framework. Upon its potential realization, we also want to apply this framework to strategically selected games, such as the invasion game, with the purpose of demonstrating a tangible proof of concept.

In conjunction with the main research goals, this internship also included a series of ancillary projects. Among these, one notable project was dedicated to the exploration of the correspondence between graph-theoretical structures and optical circuit setups. The overarching goal of this project is to develop a framework that can efficiently map theoretical graphs used for RL to an operational framework for physical quantum computing devices using few resources.

Another project involved implementing a probability recycling algorithm for Quandela's quantum computer. This algorithm is an error mitigation algorithm that specifically targets the challenge posed by high loss levels during experiments performed on complex optical circuits. By analysing and optimizing probabilities, it aims to reduce the impact of losses and improve overall performances of any algorithm. These ancillary investigations intend to contribute significantly to a complete examination of both the promising aspects and the inherent emblematic challenges of the emerging field of optical quantum computing.

The thesis is structured as follow: In an introduction (Section I) I present principles of quantum computing, the reasons for the interest in this field and different use cases. In Section II, I explain the construction of physical systems used for quantum information, detail the specificities of photonic quantum information and the optical components used in a photonic quantum computer. In Section III is introduced the Projective Simulation framework and the construction of a quantum version of it. I present the first use case in Section IV with the prisoners dilemma, its implementation on an optical circuit and the results obtained for this case. The second use case, dedicated to the multi-photon inputs, is shown in Section V and I present the framework developed for meQOPS alongside with an analysis of the results obtained from this framework. Finally I present an algorithm for an efficient optical circuit encoding of graphs and the probability recycling method respectively in Section VI and Section VII. The thesis ends with a conclusion in Section VIII that summarizes the work, developed framework and results.

The work produced regarding the implementation of Quantum Optical Projective Simulation in Section IV and especially the results presented in Section IV.2 will be published.

Contributions

I describe here the exact contributions to the work presented in this thesis. All the text presented is original and has been verified.

Section I - Introduction: The concepts presented are taken from various cited papers, books and attended lectures. Those concepts are basics of quantum information and quantum computing that can be easily found. The text was written on the basis of my understanding of those concepts.

Section II - Photonic Quantum Information Processing: The DiVincenzo criteria are well known concepts from quantum information that can be found in the cited papers. The rest is a presentation of Quandela's technology. The informations presented are given based on research papers, Quandela's website, my work experience and discussions with users of Quandela's technology.

Section III - Projective Simulation: The introduction of Projective Simulation and Directed Acyclic Graphs are taken from papers and is presented based my understanding of those paper and work from previous interns that worked on this project before me. The QOPS framework was developed by the Application Engineer team of Quandela. The presented information are given based on this work and discussion with users.

Section IV - Prisoner's Dilemma: The presented concept for the classical version of the prisoner's dilemma is well known in the literature and can be found in the mentioned papers. The specific optical quantum version and ECM version are the continuity of a previous interns work. However, the code implementation, the optimization techniques developed and the results that constitute the proof of concept for the viability of QOPS are my original work.

Section V - Multi-excitation Quantum Optical Projective Simulations: Multi-excitation Projective Simulation is a concept that already existed and that can be found in the literature. The Invasion Game is a use case that was developed for mePS and that is also present in the cited literature. The introduction of a quantum version (meQOPS framework) is the result of my work. My work ranges from the conceptualization of this framework, development of optimization techniques to the elaboration of qudit parallel circuit processing. The results and analysis presented in the last part of this section are also part of my work.

Section VI - Optical Circuit Encoding of Graphs: The initial technique presented in Section VI.1 is present in the literature and the basis of previous work for the implementation of ECM. The Clip Grouping technique was developed, tested and implemented by myself.

Section VII - Probability Recycling: The Probability Recycling technique was developed by the Theory team of Quandela and can be found in the mentioned papers. My contribution for this part consists in the implementation of this method and the tests performed on it along with the method used for the tests.

Acknowledgements

I particularly thank Vassilis Apostolou from the Application engineering team of Quandela that supervised me for this work. I also thank Arno Ricou and all the Application Engineering team with which I shared this work and that provided me important feedback.

Another thank to Giacomo Franceschetto and Joaquim Minarelli Gaspar that worked on the QOPS project before my arrival in Quandela.

Abbreviations

Noisy Intermediate-Scale Quantum (**NISQ**) | *Quantum Approximate Optimization Algorithms* (**QAOA**) | *Variational Quantum Algorithms* (**VQA**) | *Reinforcement Learning* (**RL**) | *Quantum Processing Unit* (**QPU**) | *Projective Simulation* (**PS**) | *Quantum Optical PS* (**QOPS**) | *multi-excitation PS* (**mePS**) | *Directed Acyclic Graphs* (**DAG**) | *Episodic Compositionnal Memory* (**ECM**) | *Beam Splitter* (**BS**) | *Phase Shifter* (**PS**)[1] | *Knill-Laflamme-Milburn* (**KLM**) | *quantum machine learning* (**QML**) | *Mach-Zehnder Interferometer* (**MZI**) | *Finite Difference Stochastic Approximation* (**FDSA**) | *Simultaneous Perturbation Stochastic Approximation* (**SPSA**) | *eXplainable Artificial Intelligence* (**XAI**) | *Multi-excitation QOPS* (**meQOPS**) | *Hong-Ou-Mandel* (**HOM**) | *Strong Linear Optics Simulator* (**SLOS**) | *Qudit Parallel Circuit Processing* (**QPCP**)

[1] context dependent

Contents

I Introduction

Quantum computing, which began around 50 years ago, has sparked a period of remarkable breakthroughs and innovations. Yet, the exact moments and areas where this technology will make its most substantial impact are still unknown.

The first and most direct applications of quantum computing lies in the simulation of quantum systems [6], [7]. *Quantum bits*, or **qubits**, are invested with the same advantageous quantum properties such as superposition and entanglement. Those properties enable the efficient simulation of quantum systems without succumbing to the exponential complexity of classical computing approaches [8].

Quantum properties such as photon polarization, spin orientation, or any attribute that can be represented within a two-dimensional Hilbert space, can be succinctly expressed as

$$|\psi\rangle = \alpha\,|0\rangle + \beta\,|1\rangle$$

where α and β are complex numbers that satisfy the normalization condition

$$|\alpha|^2 + |\beta|^2 = 1$$

Consequently, to fully characterize a single element system, one only requires to know the amplitude of one complex number (with the other being deduced) and the phase difference between the numbers.

In a system comprising n elements, each endowed with this quantum property, a full description should take into account every possible linear combination.

$$|\phi\rangle = \sum_i C_i\,|\varphi_i\rangle$$

with $|\varphi_i\rangle$ the combinations of all orthogonal qubits state.

Thus the information to be stored grows rapidly. We say that the associated Hilbert space scales exponentially, exactly 2^n. For a classical computer to process this vast amount of information, it will suffer from a complexity of $\mathcal{O}(2^n)$.

In stark contrast, quantum computers possess a significant advantage as they can inherently store and manipulate this information by preparing the system $|\phi\rangle$. This property allows for a complexity scaling linearly with the number of element in the system, $\mathcal{O}(n)$ [9].

The advantages of quantum computers extend well beyond the simulation of quantum systems; they can be adapted to address a diverse array of complex problems. The *Boson Sampling* problem and *Quadratic Unconstrained Binary Optimization* (**QUBO**) are examples of problems for which we benefit by using quantum algorithms [10], [11].

There exists a variety of quantum computers, each employing distinct approaches to harnessing quantum phenomena. The most widely recognized is the *Gate-based quantum computer*, which manipulates qubits through unitary transformations executed by various interacting Hamiltonian, known as *gates* [12]. *Quantum Annealing-based quantum computers* are particularly efficient in solving optimization problems [13]. Within the scope of my internship, my primary focus will be on *Photonic-based quantum computers*.

The aim of all these quantum computers is to solve, using fewer resources, problems that are deemed *hard* for classical computers. In this context, *hardness* refers to computational complexity, indicating the amount of time and resources required to solve a problem. This scenario is commonly referred to as *quantum advantage* [14]–[16].

In photonic quantum computers, the information is encoded with photons. They navigate through tunable optical circuits composed of multiple optical fibers and optical components. Optical components such as beam splitters and phase shifters are strategically employed to manipulate this information [17].

Although a handful of algorithms have been proposed to achieve quantum advantage, the progress in hardware development for quantum computation remains a significant bottleneck, despite considerable advancements in recent years.

We are currently in the era of *Noisy Intermediate-Scale Quantum* (**NISQ**) devices [18]. This means that we are working with computers that have a limited number of qubits, typically ranging from 10 to 100, which is sufficient for a multitude of applications. However, these qubits are subjected to significant levels of noise, preventing the execution of long fault-tolerant algorithms. Implementing error correction algorithms remains highly challenging and requires to dedicate a portion of the computer's resources to this task [19].

Despite the limitations of NISQ computers, there exists algorithms that can effectively operate within these constraints. One of those is *Quantum Approximate Optimization Algorithms* (**QAOA**)[20]. These algorithms belong to a different class of complexity and operate by approximating solutions to optimization problems. They terminate when the solution found by the program is sufficiently close to the actual solution. Consequently, finding a real solution with a small amount of noise added in the calculation can, in certain cases, lead to an equivalence with Approximation algorithms.

Variational Quantum Algorithms (**VQA**) [14], [21] represent another category of algorithms well-suited for noise-tolerant environments. These algorithms offer a hybrid approach, combining classical and quantum elements. The quantum portion of the algorithm potentially gives a speedup or advantage compared to a fully classical algorithm, while requiring fewer qubits compared to a fully quantum algorithm. The key advantage of VQA lies in its ability to leverage quantum computers with a limited number of qubits, thereby mitigating the impact of noise.

In this Thesis, I will delve into the exploration of a *Reinforcement Learning* (**RL**) method, called *Projective Simulation* (**PS**) applied to optical quantum computers. My initial endeavor in this thesis involves implementing the prisoner's dilemma on Quandela's *Quantum Processing Unit* (**QPU**), with meticulous consideration given to the specific limitations inherent in these computers. The objective is to establish a proof of concept for *Projective Simulation* (**PS**) on these platforms. To achieve this objective, I will use *Quantum Optical Projective Simulation* (**QOPS**), a VQA framework developed for the application of PS on QPU.

Following the initial implementation, I will develop a VQA framework using multiple photons to explore the potential of *multi-excitation PS* (**mePS**) within the context of optical computing. As a proof of concept for the framework, I will implement it to address multiple instances of the invasion game.

Additionally, I will investigate methodologies for formalizing and implementing strategies aimed at efficiently transcribing *Directed Acyclic Graphs* (**DAG**) into optical circuits. This effort seeks to optimize the computational efficiency of various algorithms based on RL with *Episodic Compositionnal Memory* (**ECM**). I will also investigate probability recycling, an error mitigation algorithm developed to improve the probability estimation of output states for lossy circuits.

II Photonic Quantum Information Processing

II.1 Photonic Qubit

DiVincenzo Criteria

The DiVincenzo criteria, formulated by the physicist David DiVincenzo, are a set of five pivotal conditions that must be met for a physical system to be a viable candidate for quantum information processing [22]. These criteria are foundational in the field of quantum computing, providing a standard for the development and evaluation of quantum systems. They are as follows:

- **Isolation and Replication** : The physical system must be capable of being isolated from its environment to prevent decoherence. Decoherence is the loss of quantum information due to interactions with the surroundings. Furthermore, the system must be replicable, ensuring that each qubit can be consistently produced with identical properties.
- **Initialization** : The system should allow for the initialization of qubits into a known state, typically either of the two computational basis states, denoted as $|0\rangle$ or $|1\rangle$. This initialization is crucial as it sets the starting point for quantum computations.
- **Measurement** : The ability to measure the state of qubits is essential for reading out the result of quantum computations. The system must allow for accurate and reliable measurements within a predetermined basis. The measured qubit states should suffer from no bias in the quantum information encoded within them.
  Unlike Quantum Non-Demolition (QND) measurements where the destruction of the systems state is avoided, only the information needs to be preserved to fulfill this condition.
- **Superposition** : The system should support the manipulation of qubits to prepare superpositions of states. Superposition is a fundamental principle of quantum mechanics and allows qubits to exist in a linear combination of the $|0\rangle$ and $|1\rangle$ states. This is a strong feature for the parallel processing capabilities of quantum computers.
- **Coherence time** : The qubits must exhibit a sufficiently long coherence time. Coherence time is the duration over which qubits can maintain their quantum state. A longer coherence time allows for more complex quantum operations to be performed before decoherence sets in.

In addition to these five criteria, the ability to perform **entanglement** between qubits is often considered an implicit sixth criterion. Entanglement is a unique quantum phenomenon where the state of one qubit becomes dependent on the state of another. More strictly said, entanglement is a non-local correlation of qubit states. This property is a key feature for quantum computers to perform operations that are impossible for classical computers [8].

The theoretical construction of quantum circuits based on linear optics necessitates a qubit encoding that facilitates the definition of the computational basis states and the generation of quantum superpositions. This encoding is a critical aspect of the design of quantum circuits, as it determines how quantum information is represented and manipulated within the system.

Spatial Qubit Encoding

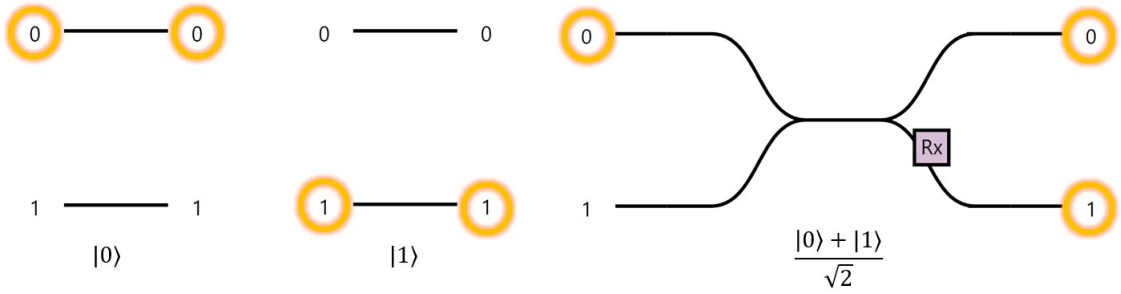


Figure 1: *Spatial encoding for optical circuit with highlighted input and output excitation. From left to right is shown the optical encoding of states* $|0\rangle$, $|1\rangle$ *and* $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$. *The* R_x *box shown in the right-hand side indicates which convention is used to tune the Beam Splitter.*

Spatial encoding is a method employed by Quandela's computer architecture [23], as depicted in Figure 1. This approach involves inputting single photons into labeled fibers corresponding to different logical states. This method thereby satisfy the initialization and preparation criteria of the DiVincenzo checklist. Spatial encoding offers a versatile

framework for the manipulation of information within photonic quantum computers. For instance, a qubit can be encoded in a superposition state such as

$$\frac{|0\rangle + |1\rangle}{\sqrt{2}}$$

The interpretation of the encoded information is adjusted at the end of the computation process, allowing for a wide range of manipulations to be performed. It is noteworthy that spatial encoding is not limited to single-photon states; it can also be extended to states which do not fit that encoding, using several photon in input, for instance any kind of Fock state such as

$$|7, 11\rangle$$

broadening the scope of potential applications in quantum computing.

While spatial encoding is the primary method supported by Quandela's computers, other encoding techniques, such as *polarization encoding*, are also frequent in the field of photonic quantum computing [17], [23]. Each encoding method offers its own set of advantages and challenges, contributing to the diversity of approaches in the development of quantum computers.

II.2 Optical Networks and Components

Optical networks within photonic quantum computers are composed of various optical components that manipulates photons. The primary components used in Quandela's *Quantum Processing Unit* (**QPU**) are *Beam Splitters* (**BS**) and *Phase Shifters* (**PS**). They play a crucial role in the operation performed within the computer [24], [25].

Basic Optical Components

Phase shifters (**PS**) are devices that act on individual spatial modes. They alter the phase of the photon by a chosen amount, denoted here φ. The operation of a phase shifter on a quantum state $|\psi\rangle$ can be mathematically written as

$$|\psi\rangle \rightarrow e^{i\varphi} |\psi\rangle$$

This transformation is essential for controlling the phase relationships between different components of a quantum state. This is a key aspect to make use of quantum interferences.

Beam splitters (**BS**) on the other hand, operate on two spatial modes. We characterize those modes by their photon numbers denoted α and β. BS are described by an angle θ that encodes the transformation the BS applies on the input photons. We denote $a^\dagger$ and $b^\dagger$ the photon creation operators in mode A and B respectivelly. The mathematical representation of a beam splitter operation on an initial state is given by:

$$\left(a^\dagger\right)^\alpha \left(b^\dagger\right)^\beta |\varnothing\rangle \rightarrow \left(\cos\left(\frac{\theta}{2}\right) a^\dagger + i \sin\left(\frac{\theta}{2}\right) b^\dagger\right)^\alpha \left(i \sin\left(\frac{\theta}{2}\right) a^\dagger + \cos\left(\frac{\theta}{2}\right) b^\dagger\right)^\beta |\varnothing\rangle$$

The matrix representation of a beam splitter is thus:

$$\begin{pmatrix} \cos\left(\frac{\theta}{2}\right) & i \sin\left(\frac{\theta}{2}\right) \\ i \sin\left(\frac{\theta}{2}\right) & \cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

This representation highlights the beam splitter's ability to mix the amplitudes of the input photons, resulting in a new quantum state that is a combination of the original spatial modes.

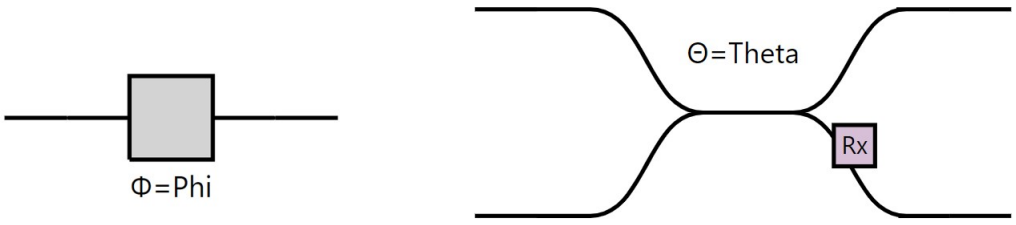


Figure 2: *Graphical representation of a Phase Shifter on the left side and a Beam Splitter on the right side with Quandela's framework Perceval. The R_x box indicates the convention used to tune the Beam Splitter.*

Quandela's framework introduces a generalized version of the beam splitter, as shown in Figure 3. This advanced version incorporates additional phase factors, allowing for a more comprehensive manipulation of the photonic state.

The generalized beam splitter operation can be expressed as the following:

$$\begin{pmatrix} e^{i(\phi_{tl}+\phi_{tr})}\cos\left(\frac{\theta}{2}\right) & ie^{i(\phi_{tr}+\phi_{bl})}\sin\left(\frac{\theta}{2}\right) \\ ie^{i(\phi_{tl}+\phi_{br})}\sin\left(\frac{\theta}{2}\right) & e^{i(\phi_{br}+\phi_{br})}\cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

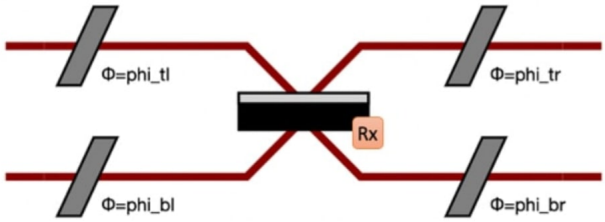


Figure 3: *Generalized version of the Beam splitter, including multiple phase shifters for a broader range of manipulation.*

This matrix includes phase factors $\phi_{tl}, \phi_{tr}, \phi_{bl}$ and ϕ_{br}, which correspond to the top-left, top-right, bottom-left, and bottom-right corners of the beam splitter, respectively. These phase factors provide an additional degree of control over the photonic state, enabling more sophisticated quantum operations.

Another important element for quantum computing using optical networks is the *permutation* operation, which allows for the reordering of indices corresponding to multiple spatial modes. This operation simplifies the manipulation of quantum states by rearranging the spatial modes in a desired sequence. Figure 4 illustrates the concept of permutation between modes 0 and 1. While permutation is a valuable element in the programming of quantum circuits, it is not directly implemented in the hardware of the computer. Instead, a series of beam splitters are arranged to achieve the equivalent functionality, effectively reordering the spatial modes as required.

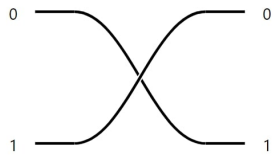


Figure 4: *Permutations between modes 0 and 1. This operation is generalized in Quandela's Perceval framework for any number of modes.*

Entanglement with Linear Optical Circuit

Entanglement is a phenomenon where quantum states of two or more objects become interconnected such that the state of one cannot be described independently of the others. entanglement is a fundamental aspect of quantum mechanics and a powerful resource in quantum computing and information processing [26]. It enables the execution of quantum algorithms that can outperform their classical counterparts for certain tasks. However, generating entanglement in photonic systems presents particular difficulties [23], [27]. Photons do not interact strongly with each other in linear optical circuits, making the traditional methods of entangling qubits, such as those used in *gate-based* quantum computers, inapplicable.

Quandela's approach to photonic quantum computing relies on single-photon sources. It can produce entangled photons but this feature is not yet implemented in the software. Thus we need to use other techniques in order to achieve this entanglement. To exploit the power of entanglement, we must employ innovative techniques that enable the generation of entangled photon states using only linear optical components. Quandela's approach to this issue is the use of *post-processing* of states, also called *heralded gate*. The most know of these heralded gate being the **KLM CNOT** [27].

In conventional gate-based quantum computers, entanglement is often achieved through the use of two-qubit gates, such as the Controlled-NOT (CNOT) gate [28], [29]. The CNOT gate is a fundamental quantum logic gate that flips the state of a target qubit conditional to the state of a control qubit. For example, if we start with an initial state $|0,0\rangle_{A,B}$ and apply a Hadamard gate to the first qubit, we obtain the state

$$\frac{(|0\rangle + |1\rangle)_A}{\sqrt{2}} \otimes |0\rangle_B$$

The subsequent application of a CNOT gate to this superposition state results in the creation of a maximally entangled state, represented as

$$\frac{|00\rangle_{AB} + |11\rangle_{AB}}{\sqrt{2}}$$

In systems that utilize superconducting qubits, the CNOT gate is typically implemented using precisely timed microwave pulses that induce the necessary quantum interactions between the qubits [30]. However, the implementation of a CNOT gate within the framework of linear optical computing requires a different strategy due to the non-interacting nature of photons [31].

The *Knill-Laflamme-Milburn* (**KLM**) [27] protocol provides a solution for achieving entanglement in optical circuits. The KLM CNOT gate operates on a probabilistic basis and relies on a technique known as heralded post-selection. This method involves some manipulation of photons through a carefully arranged series of linear optical elements. The actual CNOT operation occurs with a certain probability, and additional measurements are used to identify and select the instances where the operation was successful.

The heralded nature of the KLM CNOT gate scheme is practically characterized by the measurement of photons in specific output modes. In a typical setup, the simultaneous detection of one photon in each of the designated modes, often labeled as modes 4 and 5 in Figure 5, indicates that the CNOT operation has been successfully performed. If the expected photons are not detected in these modes, the operation is considered to have failed, and the corresponding experimental iteration is discarded [27].

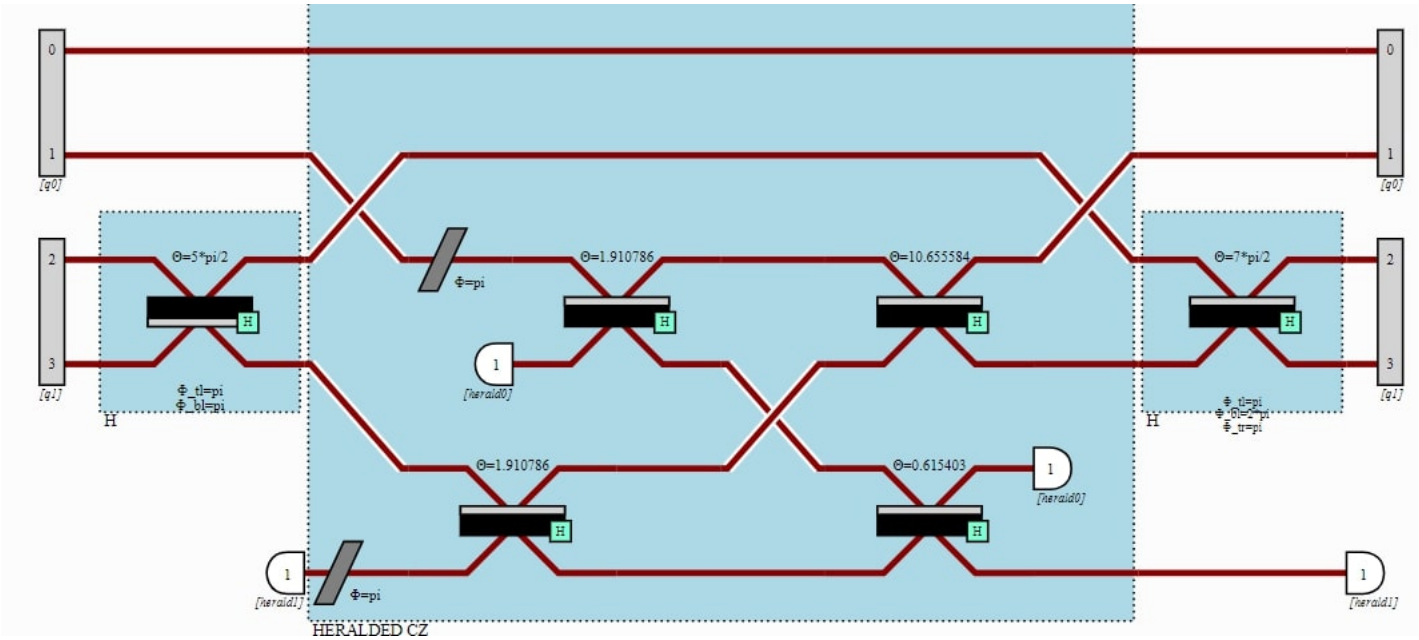


Figure 5: *6 modes - 4 photons heralded CNOT gate (KLM). The measurement of photons in the boxes denoted as herald indicates a successful operation.*

The probabilistic nature of the KLM CNOT gate implies that the desired entangling operation is achieved with a certain probability. In the case of the simplest KLM gate the success probability is $\frac{2}{27}$. Despite this limitation, the KLM CNOT gate is a valuable asset in the toolkit of photonic quantum computing, as it enables the implementation of entanglement and facilitates a range of quantum operations within optical circuits. The development of such probabilistic entangling gates represents a significant step forward for the realization of practical quantum computation using photons [32], [33].

III Reinforcement Learning and Projective Simulation

III.1 Concepts of Projective Simulation with Mathematical Structures

Reinforcement learning (**RL**) is a specialized *Machine Learning* (**ML**) approach where the decision-making process adapts based on the interaction between an agent and its environment [34]. This dynamic process involves an agent that learns to achieve a goal in an uncertain, potentially complex environment by trial, error and feedback in the form of rewards or penalties. The agent's actions are not pre-programmed; instead, they are determined by a policy that evolves as the agent learns from its experiences.

Formally, RL can be modeled as a Markov decision process [34] characterized by the set

$$\{\mathcal{S}, \mathcal{A}, \mathcal{T}(s_{t+1}|s_t, a_t), \mathcal{R}(r_{t+1}|s_t, a_t)\}$$

with:

- $\mathcal{S} = \{s_i\}$ the set of possible environment states.
- $\mathcal{A} = \{a_i\}$ the set of possible actions for the agent.
- $\mathcal{T}(s_{t+1}|s_t, a_t)$ the transition function between states (mapping from state to action).
- $\mathcal{R}(r_{t+1}|s_t, a_t)$ the reward function.

At each time step t, the agent observes the environment state, computes and executes an action based on the transition function $\mathcal{T}$. Subsequently, the environment provides a reward and can change or not its state based on the agent's action. The agent then updates $\mathcal{T}$ according to the reward and repeats the process.

The agent's objective is to devise a policy $\pi(\mathcal{S}, \mathcal{A})$ to maximize the long-term reward for all environment states. This policy is essentially a strategy that the agent follows to decide which action to take in a given state.

It's noteworthy that the agent's aim is to optimize the reward over the long term. For instance, in training an agent to play chess, favoring immediate piece captures might lead to long-term losses, whereas prioritizing cumulative rewards could result in eventual victory.

Projective simulation (**PS**) is a specific RL method introduced by [35], [36]. The main goal of PS is to maintain a logical interpretability of intermediate transitions and states within the agents memory: the *Episodic Compositional Memory* (**ECM**) [37]. The ECM is a network of clips, which are memory fragments that represent perceptual inputs, actions, or sequences of events that the agent has experienced. These clips are connected by edges that represent the transition probabilities between them, and they are updated based on the agent's experiences and the received rewards.

In PS, the agent navigates through the ECM by a random walk process, starting from a clip representing the current perceptual state and moving through the network until it reaches an action clip. An example of the PS process is shown in Figure 6. The path taken through the ECM during this random walk is influenced by the weights of the edges, which are adjusted based on the rewards received, thus reinforcing paths that have led to higher rewards in the past.

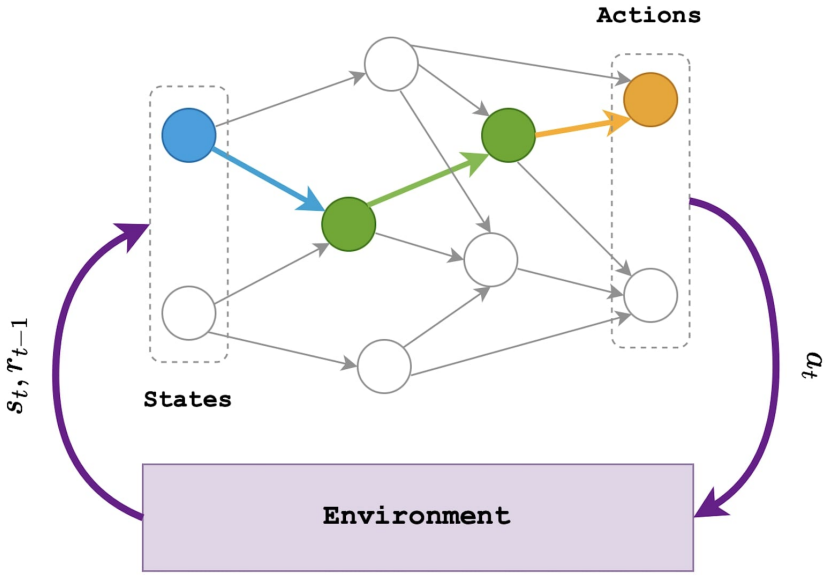


Figure 6: *Representation of the ECM and connection with the environment. A random walk is performed at each epoch, from which an action is taken. The environment gives the agent a reward based on the action and its new state. The agent updates the ECM and start a new epoch.*

One of the key features of PS is its ability to simulate future actions by projecting potential outcomes based on past experiences. This allows the agent to evaluate the potential consequences of its actions before actually executing them in the environment. The PS model also incorporates mechanisms for creativity and variability in the agent's behavior, such as the introduction of random perturbations in the random walk process, which can lead to the exploration of new paths and strategies.

The flexibility and adaptability of the PS model make it suitable for a wide range of applications, from robotics to game playing to autonomous systems [37]. Its emphasis on episodic memory and the ability to project and evaluate potential future scenarios gives it an edge in environments where the ability to anticipate and plan for future events is crucial for success.

III.2 Directed Acyclic Graph as Episodic Compositional Memory

In the realm of Projective Simulation, the concept of *Episodic Compositional Memory* (**ECM**) plays a pivotal role in the decision-making processes of agents. This memory of the system can be visualized as a *Directed Acyclic Graph* (**DAG**) [38], [39], where the nodes represent discrete events or states that an agent has encountered, and the edges denote the possible transitions between these events. Such a structure allows for the representation of complex sequences of actions and experiences without the risk of creating cycles, which could lead to infinite loops in the decision-making process.

The DAG serves as a map of the agent's knowledge. Each *vertex* (*or clip*) corresponding to a specific piece of information or a decision point, and each *edges* representing the logical progression from one piece of information to the next. When a stimulus or excitation occurs in state (input) clips (depicted in green in Figure 7), it signifies the agent's current understanding or perception of the environment. The agent then embarks on a random walk of excitation through the DAG, traversing from one clip to another, until it reaches an action clip (depicted in red in Figure 7). This action clip corresponds to a decision or an action that the agent will take to interact with the environment.

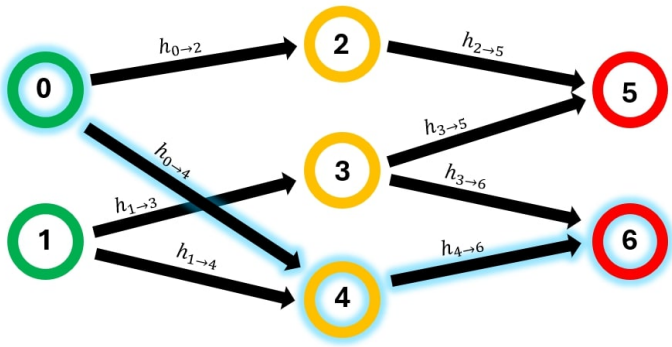


Figure 7: *Example of Directed Acyclic Graph. Here is shown a randoma walk performed. The agent triggers the input clip o. Then the walk performed is* $0 \to 4 \to 6$. *Once the clip 6 is triggered, the corresponding action is taken by the agent.*

The complexity of the task at hand determines the number of intermediate clips and connections required within the DAG [40]. For simple tasks, the graph may be relatively sparse, with few intermediate clips between the initial state and the action. However, as tasks become more complex, the number of intermediate clips and the density of connections between them increase, allowing for a more nuanced and sophisticated decision-making process.

The mapping between the environment state and the state clip is crucial for the agent's ability to navigate the DAG effectively. This mapping ensures that each possible state in $\mathcal{S}$ translates into at least one excitation configuration

$$f : \mathcal{S} \to \mathcal{S}^\star$$

where $S^\star = \{s_i^\star\}$ is the set of input clips. Here f is the morphism associated with the mapping from the environment to the state clip. Similarly, the mapping between the action clip and the executed action by the agent ensures that for $A^\star = \{a_i^\star\}$ the set of action clips, there exists a morphism $f' : \mathcal{A}^\star \to \mathcal{A}$ that associates each action clip with a corresponding action in the environment.

Each edge of the graph $h_{i\to j} \in \mathcal{H}$ is associated with a non-normalized weight $w_{i\to j} \in \mathcal{W}$. These weights play a critical role in determining the transition probabilities between clips. The transition probability from any clip i to another clip j is given by

$$p_{i\to j} = \frac{w_{i\to j}}{\sum_k w_{i\to k}} \in \mathcal{P}$$

This probabilistic framework allows the agent to explore different paths through the DAG, favoring transitions that have been reinforced through positive experiences and rewards.

In the context of Projective Simulation, the term "compositional" in ECM highlights the fact that each intermediate transition involving an intermediate clip serves as a fictitious experience that the agent can leverage during the learning process. The compositional nature of ECM enables the agent to build a broad range of hypothetical scenarios and outcomes, which it can draw upon when faced with new situations or challenges.

The use of a DAG as ECM provides several advantages. It allows for efficient storage and retrieval of experiences, as each clip can be accessed directly without the need to traverse the entire graph. It also facilitates the parallel processing of information, as multiple paths can be explored simultaneously. Furthermore, the acyclic nature of the graph ensures that the agent's decision-making process is always moving forward, preventing it from getting stuck in repetitive or unproductive cycles.

III.3 Classical Learning Procedure for PS

General Reward Mechanism

The learning procedure within the context of episodic compositional memory (ECM) begins with the initialization of weights. These weights are critical as they represent the strength of the connections between the various clips within the memory structure. To ensure a fair starting point and to minimize any initial bias towards particular action clips, the weights are typically set uniformly or randomly. For the sake of simplicity, let's assume that all weights are initialized to a value of 1. This uniformity provides a neutral baseline from which the learning process can evolve.

As the agent operates within its environment, it encounters various states that trigger corresponding state clips within the ECM. These state clips serve as the starting points for the excitation to propagate through the network. The excitation moves from clip to clip, following the edges of the graph, until it reaches an action clip. The sequence of clips that the excitation traverses forms a path, which we denote as

$$\Gamma_t = (c_0, c_1, \ldots, c_D, c_{D+1}\}$$

for a given time step t, with D denoting depth of the ECM. The first clip in this sequence c_0 is the state clip that was initially excited, and the last clip c_{D+1} is the action clip that ultimately leads to an interaction with the environment.

Following the execution of an action, the agent receives a reward at the subsequent time step $t+1$. This reward, denoted as r_t is then used to update the ECM according to a specific reward learning rule [36]. The rule dictates that at the end of each time step t, the weights of the edges along the path Γ_t are adjusted in response to the received reward. The adjustment is made as follows:

$$w^{t+1}_{c^i,c^{i+1}} = w^t_{c^i,c^{i+1}} + r_t$$

This equation signifies that for a positive reward, the connections along the path are reinforced, thereby increasing the likelihood of the excitation following the same path in the future. Conversely, for a negative reward, the connections are weakened, reducing the probability of repeating the same sequence of actions.

Glowing and Forgetting Mechanism

In addition to the reward mechanism, the ECM incorporates other mechanisms to account for changes in the reward rules and long-term results [37]. One such mechanism is the forgetting mechanism, which enables the ECM to adapt to dynamic changes in the environment by gradually diminishing the influence of past experiences. This is achieved by updating the weights according to the following rule:

$$w^{t+1}_{c^i,c^{i+1}} = w^t_{c^i,c^{i+1}} + \gamma(w^t_{c^i,c^{i+1}} - 1)$$

In this equation, γ represents the *forgetting factor*, a parameter that controls the rate at which the impact of previous experiences fades. The forgetting mechanism ensures that the ECM remains flexible and responsive to new information, preventing it from becoming overly reliant on outdated or irrelevant experiences.

Concurrently, the glowing mechanism facilitates the learning of actions associated with delayed rewards. This mechanism operates in tandem with the reward mechanism, modifying the weight update rule to incorporate a glowing factor. The updated rule is expressed as:

$$w^{t+1}_{c^i,c^{i+1}} = w^t_{c^i,c^{i+1}} + g^t_{i\to j} r_t$$

The *glowing factor*, denoted as $g^t_{i\to j}$ is determined by the following conditions:

$$g^{t+1}_{i\to j} = \begin{cases} 1 & \text{if } c_i \to c_j \text{ was taken at time step } t \\ (1-\eta)g^t_{i\to j} & \text{otherwise} \end{cases}$$

At the initial time step, the glowing factor for each edge is set to 0, indicating that no paths have been reinforced yet. The parameter η, which lies within the range [0,1], controls the speed of the glowing process. This mechanism allows the ECM to account for the temporal aspect of rewards, ensuring that actions leading to delayed gratification are appropriately valued and learned.

III.4 Quantum Optical Projective Simulation

Quantum Optical Projective Simulation (**QOPS**) is a sophisticated framework for quantum-enhanced artificial intelligence that allows for the implementation of learning processes in optical circuits for quantum computers [2]. This innovative approach to *quantum machine learning* (**QML**) leverages the unique properties of quantum mechanics to facilitate complex simulations and decision-making processes that are not possible with classical computing methods.

Unlike classical projective simulation, the QOPS framework introduces the advantages of quantum computations such as superposition, entanglement, and quantum interferences. These quantum characteristics enable a more efficient exploration of the state space compared to the classical random walk, potentially leading to faster learning and decision-making processes. In QOPS, the quantum walks are performed by the propagation of photons within an optical circuit.

The quantum walks in the QOPS framework allow for a more dynamic and interconnected exploration of possible states. This is due to the quantum nature of the particles involved, which can exist in a superposition of states and can be entangled with each other. Quantum interferences, allows for the constructive and destructive superposition of probabilities. All those properties can potentially lead to more optimal and fast decision-making pathways.

The aim of using the QOPS framework is to achieve a quantum speedup for the learning process using the quantum processes introduced [2], [41]. However, more adjustments to the learning procedure need to be done to tackle several issues. These adjustments are necessary to fully harness the potential of quantum computing and to address the challenges that arise from the quantum nature of the learning process.

As a quantum walk between the state-clips and the action-clips is performed, we cannot keep track of the path taken by the excitations. This manipulation requires to measure the photon state at each step of the circuit. Such measurements would lead to the collapse of quantum superpositions, thereby negating the quantum speedup advantage. This is a fundamental challenge in quantum computing, where the act of observation can alter the state of the system being observed.

In the QOPS framework, instead of training each individual parameter depending on the path that was taken during the random walk, we are going to tune at once all the parameters of the circuit in order to act on the complete excitation propagation. This holistic approach to parameter tuning allows for a more global optimization of the learning process. This procedure takes into account the entire landscape of edges rather than isolated paths [2]. We describe this procedure in Section III.4.

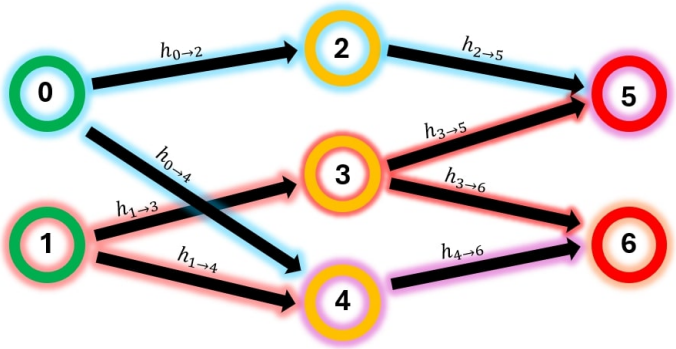


Figure 8: *Example quantum ECM with with a superposition of states* $|0,1\rangle$ *and* $|1,0\rangle$ *as input. The agent triggers the input clips by adding in input the state* $\frac{|0\rangle+|1\rangle}{\sqrt{2}}$*. Then the following transformations are made:* $|0\rangle \to P_{h_{0\to2}}|2\rangle + P_{h_{0\to4}}|4\rangle$*,* $|1\rangle \to P_{h_{1\to3}}|3\rangle + P_{h_{1\to4}}|4\rangle$ *and for the second layer* $|2\rangle \to P_{h_{2\to5}}|5\rangle$*,* $|3\rangle \to P_{h_{3\to5}}|5\rangle + P_{h_{3\to6}}|6\rangle$ *and* $|4\rangle \to P_{h_{4\to6}}|6\rangle$*.*

The quantum equivalent of the random walk is represented in Figure 8. This representation presents the evolution of the quantum walk, where the probabilities of transitioning between clips transforms into a superposition of states with different amplitudes and phases.

Superposition also allows to compute all the possible outcomes using a single input state and repeating the quantum walk for a significant amount of time.

To train the set of parameters, we act on the weights but also on the phases of the transmitted excitation. This allows us to play with interferences between the states. This manipulation of phases is crucial for the quantum walk, as it can lead to different interference patterns, which in turn can significantly affect the efficiency of the learning process.

The question remains: How to encode this quantum ECM into an optical circuit ? A method is introduced in [4], [42] and is the basis of the implementations performed in this thesis.

The proposed method involves creating a circuit which contains as many spatial modes as the ECM has nodes. We then introduce the transitions between the clips of the ECM by the use of optical components to link the modes. An example based on the ECM of Figure 8 is given in Figure 9.

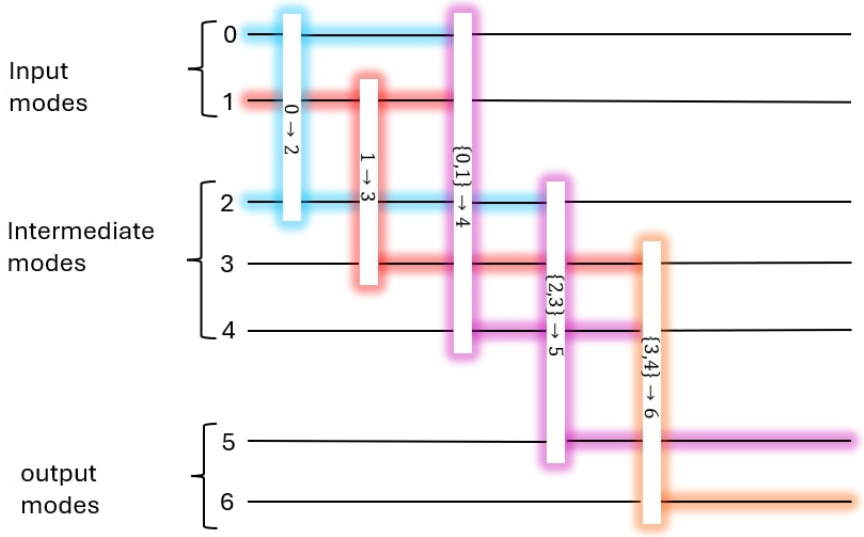


Figure 9: *Optical circuit equivalent of Figure 8, using the QOPS framework as encoding.*

In the example given in Figure 9, each link between two modes is equivalent to a *Mach-Zehnder Interferometer* (**MZI**), as shown in Figure 10. The MZI encodes a fully tunable unitary operation by adjusting the phase shifters. This tunability is essential to manipulate the interference patterns that are crucial for the quantum walk.

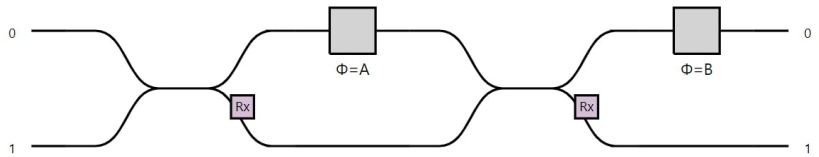


Figure 10: *Representation of a Mach-Zender Interferometer.*

To have a link with more than two input modes, it suffices to connect step by step all the input modes ($\{0,1\}, \{1,2\}, \ldots, \{n-1,n\}$). It is then possible to add a transition between the last input mode to the output mode. This modular approach allows for the construction of complex optical circuits that can be used for the specific application of quantum walks.

QOPS Framework

The QOPS framework is a well-suited direct encoding of projective simulation for single-photon excitation. However, for multi-excitation scenarios, due to multi-photon interference effects, the framework leads to some issues that we discuss in Section V and I propose a solution in Section VI.

The basic learning procedure for QOPS needs to be constructed in another way. The following steps are followed after the initialization of the circuit:

We keep in memory $\mathcal{P} = \{p\}$ of size m, the set of parameter tuning the transition between modes. At each epoch, we compute the action associated to the output state, retrieve the reward r given by the environment and start the learning process:

- Generate $dk = \{-1, 1\}^m$ from a uniform distribution.
- Compute $\mathcal{P}_\pm = \mathcal{P} \pm c * dk$ the 2 modified sets of parameters with c a constant.
- Run and retrieve the reward associated to $\mathcal{P}_+$ and $\mathcal{P}_-$ (r_+ and r_-).
- Calculate the loss associated with those changes of parameters l_+ and l_- ($l_\pm = \frac{1}{max(r_\pm, 0.0001)}$ and then the associated gradient

$$g = \left\{ \frac{l_+ - l_-}{2 * c * dk} \right\}$$

  for each parameter.
- Finally update the parameters by using $\mathcal{P}' = \mathcal{P} - n * g$ with $n = L_{earningRate} * e^{i(e_{poch} * d_{ecay})}$.

This new algorithm, while promising, does have its limitations. One drawback is that the procedure requires three runs for a single cycle or '*epoch*' of the learning process. This increases the computational resources required for the learning process. However, the potential benefits offered by quantum speedup could outweigh this limitation.

This process is iterative and requires careful calibration to ensure that the quantum system is moving towards an optimal state for learning and decision-making.

Finite Difference Stochastic Approximation

The *Finite Difference Stochastic Approximation* (**FDSA**) is a pivotal technique for optimization [43], particularly within the scope of stochastic problems. This method was introduced in [44]. The primary objective of FDSA is to compute the gradient of a loss function, denoted a $\frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \vec{\theta}}$ where $\mathcal{L}_{as}(\vec{\theta})$ is the loss function. This method is used to efficiently calculate a gradient for VQAs and consists into the approximation:

$$\frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \vec{\theta}} = \left(\frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \theta_0}, \frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \theta_1}, \dots \right) \rightarrow \frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \theta_k} \approx \frac{\mathcal{L}_{as}(\vec{\theta} + \varepsilon \vec{e}_k) - \mathcal{L}_{as}(\vec{\theta} - \varepsilon \vec{e}_k)}{2\varepsilon} \tag{1}$$

The method in question, facilitates the computation of the individual gain that can be attributed to the variation of a single parameter within the system. This is particularly valuable as it allows for a granular understanding of how each parameter influences the overall performance and behavior of the VQA. By isolating the effect of a single parameter, researchers and practitioners can make informed decisions about which parameters may require fine-tuning to optimize the algorithm's efficacy.

However, this approach is not without its limitations. As the dimensionality of the parameter space increases, the computational resources required to calculate these individual gains grow also. This is because the method necessitates the evaluation of the loss function at two distinct points for every single parameter, a process that becomes increasingly taxing as the number of parameters scales up. The computational overhead associated with this method can become a significant bottleneck, especially when dealing with complex systems that are characterized by a large number of parameters.

Simultaneous Perturbation Stochastic Approximation

The *Simultaneous Perturbation Stochastic Approximation* (**SPSA**) [43] represents a significant advancement when juxtaposed with its predecessor. The *Finite Difference Stochastic Approximation* (**FDSA**). The core philosophy of SPSA is predicated on the simultaneous perturbation of all parameters within the model, as opposed to the sequential approach adopted by FDSA. This methodology is encapsulated in the following mathematical expression:

$$\frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \vec{\theta}} = \left(\frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \theta_0}, \frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \theta_1}, \dots \right) \rightarrow \frac{\delta \mathcal{L}_{as}(\vec{\theta})}{\delta \theta_k} \approx \frac{\mathcal{L}_{as}(\vec{\theta} + \varepsilon \vec{\delta}_k) - \mathcal{L}_{as}(\vec{\theta} - \varepsilon \vec{\delta}_k)}{2\varepsilon \delta_k} \tag{2}$$

Here, the perturbation vector $\vec{\delta}$ is introduced to facilitate the approximation of the gradient of the cost function. This vector is composed of elements that are independently generated, adhering to a symmetrical distribution centered at zero. The symmetry of this distribution is essential as it ensures that the perturbations are unbiased, which is a critical aspect in maintaining the integrity and reliability of the optimization process.

The use of such a perturbation method involves a careful balance between the efficiency of the training process and the computational resources expended. Efficiency, in this context, refers to the ability of the training process to converge to an optimal set of parameters that minimize the cost function. On the other hand, computational cost is concerned with the amount of computational power and time required to perform the training. In scenarios where the parameter space is exceedingly large, it becomes advantageous to employ a greater number of training epochs. The trade-off is justified by the acceleration in the training process, which is achieved by leveraging the stochastic nature of the perturbation vector $\vec{\delta}$.

IV Prisoner's Dilemma

IV.1 Classical Prisoner's Dilemma

The prisoner's dilemma is a fundamental concept in game theory [5] that presents a scenario where two rational agents, often referred to as players, are faced with a decision-making problem. A game is formally constructed as a tuple $(\mathcal{P}, \mathcal{S}_p, \mathcal{R})$. $\mathcal{P}$ is the set of players, $\mathcal{S}_p$ are the sets of possible actions for each player and $\mathcal{R}$ is a payoff function $\mathcal{R} : \mathcal{P} \times \mathcal{S} \to r$ with r the reward given to the players. The dilemma is structured around two available strategies for each player: collaboration (denoted as strategy C) and defection (denoted as strategy D). In the classical version of this thought experiment, the two agents, Alice and Bob, engage in a game with a set of predefined rules that determine the outcomes based on their chosen strategies.

The rules of the game are as follows:

- If both Alice and Bob choose to collaborate, they each receive a reward of 3 coins.
- If both decide to defect, they each receive a lesser reward of 1 coin.
- If one collaborates while the other defects, the collaborator receives no coins, and the defector receives a higher reward of 5 coins.

These rules can be succinctly represented in a reward table, which provides a clear visualization of the possible outcomes for each combination of strategies:

Alice \ Bob	C	D
C	(3,3)	(0,5)
D	(5,0)	(1,1)

Analyzing this table, we can deduce that the most beneficial outcome for both agents collectively is achieved when they both opt to collaborate, resulting in a total distribution of 6 coins. This outcome is considered optimal as it maximizes the overall reward that is distributed between the two players. However, this optimal outcome is contingent upon a mutual trust between Alice and Bob, as any deviation by one party could lead to an imbalance in the distribution of rewards.

Upon further examination, we observe that regardless of Bob's strategy, Alice can maximize her individual reward by choosing to defect. This observation holds for Bob as well. This leads to the emergence of a second strategy, which prioritizes individual gain over collective benefit. We formally define some concepts from game theory [45], [46]:

- **Dominant Strategy**: A strategy s_A is said dominant for Alice if her reward $\mathcal{R}_A(s_A, s_B) \geq \mathcal{R}_A(s'_A, s_B), \forall s'_A \in S_A, \forall s_B \in S_B$ with S_A and S_B the sets of all possible strategies for Alice and Bob respectively.
  In the case of the prisonners dilemma, the strategy D is a dominant strategy for both Alice and Bob.

- **Pareto Optimal Strategy**: a pair of strategies (s_A, s_B) is a Pareto Optimal strategy if no player can increase its individual reward without decreasing the opponents.
  In the prisonners dilemma, the pair of strategies (C,C) is Pareto Optimal.

- **Nash Equilibrium**: a pair of strategies (s_A, s_B) is a Nash Equilibrium if $\mathcal{R}_A(s_A, s_B) \geq \mathcal{R}_A(s'_A, s_B)$ & $\mathcal{R}_B(s_A, s_B) \geq \mathcal{R}_B(s_A, s'_B), \forall s'_A \in S_A, \forall s'_B \in S_B$.
  For the prisonners dilemma, the pair of strategies (D,D) is a Nash Equilibrium.

- **Mixed strategies**: we call ς_A a mixed strategy if Alice plays different strategies in $S_A = \{s_{A1}, \ldots, s_{AN}\}$ with probabilities $\sigma_A = \{p_{A1}, \ldots, p_{AN}\}$ with $\sum_i p_{Ai} = 1$.
  In the context of the prisonners dilemma, if Alice plays C with probability 0.5 and D with probability 0.5, Alice plays a mixed strategy.

We can notice that we still can construct equilibria with mixed strategy if a pair of mixed strategy $(\varsigma_A, \varsigma_B)$ if $\mathcal{R}_A(\varsigma_A, \varsigma_B) \geq \mathcal{R}_A(\varsigma_A, \varsigma'_B)$ and $\mathcal{R}_B(\varsigma_A, \varsigma_B) \geq \mathcal{R}_B(\varsigma'_A, \varsigma_B)$.

IV.2 Quantum Prisoner's Dilemma

Quantum Games and application to PD

Quantum game theory is an intriguing extension of classical game theory that incorporates the principles of quantum mechanics. This advanced branch of study introduces quantum aspects such as superposed states, entangled states, and the superposition of strategies, which significantly expand the strategic possibilities available to players.

In quantum game theory, players have access to a broader range of strategies due to the phenomenon of superposition, where a quantum system can exist in multiple states simultaneously [47], [48]. This contrasts with classical game theory, where the strategies are typically discrete and well-defined. The concept of entanglement further complicates the strategic landscape. This interconnectedness means that the choice of strategy by one player can instantaneously affect the other player's outcomes. To illustrate how quantum game theory can be applied, let's consider implementing the prisoner's dilemma on quantum computers, such as those developed by Quandela. In a quantum version of the prisoner's dilemma, Alice and Bob would not be restricted to merely choosing between collaboration or defection. Instead, they could prepare a quantum strategy that is a superposition of collaborating and defecting, which is an equivalent to mixed strategies. The quantum strategies are represented by quantum bits, or qubits, which can be in a state of 0, 1, or any other quantum superposition of these states.

When the game is played on a quantum computer, the qubits corresponding to the players' strategies can be initially entangled, allowing for different strategies to appear. The players then apply quantum operations to their respective qubits, which can be thought of as choosing their strategies in the quantum realm. After the operations are applied, the qubits are measured, and the outcome of the game is determined based on the resulting state of the qubits.

The quantum prisoner's dilemma introduces new equilibria and potential outcomes that are not present in the classical version of the game [48], [49]. For example, there may exist a quantum strategy that allows both players to achieve better outcomes than the classical Nash equilibrium, potentially leading to a win-win situation that is unattainable with classical strategies.

Envisioning a quantum circuit with four modes and two photons as the input, we can delve into the intricacies of quantum strategies within the framework of game theory. The initial state of the system is denoted as $|1,0,1,0\rangle$, representing the presence of a photon in modes one and three. This state serves as the starting point for the strategic interplay between two players, Alice and Bob.

Alice employs a reflectivity tunable beam splitter between modes one and two to encode her strategic choice, while Bob does the same between modes three and four. In the quantum version of the game, the players choose their strategies by acting on the Hilbert space of the physical system. The beam splitter settings are adjustable, allowing Alice and Bob to manipulate the reflectivity and, consequently, their strategies within the game. The output state of the circuit corresponds to the strategic decisions made by the players.

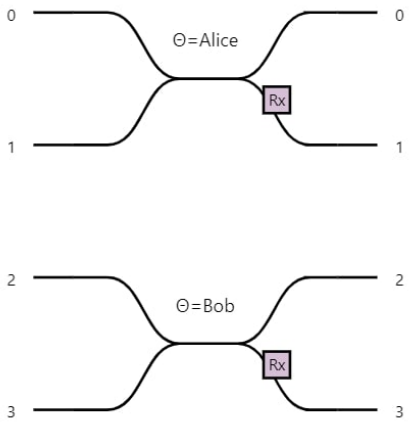


Figure 11: *Simple Prisoner's dilemma encoding. The player A (Alice) tunes the upper beam splitter by adjusting its reflectivity. She plays either $|0\rangle$ = Collaborating, $|1\rangle$ = Defecting or a superposition of those strategies. The player B (Bob) follows the same procedure.*

In this quantum game, the state $|1,0\rangle$ signifies that Alice opts for cooperation, whereas the state $|0,1\rangle$ indicates her defection. A similar encoding applies to Bob's choices, with the corresponding states for modes three and four. The quantum circuit, thus, becomes a physical representation of the players' decisions, with the output state encoding the combined strategies of Alice and Bob.

To gain insights about the output state, quantum state tomography can be performed. This process involves running the experiment multiple times, each iteration contributing to a statistical average that approximates the true quantum state. By analyzing the results of these iterations, one can deduce the reward outcomes associated with different strategic combinations.

The classical strategies are then transcribing like:

$$C \to BS(0) = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \;\&\; D \to BS(\pi) = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Computing the different output probabilities, considering general beam splitters where Alice and Bob respectively choose to play with θ_A and θ_B, we can give the following rewards for Alice and Bob:

$$\mathcal{R}_A(\theta_A,\theta_B) = 5\times\left|\sin\left(\frac{\theta_A}{2}\right)\cdot\cos\left(\frac{\theta_B}{2}\right)\right|^2 + 3\times\left|\cos\left(\frac{\theta_A}{2}\right)\cdot\cos\left(\frac{\theta_B}{2}\right)\right|^2 + 1\times\left|\sin\left(\frac{\theta_A}{2}\right)\cdot\sin\left(\frac{\theta_B}{2}\right)\right|^2$$

$$\mathcal{R}_B(\theta_A,\theta_B) = 5\times\left|\cos\left(\frac{\theta_A}{2}\right)\cdot\sin\left(\frac{\theta_B}{2}\right)\right|^2 + 3\times\left|\cos\left(\frac{\theta_A}{2}\right)\cdot\cos\left(\frac{\theta_B}{2}\right)\right|^2 + 1\times\left|\sin\left(\frac{\theta_A}{2}\right)\cdot\sin\left(\frac{\theta_B}{2}\right)\right|^2$$

Quantum Strategies

Building upon this foundational understanding, we can explore alternative encodings of the game to incorporate quantum phenomena more deeply. One innovative approach, introduced in [48], involves having an entangled state as input:

$$\frac{|1010\rangle + i\,|0101\rangle}{\sqrt{2}}$$

This state can be generated using a modified version of the KLM CNOT gate, which has a success probability of $\frac{1}{9}$ [50]. The post-processing of this state requires the presence of two photons in output mode four, a requirement that can be visualized in a circuit diagram for Bell pair generation in Figure 12.

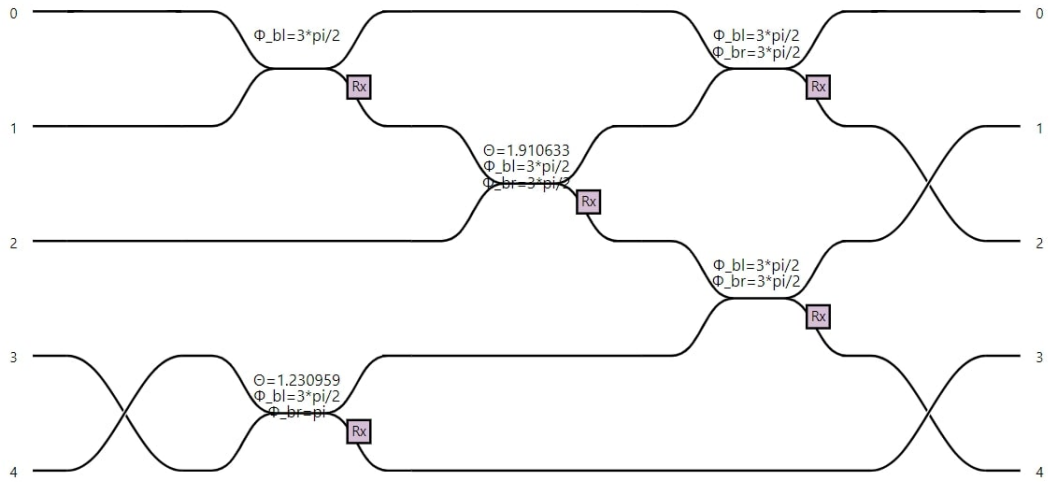


Figure 12: *Circuit for Bell Pair generation. The input state is* $|1,1,1,1,0\rangle$ *and gives the output* $\frac{|1,0,1,0\rangle - i|0,1,0,1\rangle}{\sqrt{2}}$ *conditional to the measurement of* $|2\rangle$ *in the last mode*

The encoding of strategies in this quantum context relies on the phase information carried by the states. To decode the players' actions after their moves, a decoding circuit is introduced (shown in Figure 13). This circuit is crucial for extracting the phase information from each substate of the output, allowing for determining the strategies employed by Alice and Bob.

strategy for A\ B	Output state
C,C	$\frac{\|1010\rangle + i\|0101\rangle}{\sqrt{2}}$
C,D	$\frac{\|1001\rangle - i\|0110\rangle}{\sqrt{2}}$
D,C	$\frac{\|0110\rangle - i\|1001\rangle}{\sqrt{2}}$
D,D	$\frac{\|0101\rangle + i\|1010\rangle}{\sqrt{2}}$

What is interesting in this situation is that the agent can act on the output by varying the phase of the state within the circuit. The unitary operation carried out by the beam splitter is now:

$$\hat{U}(\theta,\phi) = \begin{pmatrix} e^{i\phi}\cos\left(\frac{\theta}{2}\right) & \sin\left(\frac{\theta}{2}\right) \\ -\sin\left(\frac{\theta}{2}\right) & e^{i\phi}\cos\left(\frac{\theta}{2}\right) \end{pmatrix}$$

An intriguing aspect of this quantum game is the introduction of a new type of strategy, labeled as strategy Q. This strategy is associated with the unitary operation $\hat{U}(\theta = 0, \phi = \pi)$.

The strategy Q allows players to choose defection but also to influence the opponent's strategy by reversing it from cooperation to defection or vice versa. This adds a layer of complexity to the game, as players must consider not only their own strategies but also the potential impact on their opponent's choices.

This strategy Q represents a counter strategy to the strategy D as $\mathcal{R}_A(Q_A, D_B) = 5$ and $\mathcal{R}_B(Q_A, D_B) = 0$. We also notice that C is the counter strategy to Q and naturally D is the counter strategy to C. Thus there exist no dominant strategy and we expect the players to find an equilibrium for the rewards $(3, 3)$. The strategy (Q,Q) is now also Pareto Optimal.

After each player performs its move, the decoding circuit, shown in Figure 13, is employed to interpret the results. This decoding circuit corresponds to the complex conjugate of the initial entangling gate. The output states are then measured, providing the necessary information to calculate the rewards and facilitate a rapid learning process. However, this method introduces additional spatial modes, which may pose challenges regarding the quantum processing unit's (QPU) spatial capacity. This limitation becomes more pronounced in more complex scenarios, such as neural network-based prisoner's dilemmas, detailed in Section IV.3.

The design and optimization of the decoding circuit are the results of collaborative efforts, including contributions from previous interns.

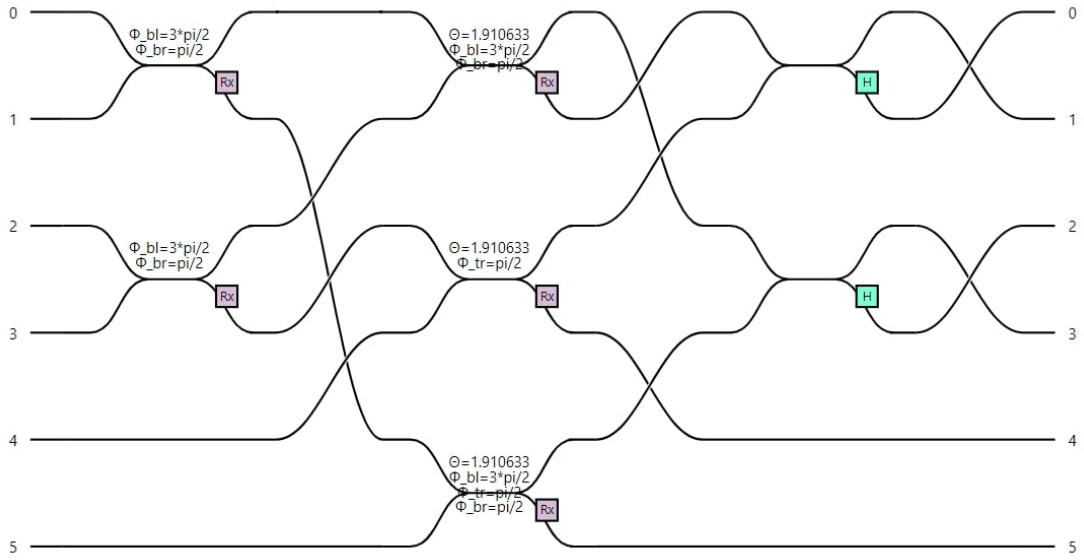


Figure 13: *Bell state decoding circuit. It permits to unentangle the states in order to retrieve similar results as the classical strategy. This unentangling is conditional to the measurement of 1 photon in the modes* $0+1$ *and 1 photon in the modes* $2+3$.

IV.3 Prisoner's Dilemma with Neural Network Induced Unitary

The quantum version of the game have introduced several changes in the concepts of the game and in the possibilities for each player, allowing them to tune the phases according to their will and thus have an influence on the opponents play.

One can notice that in the quantum case, every strategy $s = (\theta, \phi)$ comes with a counter strategy $\bar{s} = (\theta, \phi)$. To broaden the scope of available strategies, we can consider constructing a general unitary operator the is implemented in the circuit as an ECM, shown in Figure 14.

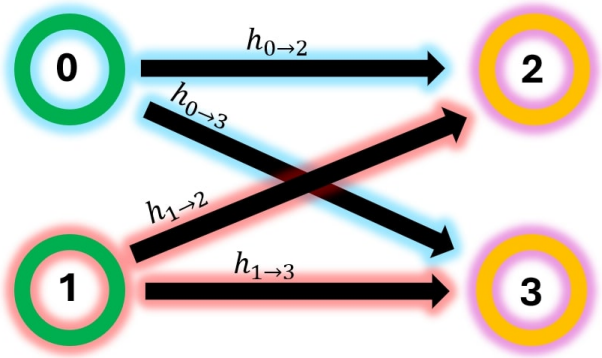


Figure 14: *ECM of a single player for the prisonners dilemma. In entry is a superposition of* $|0,1\rangle$ *and* $|1,0\rangle$*, corresponding to the part of the entangled state given to the player* $\frac{|1,0,1,0\rangle - i|0,1,0,1\rangle}{\sqrt{2}}$*. The transitions within the ECM are tuned by the player to determine its strategy.*

The unitary operation that is implemented in the circuit is of mathematical form

$$\hat{U}(a,b,c,d) = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$$

Using this general unitary, if Alice plays a pure strategy $s_A = (a,b,c,d)$ there always exists a counter strategy $s_B = (-ib, a, -d, -ic)$ such that Alice gets a reward of 0 and Bob gets 5. Alice should then rely on mixed strategies to find an equilibrium. In [51] is shown that with a general unitary, a mixed Nash Equilibrium is found for a reward of $(2.5, 2.5)$.

Introducing the ECM as an optical circuit comes with challenges. The procedure to find the exact circuit in order to implement the ECM is detailed in Section VI.

The circuit used to encode this ECM is show in Figure 15

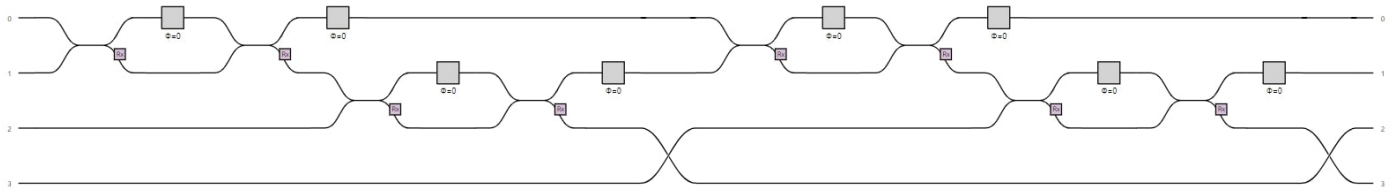


Figure 15: *Circuit corresponding to the ECM shown in Figure 14. The transitions are implemented with MZIs based on the method developed in Section VI.*

In this circuit, the state (input) clips are the 2 first modes (0 and 1) corresponding to the clips 0 and 1 and the action (output) clips are the 2 last spatial modes (2 and 3) corresponding to the clips 2 and 3.

At the end of the computation, we discard cases for which we measure an excitation in the input modes, meaning that there was no excitation propagation in the ECM.

IV.4 Equilibrium Scenarii in the Prisoner's Dilemma

Classical Version

In the classical version (Section IV.1) we expect a Nash Equilibrium to be found at respective reward (1-1), corresponding to the strategy where both players choose to defect.

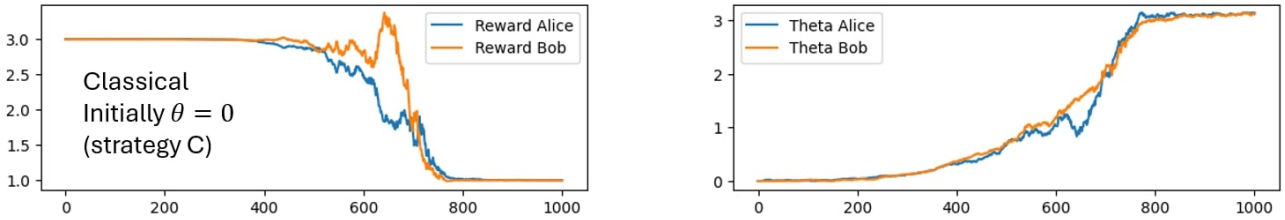


Figure 16: *Results showing the Nash equilibrium from learning agent in the classical strategy game. On the left, the reward evolves until reaching the distribution (1-1), which correspond to both players defecting. On the right side, the player parameters evolve until the beam splitters have changed from fully reflective to fully transmitive, corresponding to a shift for collaboration to defection.*

From Figure 16 we effectively get the expected Nash Equilibrium as the agent learns how to maximize its individual reward. The evolution of the parameters show a strategy change from the strategy C (collaborating) for which the reflectivity of the beam splitter is tuned such as the transformation is the identity ($\theta = 0$) to the strategy D (defecting) for which the beam splitter transformation is

$$\hat{BS}(\theta = \pi) \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$$

Quantum Version

In the quantum version of the game, we start with both player having the same strategy C. The first expected evolution is the player going for the strategy D as each player gains in defecting when the other player collaborate. This strategy dominates until whenever 1 player tunes the beam splitter is such a way that the mixed strategy favors D (D is chosen in more than 50% of the times).

At this point the dominant strategy for the second player is not anymore the strategy D but becomes a new strategy Q (defecting and reverse the opponent's play) such that the second player chooses the counter strategy of its opponent. At this point, the strategy Q becomes the optimal for both players and we expect a Nash equilibrium for the reward distribution 3-3 with both players having the strategy Q.

As expected, we show on Figure 17 the evolution of the mean reward and parameters over 1000 epochs. Initially both players collaborating, we retrieve what was expected, both players tuning to defect until one reaches almost $\theta = \frac{\pi}{2}, \phi = 0$ which is $|\phi\rangle = \frac{1}{\sqrt{2}}(|C\rangle + |D\rangle)$. Then we see the switch of strategy towards Q ($\theta = 0, \phi = \frac{\pi}{2}$).

The aim of this version of the game is to have a Nash equilibrium (no player can improve its individual reward by changing its individual play) that corresponds also the a maximized cumulative reward (here 6).

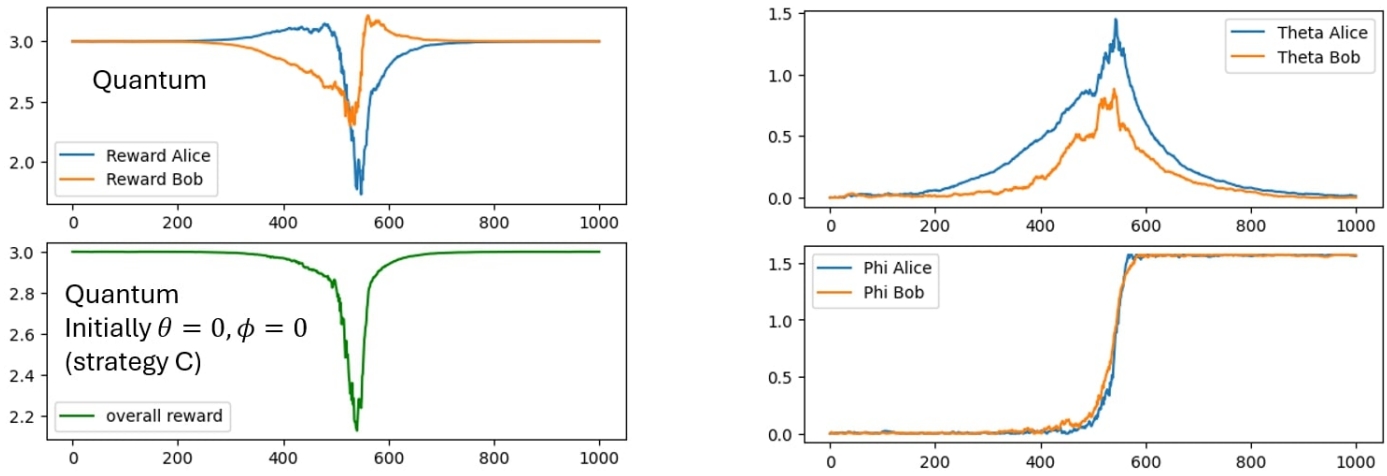


Figure 17: *Evolution of the mean reward of the players along with their parameters in the quantum strategy game. The evolution of reward and parameters suggests that the players shift their strategies in multiple steps $C \to D \to Q$. This shifting stops when the 2 players reach an equilibrium which is Pareto Optimal.*

General Unitary Version

From [51], we expect the ECM implementation to lead to a Nash equilibrium of reward (2.5-2.5).

Finding a reasoning that would correspond to what the players learn and how they adapt becomes much more complicated in this case because of the number of parameters that are trained (8 for each player) and the different possible strategies. Here we verify that the constructed circuit actually produces the expected Nash equilibrium.

Shown in Figure 18, we obtained the expected results from [51]. The introduction of the ECM as a way to tune every possible transition is interesting as a proof of concept that confirm the validity of using optical circuits to compute excitation walk over a DAG, here specifically an ECM. It also demonstrates the ability of the QOPS framework is expressive enough to approximate such states.

It also confirms the work for the construction of optical circuits to simulate graphs I made (see Section VI) as a simplification and improvement of the previously done implementation.

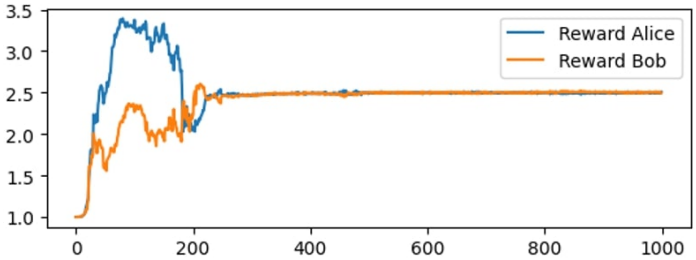


Figure 18: *Evolution of the individual rewards for the general unitary version of the game. The evolution does not permit to distinguish very defined strategies but the equilibrium suggests a Mixed Nash Equilibrium.*

V Multi-excitation Quantum Optical Projective Simulation

V.1 Classical Multi-Excitation Projective Simulation

Projective Simulation (PS) is a novel and significant advancement in the realm of computer science, in the sub-field of machine learning. PS is particularly noteworthy for its role in making the decision-making processes of AI systems more accessible and understandable to humans, which is a crucial step towards the development of *eXplainable Artificial Intelligence* (**XAI**) [52].

A generalization of PS is the *Multi-Excitation Projective Simulation* (**mePS**) technique. It is a sophisticated method that enables the simulation of complex cognitive processes comparable to human thought [3] while keeping the advantage of interpretability of PS. The mePS framework operates by facilitating the treatment of multiple signals or 'excitations' through a specially designed network. This network, or *hyper-graph*, acts as a conceptual model for multiple thoughts or ideas developing simultaneously. This property aims at mirroring the nature of human reflexion with decisions based on composite thoughts.

The conceptualization of the mePS methodology derive from the principles observed in quantum many-body physics [3]. This quantum inspiration is not simply a theoretical basis but serves as a practical guide to address the challenges posed by the complexity of treatment and predictions of hyper-graphs. This transformation has profound implications, not only in terms of theoretical interest but also in practical applications. It leads to significant reductions in the computational resources required, thereby making the process more efficient and sustainable.[3]

Given the great advancements made by mePS, the exploration of quantum models within this framework presents an exciting opportunity for researchers. The prospects of quantum models to further refine and enhance the capabilities of mePS are vast. This direction promises to reveal innovative insights and methodologies.

The synergy between quantum computing and AI, is of high interest for scientific exploration as is could facilitate the treatment of information during the learning process of an agent.

V.2 Multi-Excitation Use Case - Invasion Game

Invasion Game Scenario

The invasion game, introduced in [35], represents a strategic scenario used to test the capabilities of mePS. It is characterized by an environment state being described with a multitude of variables. The environment dispenses rewards to an agent depending on its actions. It then learns what action to take by treating the different variables and reward obtained in previous experiences. The game's foundation is straightforward: it involves two protagonists: an attacker and a defender. The players are positioned by the side of a wall, each one on a different side. The wall only posses a pair of doors.

At the commencement of each round, the attacker gives the defender a sequence of variables that are codifying the choice of door for the coming assault. The defender, functioning as the learning agent, is tasked with the challenge of deducing the attacker's choice of door based only on the variables announced. A graphical representation of the game is shown in Figure 19.

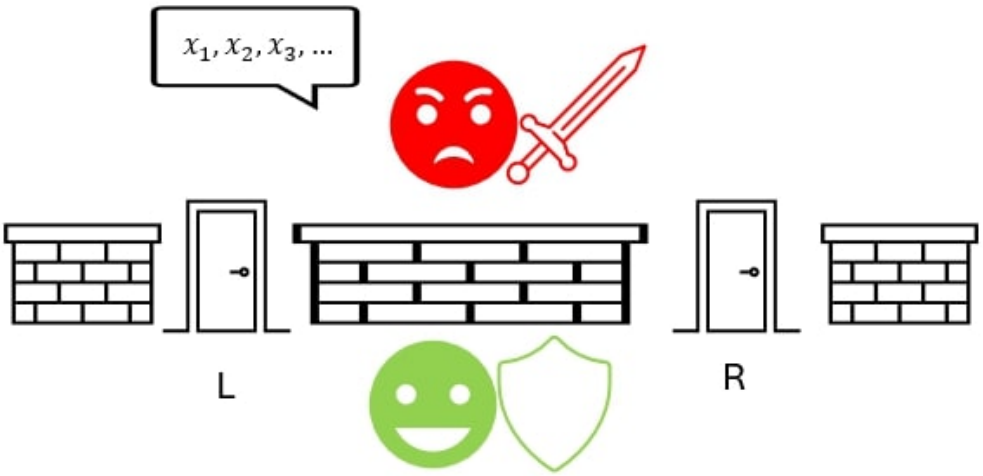


Figure 19: *Representation of the Invasion Game. In red the attacker disclose multiple variables encoding its attack. The green player chooses, based on those variables, which door to defend.*

The game's architecture permits the formulation of various rules. The rules govern the number of variables the attacker may announce, the potential values these variables can assume, and the methodology employed to codify the attack predicated on these variables.

One of the important challenges confronting the learning agent is the obligation to make decisions without the capability of relying on a memorization of values of variables. The attacker's strategy may depend on a complex set of variables, some of which could be deliberately misleading, serving as decoys to confound the defender.

Complexity Models of the Invasion Game

In the exploration of the Invasion Game, we distinguish three distinct variants. Each of the variant is designed to gradually increase the complexity and challenge presented to the learning agent. These variants are crucial for understanding the depth and adaptability required for an agent to learn effectively in dynamic environments. For the sake of clarity, we denote the left door as 0 and the right door as 1.

1. **Binary Variable Model:**
   · The first variant introduces the attacker announcing two variables, $\{x_1, x_2\}$, where each variable is binary.
   · The attacker's choice of door is determined by the expression $D_{oor} = (x_1 + x_2)\%2$.
   · This model represents the most fundamental level of the game, serving as an initial testing for our algorithms before progressing to more complex scenarios. It allows the learning agent to focus on the basic mechanisms of decision-making based on binary inputs.

2. **Binary Decoy Variable Model:**
   · Advancing to the second variant, the attacker discloses three binary variables, $\{x_1, x_2, x_3\}$.
   · The door selection still is dictated by the formula $D_{oor} = (x_1 + x_2)\%2$, which intentionally excludes x_3 from the decision process.
   · This variant adds a layer of complexity, as the learning agent must identify and disregard the decoy variable x_3, which serves no purpose other than to mislead. This tests the agent's ability to differentiate between relevant and irrelevant information, a skill that is predominant in more complex environments.

3. **Generalized Multi-Variable Model:**
   · The third and most sophisticated variant presents the attacker with a set of variables $\{x_1, \ldots, x_n\}$, where each variable can assume a range of values within the set $\{0, \ldots, m\}$.
   · The chosen door is ascertained by the equation $D_{oor} = \left(\sum_{k=1}^{p} x_k\right)\%2$.
   · This generalized model introduces a multitude of variable values and the potential for numerous deceptive variables, significantly elevating the game's complexity. The learning agent is now faced with the hard task of treating a larger set of variables, some of which may be designed to distract, while others are crucial to predict the attacker's behavior. This version challenges the agent's capacity for pattern recognition, strategic thinking, and the ability to learn from a complex array of inputs.

Each variant of the Invasion Game is meticulously constructed to simulate different levels of strategic complexity and to evaluate the learning agent's proficiency in decision-making. As the game evolves from the Binary Variable Model to the Generalized Multi-Variable Model, the learning agent must demonstrate an increasing level of 'cognitive sophistication'. The Invasion Game thus serves as an good microcosm for studying and developing advanced artificial intelligence capable of navigating in multifaceted strategic landscapes.

V.3 Optical Circuit Implementation of Multi-Photon Circuits

Challenges in Multi-Photon Circuit Design

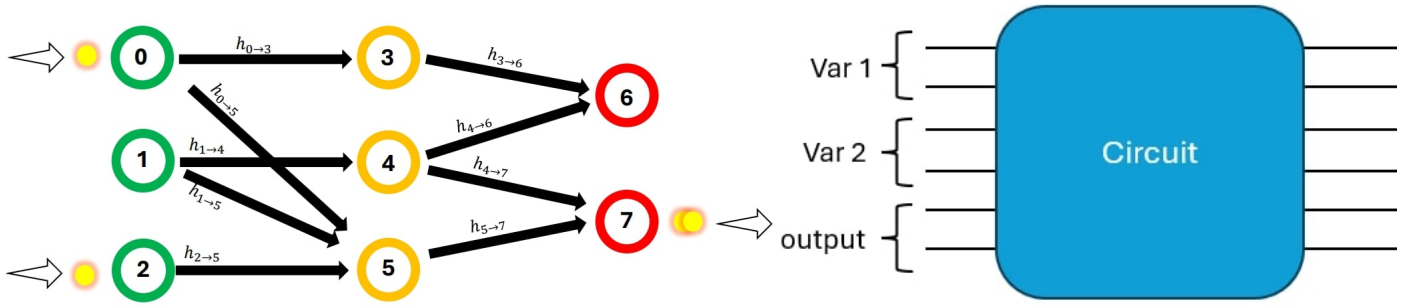
The design and implementation of multi-photon optical circuits present a myriad of challenges that significantly surpass those encountered in single-photon setups. The primary reason for this increased complexity lies in the exponential growth of possible output states as the number of photons involved in the process increases. Unlike the single-photon case, where the output is binary and thus relatively straightforward to encode, multi-photon systems can exist in a superposition of multiple states, leading to a combinatorial increase of potential outcomes.

The encoding of the decision process in multi-photon circuits is a task of high interest. It necessitates the development of novel encoding schemes of circuits that can effectively treat the high-dimensional quantum states. These schemes must not only accommodate the multitude of possible output states but also ensure that the encoded information is resilient to quantum noise and decoherence, which are important challenges in quantum systems.

Naive Implementation

The foundational concept for the initial implementation of meQOPS was to elaborate a process based on the established procedures of the single photon case. This approach was predicated on the naive assumption that the principles governing single photon dynamics could be extrapolated to a multi-photon context.

At the core of this implementation was the idea of leveraging an Episodic Compositionnal Memory (ECM) to facilitate decision-making processes within the system. The ECM was designed to interpret the distribution of excitations across two distinct outputs. If all excitations were detected in output 1, the system's agent would be instructed to defend the left door, while detection in output 2 would signal the defense of the right door. This binary decision-making protocol is depicted in Figure 20b, which illustrates the basic idea for meQOPS implementation.



(a) *Arbitrary example of expected graph* **(b)** *Corresponding circuit for a basic meQOPS implementation*

Figure 20: *Basic encoding of meQOPS showing the translation between a graph and an optical circuit in the naive case.*

To further clarify this concept, a graphical representation was provided in Figure 20a. This figure aims at illustrating the expected outcomes and the operational efficacy of the ECM, predicated on the distribution of excitations across the outputs.

Multi-Photons Interferences - Limitations of the Naive Implementation

Despite the promising nature of this approach, it became apparent that the encoding strategy had several strong limitations that required a re-evaluation of the design of the system.

The first limitation was the encoding's failure to consider certain output states, such as

$$|0,0,0,0,1,1\rangle$$

which represents one excitation in output mode 1 and another in output mode 2. This oversight exposed a deficiency in the encoding scheme, underscoring the need for a more comprehensive and inclusive approach capable of encapsulating the entire range of possible output states.

One idea is to consider only the desired output states, which are

$$|0,0,0,0,2,0\rangle \text{ and } |0,0,0,0,0,2\rangle$$

It is important to note that the actual computer used by Quandela does not support photon number resolution, such that those states are read the same as $|0,0,0,0,1,0\rangle$ and $|0,0,0,0,0,1\rangle$. Additionally to the complexity of this process, necessitating to discard a broad range of states, it also does not make a difference between states like $|0,0,0,0,2,0\rangle$ and a lossy state $|0,0,0,0,1,0\rangle$ for which we don't know if the second photon was actually going to end up in the same state or not.

Other solutions exists will be described during the construction of the final circuit.

A more significant challenge was posed by the *Hong-Ou-Mandel* (**HOM**) effect. The system's dependence on Mach-Zender interferometers to modulate transitions between spatial modes posses issues given the unpredictable nature of the HOM effect [53]. This quantum phenomenon, crucial for aggregating excitations into a single output mode, is non-deterministic. As depicted in Figure 21, the HOM effect does not allow for a tuning in the transitions between modes, thus eliminating any certainty in the directionality of excitations.

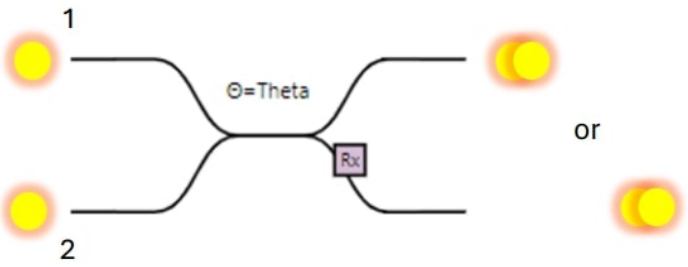


Figure 21: *Representation of the Hong-Ou-Mandel effect*

Figure 22 shows the expected behavior of the HOM effect within the basic ECM encoding framework. This visualization serves to highlight the problematic behavior of the researched effect and its implications for our computational strategy.

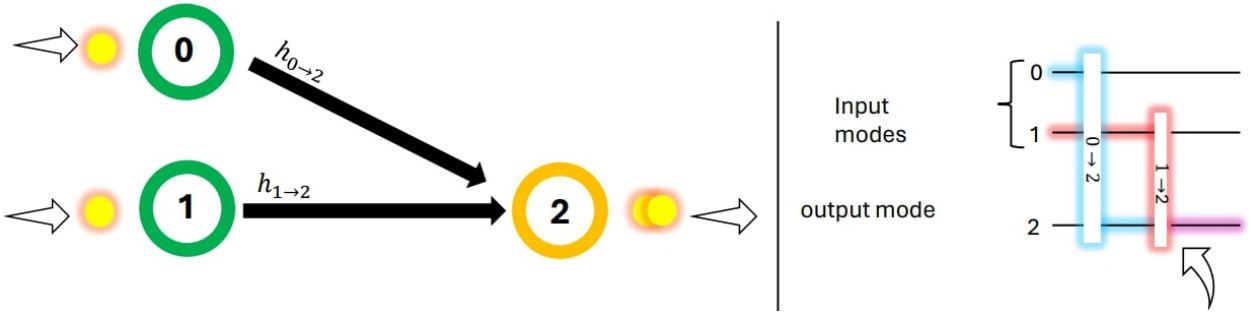


Figure 22: *expected HOM from basic ECM encoding. On the left side, an example or directed transition from 2 excitations in 2 modes to 1 single output mode. On the right side the corresponding naive circuit. The second transition on the right side shows the issue presented by the naive implementation.*

On the right-hand side of the figure is presented the actual MZI links that are added in the circuit in addition with the excitation propagation. The second transition is aimed at getting the photon present in the input mode 1 and adding it to the already excited mode 2. The HOM effect issue occurs as is cannot be controlled whether the 2 photons will end up in the mode 1 or the mode 2.

Given these limitations, it is essential to explore an alternative computational methodology. This new approach must overcome the shortcomings of the initial implementation, particularly in regard to the comprehensive encoding of output states and the manageability of photon behavior in the presence of the HOM effect.

V.4 Qudits Parallel Circuit Processing (QPCP)

The interest in using a quantum computer for such calculation lies in the speedup achieved by using a quantum system able to perform a quantum walk within an optical circuit instead of simulating the random walks of excitation in a graph.

In qudit parallel computing, the term qudit refers to a specific information encoding used for optical quantum computer. Instead of encoding the information on a basis of 2 states, namely $\{|0\rangle, |1\rangle\}$, we encode the information by a quantum unit of higher dimension. We talk about a quantum d-ary unit for a unit of information with a Hilbert space of size d [54]. Thus the information is encoded in the states $\{|0\rangle, |1\rangle, \ldots, |N\rangle\}$ It can be seen as an generalized quantum equivalent of the classical hexadecimal encoding.

For out case, it allows for a easier encoding of the input state, using the attackers choice of variable value directly as an input mode. This also permits to have an easier readability of the results as the photons are going to be broadly spread out in the circuit.

The initial idea was to be able to differentiate the photons such that the non-interaction between them allows for an easier manipulation and permits to actually transmit the photons in the output modes. If we take the example given in Figure 22, even if we do not suffer from the HOM effect, adding a MZI as link for modes 1 and 2 permits the photon already in mode 2 to propagate in the other way to mode 1. It could be interesting to do this manipulation by playing with the polarization of photons in order to be able to manipulate only one independently of the other but it would be

restricted to a very small number of photons, with a lot of optical components to tune and that are not implemented in Quandela's QPU.

Research has shown that some degree of control can be achieved using the principle of partial distinguishability of photons, as referenced in [55]. This approach has led to an improvement of 7% in the guidance of photons, a significant increase when compared to the random case of fully indistinguishable photons. However, it's important to note that this effect has only been demonstrated for a minimum number of photons of 7. There is no evidence to suggest that this improvement holds true for a lower number of photons.

Despite these findings, the limitations of this partial distinguishability bunching technique are significant. In particular, it is not applicable to the case of meQOPS.

Instead, it is possible to play differently with the differentiability between photons by using quantum parallel computing.

The idea that is behind the quantum parallel computing is to use a single big circuit divided into several different sub-circuits in order to perform the quantum walk of each excitation. Using this method, each photon is going to propagate within its own associated sub-circuit as it would in a single excitation QOPS. The decision process it decided depending on the multiple outcomes of those different circuits.

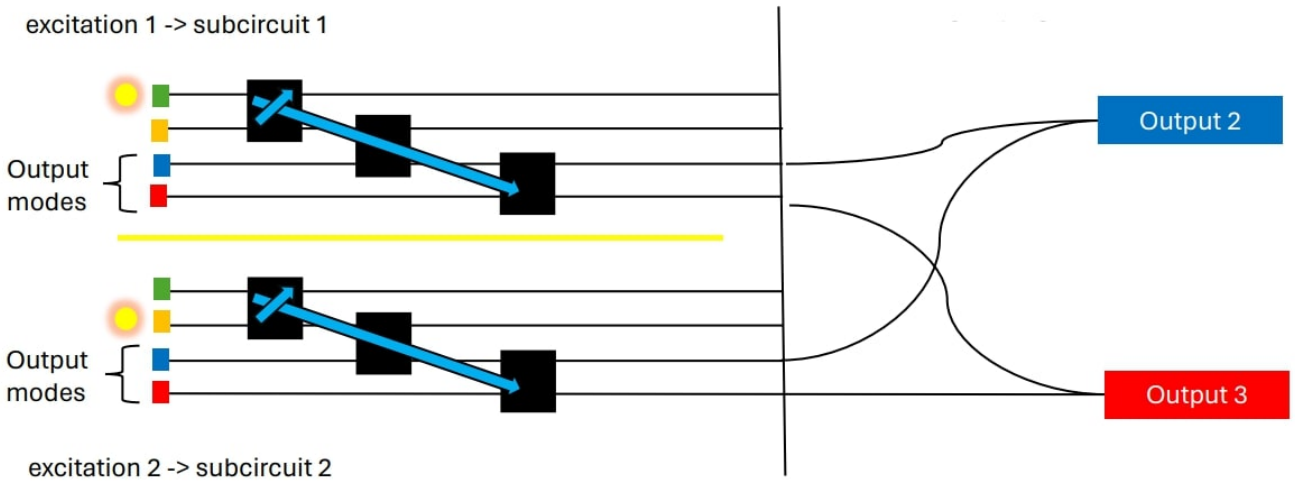


Figure 23: *Example of parallel meQOPS. The circuit is divided into smaller sub-circuits that each serves to the propagation of a unique single-photon. On the right side is represented the classical processing of the results that adds up the number of photon ending in the same hyper-mode.*

In Figure 23 is shown an example of parallel quantum computation. The example shown here actually corresponds to the exact circuit used for the Binary Variable Model. Lets decompose it to understand better the logic behind it.

The attacker announce 2 binary variables. In this examples, those variables take the values $x_1 = 0$ and $x_2 = 1$.

The circuit shown in Figure 23 first is made out of 2 sub-circuits, one for each variable. In each of the sub-circuits, we distinguish 2 input modes, corresponding to the variable being either 0 or 1, and 2 output modes, corresponding to the possible actions to take in the end (right door = mode 2, left door = mode 3).

The photons are placed in the input modes depending on the attackers variables. The agents agents goal is then to train each MZI, represented as black squares such that the excitation can propagate within the whole circuit to reach any mode of interest.

On the right-hand side, the output boxes indicate that we sum the corresponding modes such that we can reconstruct at the end a sort of photon number resolution. Even if the scheme shows those output boxes for the modes 2 and 3, they are actually performed for every modes of the sub-circuits.

The photon number resolution is achieved by summing the photon output of the considered modes. As this summing is classically done after retrieving the output state, one can notice that it is possible to compute independently the results for each sub-circuit one by one and classically reconstruct the state.

This method has several advantages and disadvantages. On the positive side, it allows for using single photon interference to get a potential quantum advantage without being perturbed by multi photon interference and thus conserves it's readability for Multi-excitation computing. The second big advantage is that it offers some kind of photon number resolution as the output 2 and output 3 results of the classical treatment of the general output state obtained.

On the negative side, this method becomes rapidly very expensive computationally when going for bigger instances. As every sub-circuit needs to be optimized for each excitation, the training time becomes very large for a higher number of excitations.

Finally, the main negative effect is that we do not take advantage of the interaction of the multiple photon as we are trying at maximum to avoid it. As a result, we do not achieve the initial claimed advantage by [3]. It would be interesting to investigate other ways to perform the calculation by conserving this sensible interactions.

V.5 Performances of QPCP over the Invasion Game Complexity Models

In order to accomplish the analysis of this framework, we took the approach of implementing 3 different circuits. Those circuits are not arbitrary but rather correspond to the 3 models, namely Binary Variable, Binary Decay Variable and Generalized Multi-Variable, we discussed in the previous sections.

For the implementation of the Generalized Multi-Variable Model, we decided to use three variables and three possible values for each of them. This decision was not made lightly, but rather, it was based on careful consideration of the model's characteristics and the specific requirements of our study. By using more than three variables, the model becomes computationally very heavy and hard to train in reasonable time with a classical computer. Already using three variables and three values, it becomes impossible to implement the circuit in the quantum computer or noisy simulator as it becomes way too big.

For each of these models, we use a specific process to analyze and understand them. This process involves performing a computation with a particular tool known as Perceval's Strong Linear Optics Simulator, often abbreviated as SLOS.

SLOS is a powerful tool that has been specifically designed for this type of computation. Two of the most significant features of SLOS are its ability to produce noiseless results and the ability to perform local computations. Those are critical features because it allows us to obtain clear, unambiguous data from our computations, allowing to formulate hypothesis on the capabilities of the computation. However it cannot be used for the analysis of quantum advantages and for the resistance of the program to noise.

In addition to its noiseless operation, SLOS also has the capability of implementing post-selection in the computation. Post-selection is a technique used in quantum computing to select only those results that meet certain criteria. In our case, we use post-selection to ensure that we are only considering the most relevant and useful data from our computations.

To give a practical example of how this works, let's consider a situation where we ask SLOS for 100 samples. In a typical computation, this might mean that we receive 100 samples, some of which meet our criteria and some of which do not. However, with post-selection implemented, SLOS ensures that all 100 samples we receive meet our criteria. In other words, we get 100 samples *after* post-selection.

In addition to the computations we perform using Perceval's *Strong Linear Optics Simulator* (**SLOS**), we also utilize another tool in our study. This tool is a noisy simulator, which has been specifically designed to reproduce the performances of Quandela's Quantum Processing Unit (QPU). It introduces loss of photons, variations in the phases and uncertainties in the calibration of the optical components. The use of a noisy simulator is a crucial aspect of my work. This is because real-world quantum systems, such as Quandela's QPU, are not perfect. They are subject to various forms of noise, which can introduce errors into the computations they perform. By using a noisy simulator, we can model these errors and gain a better understanding of how they might affect our results.

The noisy simulator we use has been carefully calibrated to match the performance of Quandela's QPU as closely as possible. This means that the results we obtain from the simulator should be very similar to those we would obtain if we were to perform the same computations on the actual QPU.

Because of this close match, we consider the simulator to be 'close enough' to the QPU for our purposes. In other words, we believe that the results we obtain from the simulator are sufficiently accurate for us to use them in our comparisons with the noiseless outputs from SLOS.

Performing these comparisons is a critical part of my work. By comparing the noiseless outputs from SLOS with the noisy outputs from the simulator, we can gain valuable insights into the effects of noise on our computations. This, in turn, can help to develop strategies for mitigating these effects and improving the accuracy of our results.

In our computational model, we employ a specific rule to calculate the reward. This rule is not arbitrary but has been carefully designed to reflect the success or failure of each run in an epoch. Here's how it works:

Firstly, it's important to understand that our computations are organized into epochs. An epoch is a complete cycle of computation, and for each epoch, we perform a substantial number of experiments. Specifically, we perform 10^5 experiments for each epoch. This large number of runs ensures that we have a robust sample size, which in turn increases the reliability of our results.

Now, let's move on to the actual computation of the reward. For each run in an epoch, we retrieve an output state. This output state corresponds to a door, either the correct door or the wrong door. If the output state corresponds to the correct door, we award 1 point. This is a positive reward, reflecting the fact that the run has been successful.

On the other hand, if the output state leads to the wrong door, we give -1 point. This is a negative reward, indicating that the run has been unsuccessful. By giving a negative reward for unsuccessful runs, we ensure that our reward system accurately reflects the success or failure of each run.

Once we have assigned points to each run in the epoch, we then compute the total reward. The total reward is not simply the total number of points. Instead, it is given by the average over all the points given in the epoch. This means that we add up all the points, and then divide by the total number of runs (10^5). This gives us an average reward value for the entire epoch.

There is one special case that we need to consider. Sometimes, a run may generate an output state that does not correspond to a valid action. In other words, the output state does not lead to either the correct door or the wrong door. In such cases, we assign a reward of -2. This is a more severe penalty, reflecting the fact that the run has not just been unsuccessful, but has failed to produce a valid result, leading to no decision made.

The rule we have established for our computational model allows us to draw several insightful observations. These observations are not just incidental findings, but they provide us with a deeper understanding of the dynamics at play in our model. Let's delve into these observations one by one :

Firstly, we notice that the learning process in our model is highly sensitive to the reward system. Specifically, when the total reward is negative, the learning process triggers drastic changes in the parameters. This is a significant observation because it shows that the model is actively learning and adapting based on the outcomes of each run. The negative reward essentially signifies that the wrong door has been chosen. When this happens, the model responds by making substantial adjustments to its parameters. This is a clear indication of its capacity for self-correction and improvement.

Secondly, we must acknowledge the challenges associated with running computations on the QPU. Due to frequent maintenance operations and high demand, obtaining runs on the QPU can be quite difficult. This is an operational challenge that we must contend with. However, it's important to note that these challenges do not detract from the validity or reliability of our model. They simply represent logistical hurdles that we must navigate in the course of our work.

Binary Variable Model

The Binary Variable Model, as the name suggests, is a computational model that operates on binary variables. This model is implemented using two variables and two values, which are provided by the attacker. The choice of door to attack is determined by the attacker using a specific rule. This rule is expressed mathematically as:

$$(x_1 + x_2)\%2$$

The corresponding circuit for this model is a complex structure that utilizes 16 parameters. These parameters are the ones to be calibrated to ensure the optimal functionality of the model. The circuit is visually represented in Figure 24, which provides a clear and detailed view of its structure and components.

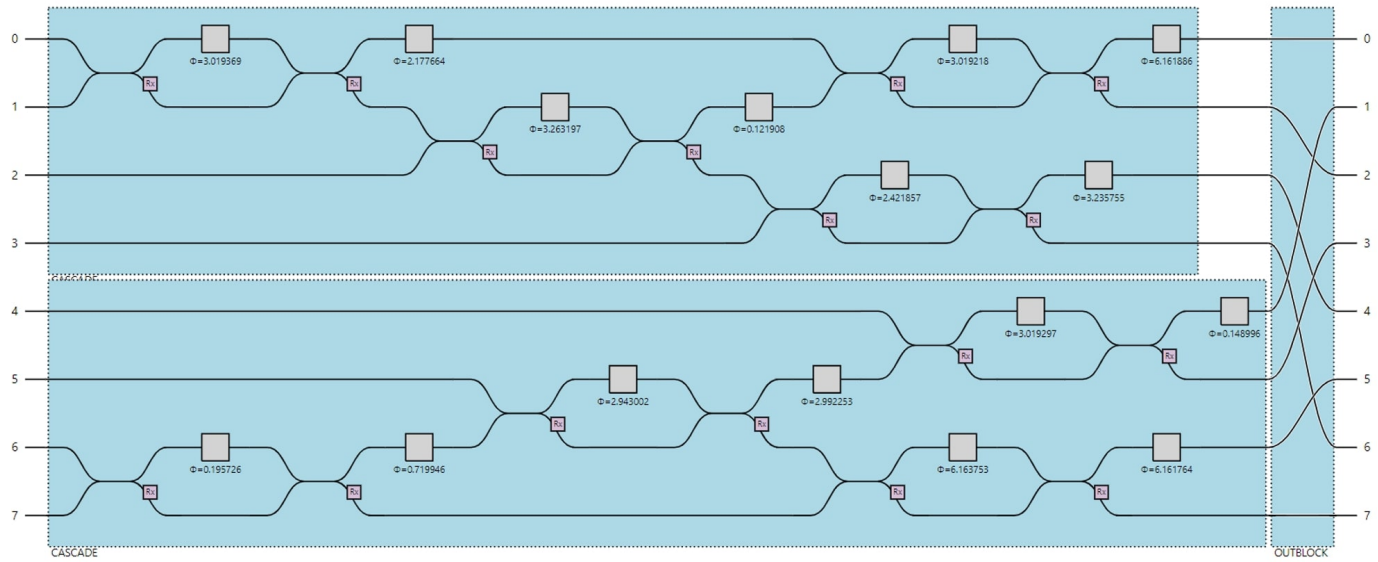
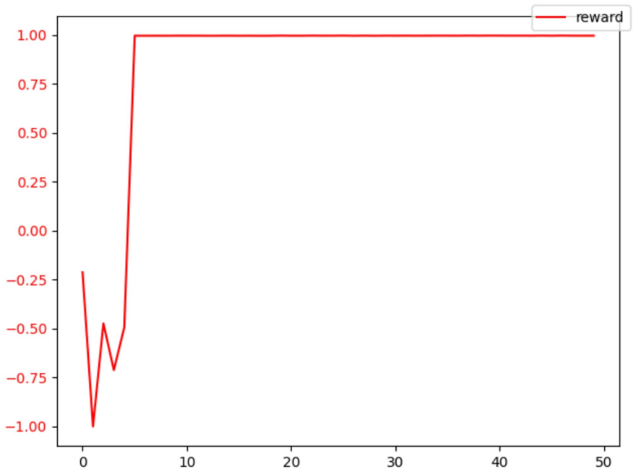


Figure 24: *Circuit used for the Binary Variable Model computed with QPCP*

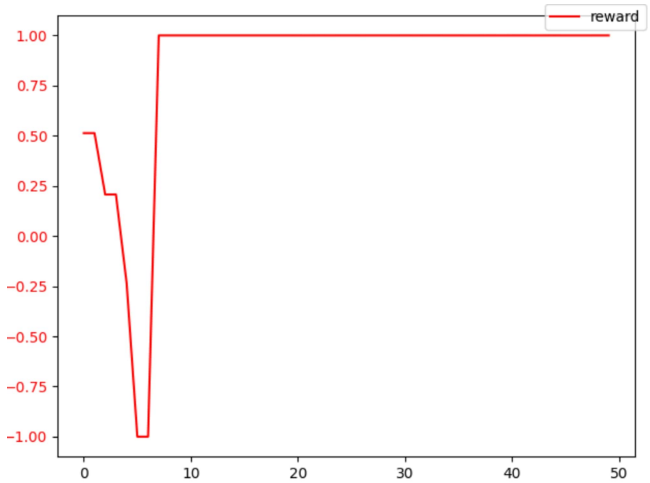
To evaluate the performance and effectiveness of the Binary Variable Model, we conducted experiments using both the SLOS and the noisy simulator. The results of those experiments are shown in Figure 25.

The results of these experiments were quite revealing. We found that the runs on both the SLOS and the noisy simulator produced very similar results. This similarity in results indicates that the model is robust and performs consistently under different noise conditions.

Furthermore, the results showed that the agent is able to learn effectively in this simple model. Despite the noise introduced in the simulator, the agent was able to achieve an ideal parameter configuration after approximately 10 epochs. This demonstrates the model's resilience and its ability to adapt and optimize its parameters in response to the learning process.



(a) *SLOS results showing a fast learning* ≈ 10 *epochs*



(b) *noisy simulator results also showing fast learning within the same range of* ≈ 10 *epochs as SLOS*

Figure 25: *results retrieved from the experiments performed for the Binary Variable Model*

Binary Decoy Variable Model

For this model, we are using 3 variables and 2 values given by the attacker for the implementation of this model. The choice of door to attack is still determined by the attacker using the rule :

$$(x_1 + x_2)\%2$$

The corresponding circuit is using 44 parameters and is represented in Figure 26

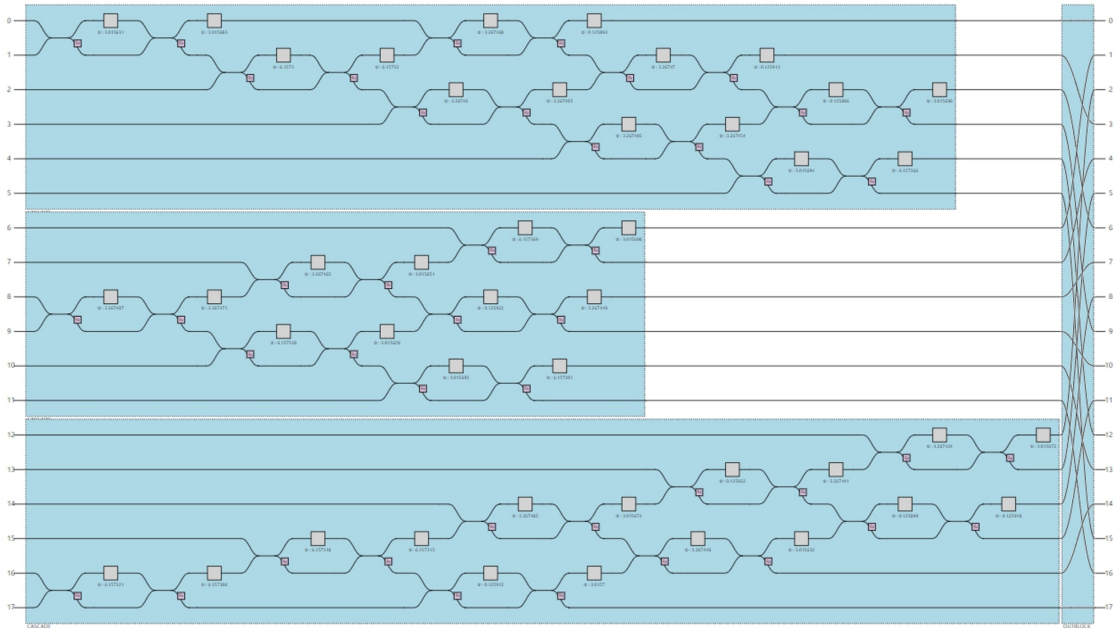
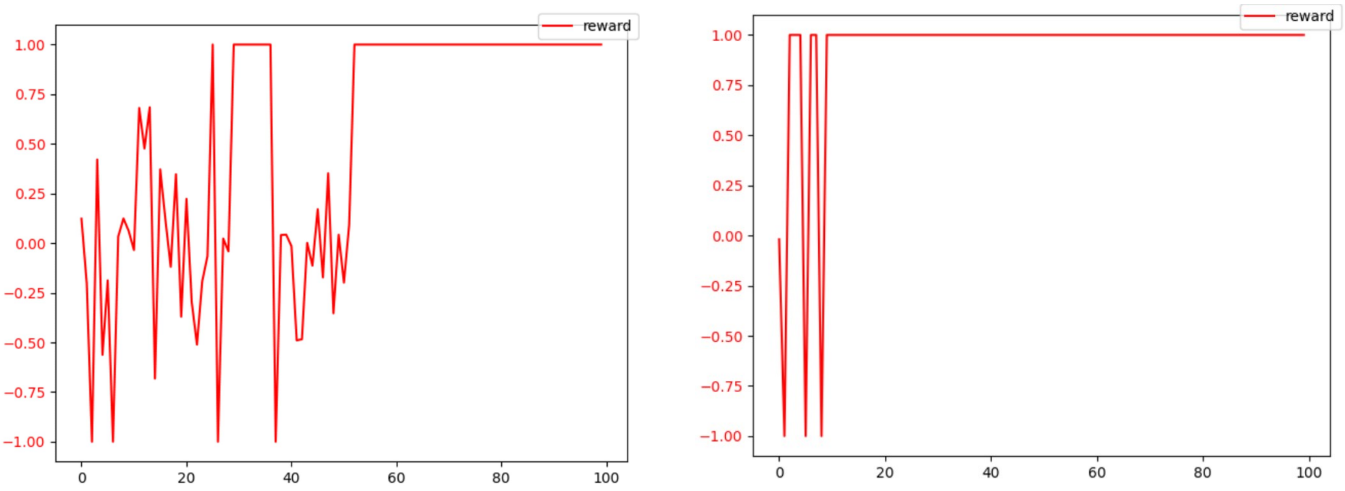


Figure 26: *Circuit for the Binary Decoy Variable Model computed with QPCP*

After implementing this circuit on both SLOS and noisy simulator we get the results shown in Figure 27.

To understand those results, it is important to understand that during the optimization of the circuit, the simultaneous perturbation stochastic approximation acts by randomly shifting the parameters in one way or another with respect to the other parameters. The direct consequence is that the tuning of the parameter highly depends on a random process that can possibly change the training time. So we cannot conclude here that the noisy simulator trains the parameters more efficiently in the general case. However, the lossy output distribution has the tendency to lower the reward, because of biases introduced in the results, inducing more drastic changes in the parameters. This bias might be a partial explanation for the faster moving reward curve. In any case, we show here that the procedure that was implemented for the Binary decoy variable model produces good results and thus the learning occurs.

The range of epoch for the program to learn is still pretty similar, getting from ≈ 10 to ≈ 60.



(a) *SLOS results showing a learning process finallized after about* ≈ 50 *epochs*

(b) *noisy simulator results with high performance, learning in* ≈ 10 *epochs.*

Figure 27: *Results retrieved from the runs performed for the Binary Decoy Variable Model. Those results need to be analysed looking the the functioning of the algorithm to explain this impressive performance.*

Generalized Multi-Variable Model

The Generalized Multi-Variable Model is a more complex and comprehensive model that expands upon the Binary Variable Model. This model is implemented using three variables and three values, which are provided by the attacker. The choice of door to attack is still determined by the attacker using a specific rule not employing all the variables, often referred to as the Decoy rule. This rule is expressed mathematically as $(x_1 + x_2)\%2$

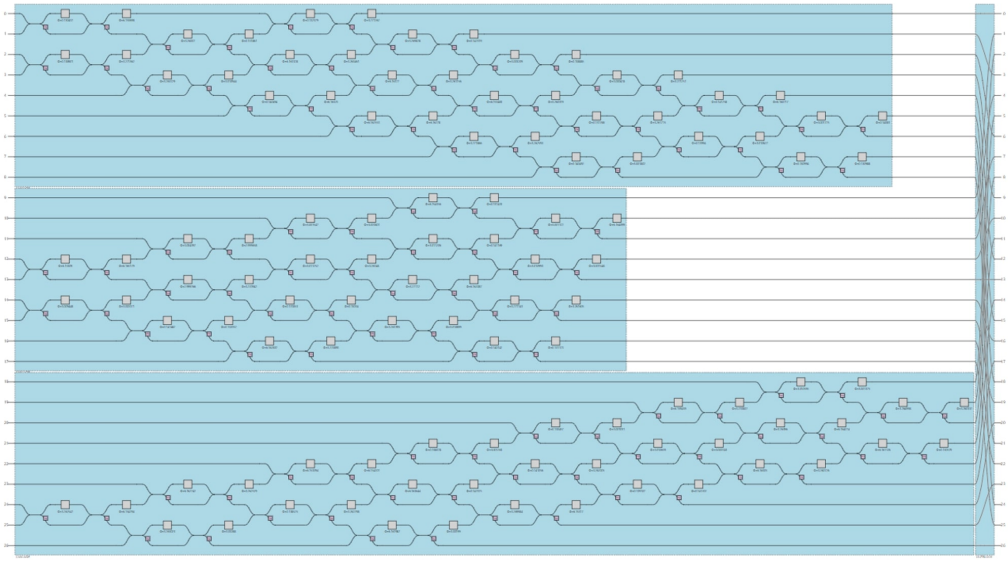


Figure 28: *Circuit for the Generalized Multi-Variable Model computed with QPCP*

The corresponding circuit for this model is a complex structure that utilizes 44 parameters. The circuit is visually represented in Figure 28, which provides a clear and detailed view of its structure and components.

To evaluate the performance and effectiveness of the Generalized Multi-Variable Model, we conducted experiments using the SLOS.

Unfortunately, we were unable to achieve results with the simulator due to the high number of optical modes required (26). Quandela's QPU and noisy simulator are limited to 20 optical modes, which posed a significant challenge for our experiments. The results are presented in Figure 29.

Despite this limitation, we were still able to get the agent to learn with this framework, with the three variables-three values with a learning time range of ≈ 70 epochs. This was demonstrated by the successful learning procedure of an agent using multiple excitations during the perfect runs (SLOS). This shows that even with the limitations of the simulator, the Generalized Multi-Variable Model is capable of effective learning and adaptation.

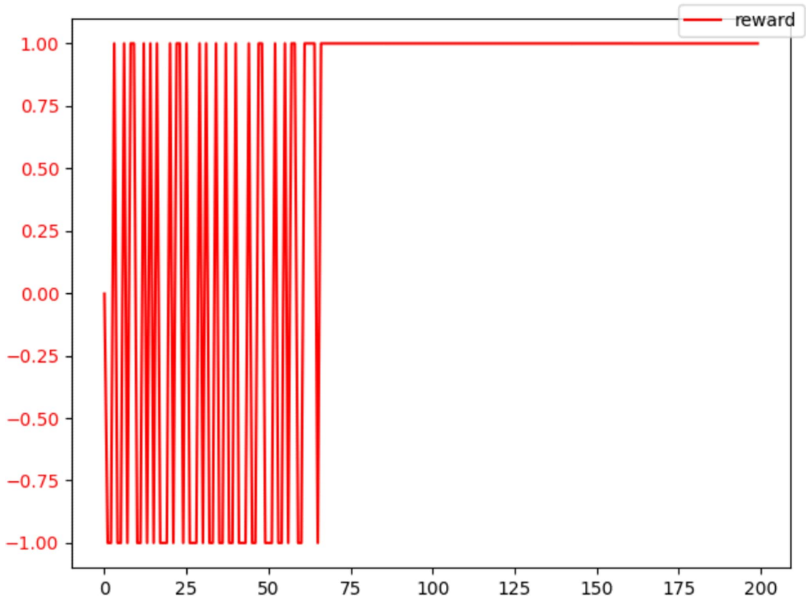


Figure 29: *SLOS results for the three variables - three values Generalized Multi-Variable Model. The model learns the right set of parameters in about ≈ 70 epochs.*

Improvements for parallel circuit computing

One main problem with this implementation is that the result rely on the classical information processing after getting the results in order to reconstruct a state. This dependence on classical methods can potentially limit the efficiency and capabilities of the system.

One could think of a solution using the logic introduced by the prisoner's dilemma problem with entangled photons. Coming with its set of challenges, we can look at the 2 agents from the prisoner's dilemma as one single agent trying to maximize the reward given a new payoff function tuned for this purpose.

For a 2 variable play with 2 possible values, one can imagine using the same input state

$$\frac{|1010\rangle - i\,|0101\rangle}{\sqrt{2}}$$

applying a specific operation corresponding to the the variable announced and finally add the ECMs of the agent. Further research will be pursued to implement and test this method, unfortunately not finished by the submission date of this thesis.

VI Optical Circuit Encoding of Graphs

VI.1 Naive Encoding of graphs

In the scope of mathematical structures, a graph is an object composed of *vertices* (or nodes/clips) and *edges*. Those edges can sometimes be associated with weights and directions. Mathematically an *undirected* graph is denoted as $G = (V, E, H)$ where V represents the set of vertices and E denotes the set of edges [56]. Each edge is associated to 2 vertices known as its *endpoints*. In the case of an *undirected* graph, edges are an unordered pair$\{v_1, v_2\}$ of vertices. Directed graphs, on the other hand, maintains the order of the pair of vertex, which is crucial for the representation. Each element of E can be written as $\{v_1 \to v_2, h_{1\to 2}\}$, where h is the weight associated with the edge.

For this particular study, our interest lies in *Directed Acyclic Graphs* (**DAGs**), which are a specific category of directed graphs. DAGs are restricted to graphs that do not contain any loops. In the specific context of *Episodic Compositionnal Memory* (**ECMs**), we also take into account the weights associated with the edges.

The fundamental idea to translate a DAG into optical circuits is to associate one spatial mode per vertex. This association implies the translation between edges and links between two spatial modes, for which we use *Mach-Zehnder Interferometers* (**MZIs**), as illustrated in Figure 30.

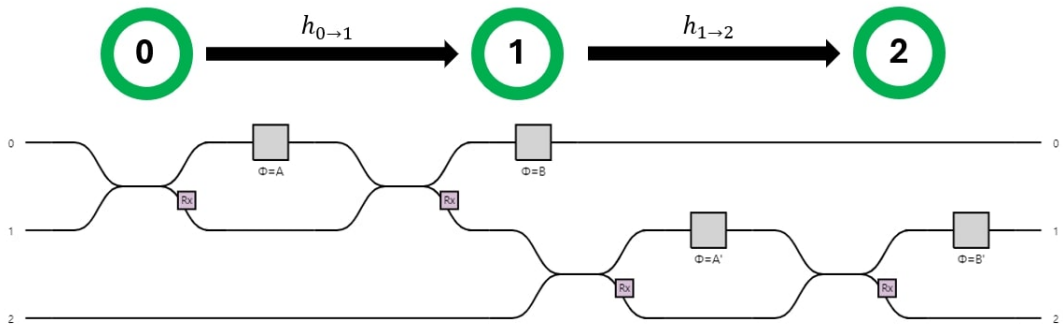


Figure 30: *Example of translation from a simple DAG into an optical circuit (above : graph, bellow : corresponding circuit)*

Back-Propagation of Excitation

As discussed in Section IV.3, we want to be able to link several input clips to a single output clip. This permitting the excitation to propagate in a manner similar to a neural network. However, this approach leads to certain complications when introduced as an optical circuit.

In Figure 31 is illustrated an example of backpropagation. The underlying idea is to naively connect each state clip to output clips using MZIs. As the transitions are applied one by one, if 2 input clips (for example clips 0 and 1 in Figure 31) are connected to a single action clip (lets say clip 2), the clip 0 links with the clip 2, allowing for a potential propagation $0 \to 2$. Following this, the clip 1 connects to the clip 2 allowing for the propagation $1 \to 2$ but also $2 \to 1$. Following the order of transitions, we might end up in a case where an excitation follows the path $0 \to 2 \to 1 \to 3$, thus losing all interest in using this method for controlling the direction of propagation.

This problem of directionnality comes from the fact that we cannot favor a direction of propagation by the use of MZIs.

Even if we can avoid this phenomenon for 2-input-2-output graphs by strategically playing with the order of connections, we cannot avoid to encounter backpropagation to input clips. This causes a loss of information. Furthermore, this issue is unavoidable for larger circuits by different ordering for the MZI connections.

It's important to note that even if we have a single photon as input, the input state can be initialized as a superposition. We aim to tune every possible transition because the input might change from one epoch of the learning process to another.

This concept of backpropagation is introduced here with a simple circuit that corresponds to the ECM from the Prisoner's Dilemma (see Section IV). However, it can be generalized to any kind of DAG. The subtlety of the problem lies in the fact that the graph is directed.

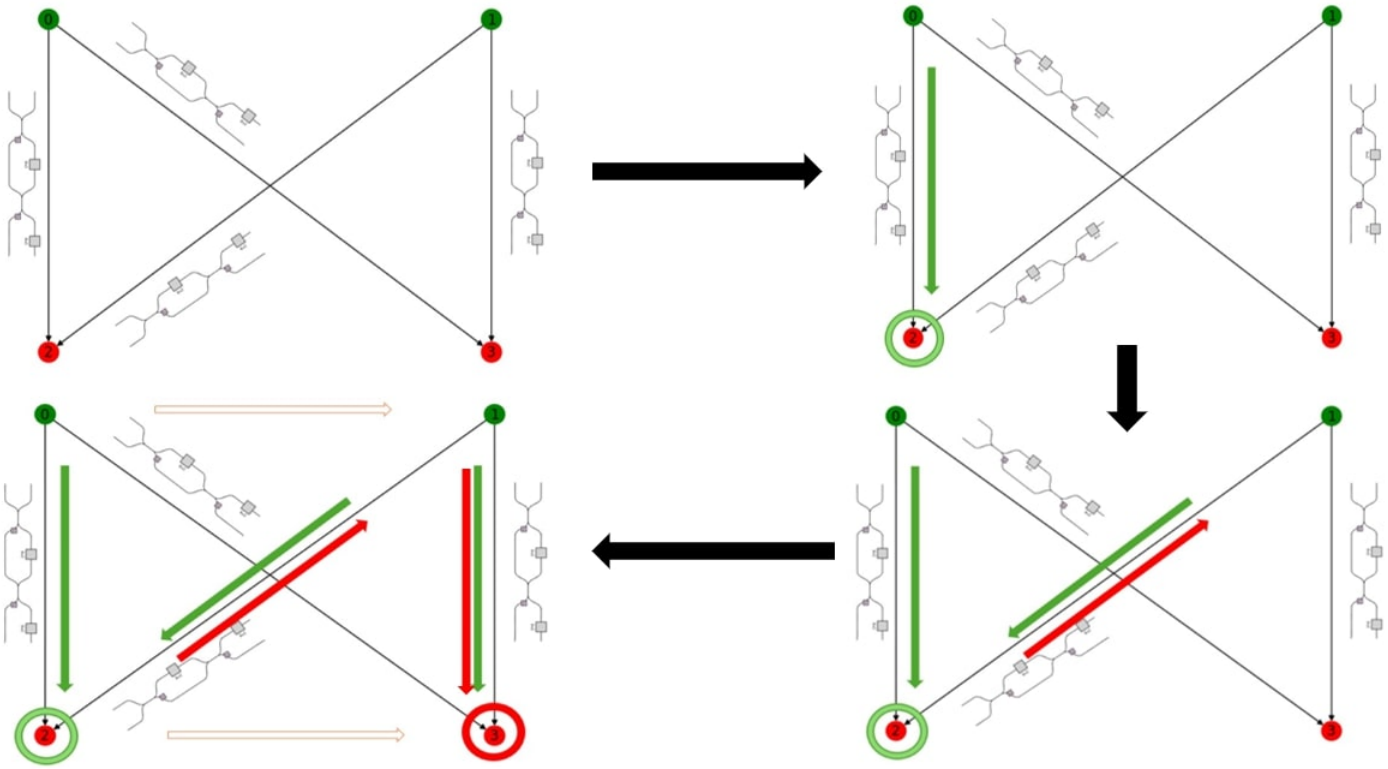


Figure 31: *example of backpropagation. By following the path of the excitation corresponding to the order of the transitions, we notice that the original wanted (green) excitation finally propagates to an unwanted node (red propagation).*

Isolation of Output Clips with Ancillary Nodes

The initial solution proposed to address the issue of backpropagation in optical circuits was to introduce of what are known as ancilla modes. The primary objective of this approach is not to completely eliminate backpropagation, but rather to prevent an excitation from passing through multiple output modes, as depicted in Figure 32.

The fundamental concept behind this solution is the creation of sort a of clone (an ancilla) of an input clip. This clone will contain the excitation that is not intended to be transmitted to the first output clip. The original input clip will be linked to the first output, and the clone will be linked to the second output clip.

However, this method is not without its challenges. The first issue, as previously explained, is that backpropagation still occurs. This means that it is still possible for an output clip that is already excited to lose its excitation. The second major issue is the added complexity this method introduces when creating the circuit. It necessitates to add extra optical modes, which number increases with the size of the DAG.

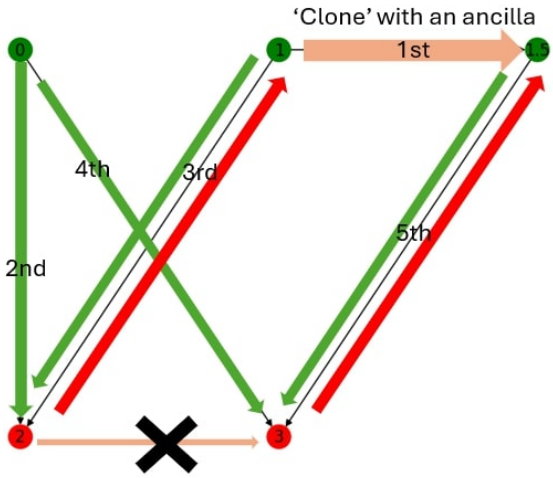


Figure 32: *Example of the ancilla solution. The introduction of the ancilla permits to avoid the unwanted propagation to reach the output clips.*

Attempting to implement the ancillas such that the order of transitions does not matter results in an exponential increase in the number of modes with respect to the number of output modes, as expressed by the formula $\mathcal{O}(in + out) \longrightarrow \mathcal{O}(in^{out} + out)$. It can be shown noticing that every input mode will create an ancilla for every connected output mode, thus the size of the optical circuit grows exponentially with the number of output states.

Even with smart implementation, it is still necessary to introduce more spatial modes to perform the computation, which poses a problem when considering the physical limitations of the computer. Being able to save even one or two modes can be crucial in the realization of some computations on a quantum computer. The limitation of Quandelas' computer was of 12 modes by the beginning of this thesis.

Using an ancilla solution permitted to implement the prisoners dilemma circuit using 16 modes, which was already to much for the circuit. The maximum number is now 20 modes.

In light of these challenges, I have been working on an alternative method, developed and formalized by myself. This new approach, aims to tackle this problem in a more efficient and effective manner.

VI.2 Node Reduction Clip Grouping

Direct Propagation with Hyper-Nodes

Upon examining the graph depicted in Figure 14, it is easily noticeable that both input clips are exciting each 2 of the output clips. This observation can be leveraged to our advantage by recognizing that this scenario does not necessitate a connection between both input clips and the output clips. Instead, what is essential for is only for the excitation to have the ability to propagate to the output clips. The ultimate determinant of success will be the distribution of excitation in the output clips and their respective phases. Interestingly, the phase here is not of primary importance as it can be modified for each output clip at a later stage.

This observation leads to a significant consequence: we can consider the group composed of the two input clips as a single clip. This unified clip will transmit all the excitation from the sub-input clips simultaneously to the output clips such that the merged excitation can thus be completely transmitted. As demonstrated in Figure 33, the grouping process involves adding Mach-Zehnder Interferometers (MZIs) between the clips within the formed groups. This arrangement allows the photons to be fully or partially transmitted at once to the output clips.

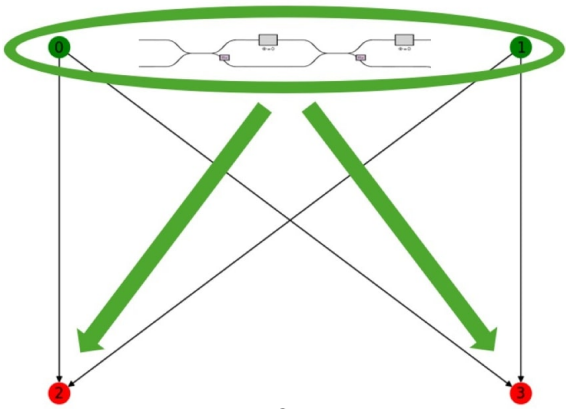


Figure 33: *Grouped ECM from Figure 14. This method permits to transmit the excitation to the output clips at once, thus not suffering from backpropagation.*

This strategy effectively circumvents the issues of backpropagation and ancilla modes simultaneously. However, it becomes more complex for larger graphs as all input clips do not always act on the same output modes. This complexity necessitates an analysis of graphs to characterize the different groups of input clip.

Despite these challenges, this initial step in the grouping technique is sufficient to construct the circuit used for the Episodic Compositionnal Memory (ECM) for the Prisoner's Dilemma game. In Figure 34 is shown the circuit used for the prisoners dilemma game. The first and third MZIs are utilized to group the input clips before each transmission. The phase shifters are fully tunable and are trained during the learning process to enable all possible manipulations of the excitation. The second MZI corresponds to the propagation of the excitation to the first output clip, and the fourth MZI dictates the excitation propagation to the second output clip.

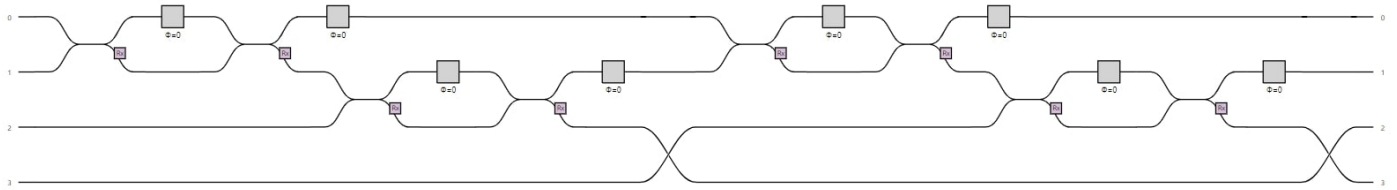


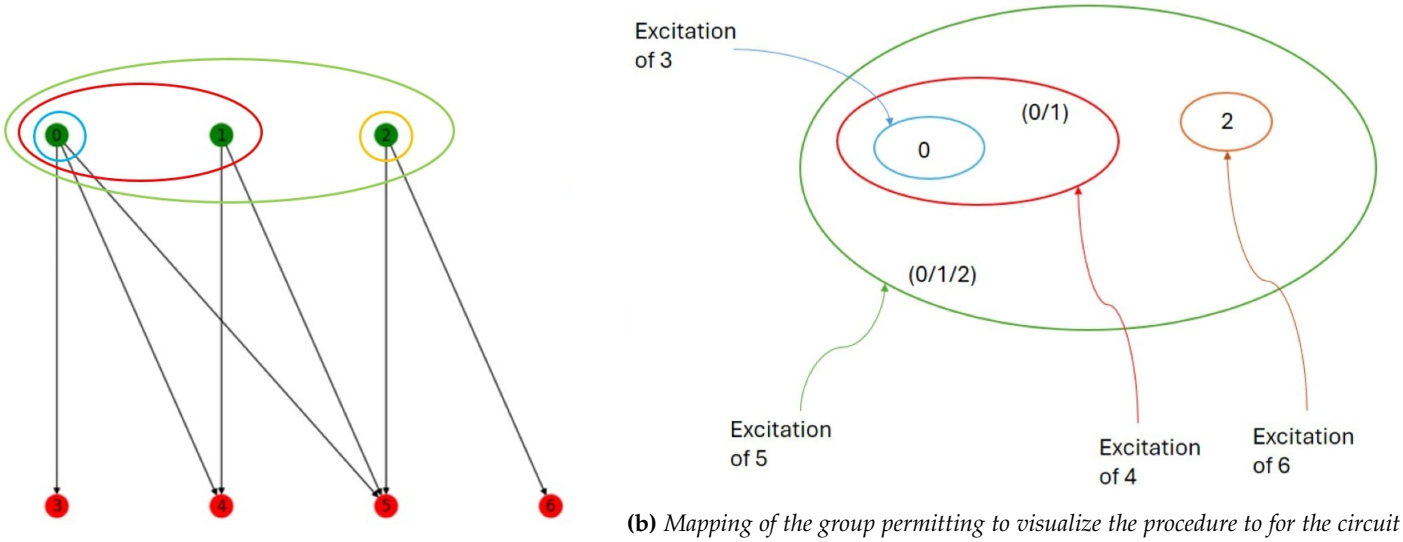
Figure 34: *Circuit corresponding to the ECM shown in Figure 14. The 2 input modes are grouped using MZIs and then transmit the excitation to the first output mode (mode 2). Following this, the input modes are grouped again to reshare their excitation and transmit to the second output mode (mode 3).*

Categorization of Clip Groups

Particularly when dealing with more complex graphs, the process of categorizing groups becomes a crucial step in constructing an ideal circuit. This categorization helps in simplifying the structure and understanding the interconnections within the graph. Currently, our primary interest lies in 2-layer graphs, which, despite their apparent simplicity, can exhibit a rich variety of behaviors and characteristics.

The extension of our analysis to graphs with a higher number of layers is a direct process. This is achieved by separating the transitions between layers and adapting the same technique that was previously employed for hyper-graph transitions, as demonstrated in [3]. This approach allows us to maintain the same analytical framework while extending our reach to more complex structures.

To illustrate this process, let's consider the example depicted in Figure 35. This figure presents a 2-layer graph that, while more complex than some of the simpler structures we might encounter, provides a valuable case study for our discussion.



(a) *Example of a complicated 2 layer graph.*

(b) *Mapping of the group permitting to visualize the procedure to for the circuit of Figure 35.*

Figure 35: *Example of grouping for a complex 2 layers graph.*

The primary objective in this analysis is to identify groups of input clips that are going to act commonly on the action clips. This commonality of action can reveal important insights about the structure and behavior of the graph. The most straightforward way to achieve this is to label all input clips that are associated with a particular output clip and repeat this process for every output clip.

In the example shown in Figure 35, we have the following associations:

- output 3 $\longrightarrow$ input 0
- output 4 $\longrightarrow$ input 0,1
- output 5 $\longrightarrow$ input 0,1,2
- output 6 $\longrightarrow$ input 2

Once this mapping is established, the representation in Figure 35b becomes useful for identifying the order in which to apply the transitions. Given that a group of inputs cannot be unlinked, it is crucial to link the input clips only immediately before the corresponding transition. Furthermore, the transitions should be applied in increasing order with respect to the size of the group.

Ensuring that these conditions are met allows for the excitation to propagate from any input clip only to any connected output clip. This controlled propagation is a key feature of the system and is crucial for its correct operation.

The corresponding circuit, which adheres to these principles, is depicted in Figure 36. This circuit provides a visual representation of the graph and serves as a guide for understanding its behavior.

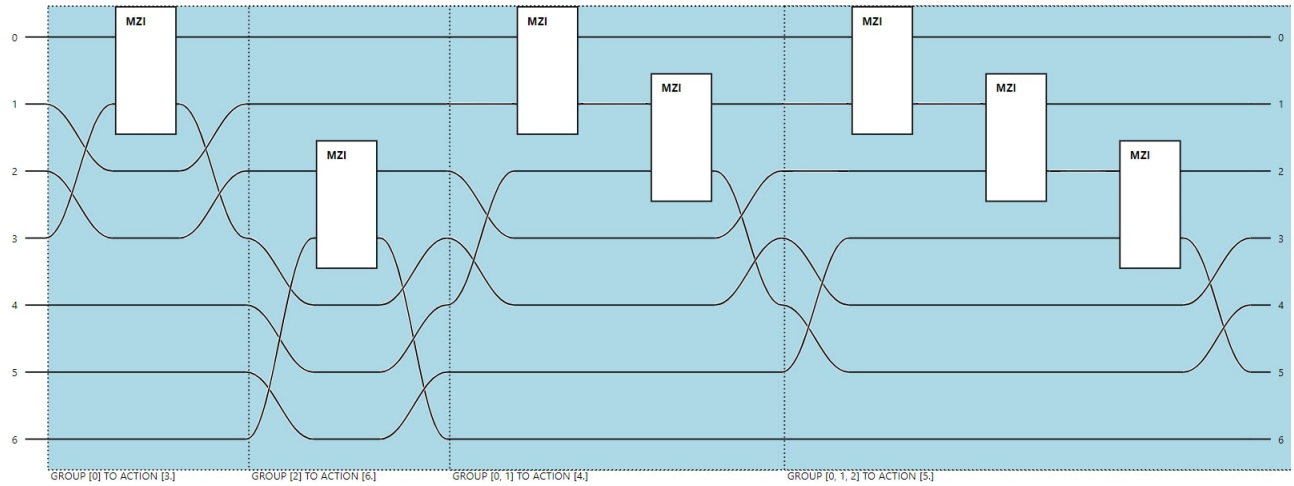


Figure 36: *Circuit corresponding to the complex graph shown in Figure 35a.*

Ancillary Nodes Exceptions

In the context of our model, there are specific conditions under which the introduction of ancillary modes becomes necessary. An illustrative example of this can be seen in Figure 37.

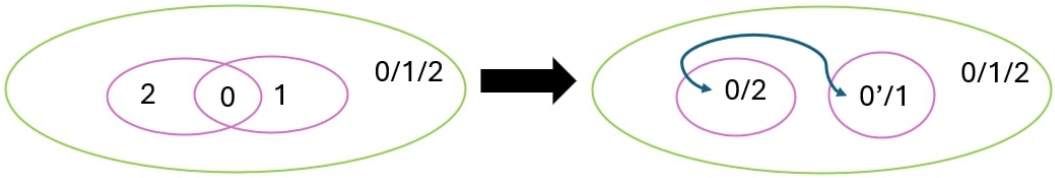


Figure 37: *Example of grouping requiring the introduction of an ancilla*

The condition that necessitates the introduction of ancilla modes can be formalized as follows:

when two groups within the model contain the same subgroup, the introduction of at least one ancilla mode becomes necessary to decouple the groups. This decoupling is crucial for ensuring the independent operation of the groups, thereby preserving the integrity of the computational process.

It's worth noting that while there may exist techniques that could potentially reduce the need for ancillas even further. Such techniques are beyond the scope of this thesis. The focus here is on reinforcement learning, and the current model, with the use of ancilla modes, is deemed sufficient for this purpose.

Formalisation of a Procedure

In the context of the specific problem that we are currently addressing, we use 2-layer ECMs. Pistes for the generalization of this procedure for multi-layer ECMs will be presented in a later section.

Let's first define several useful variables:

- $\mathcal{I} = \{i\}$ the set of input clips.
- $\mathcal{O} = \{o\}$ the set of output clips.
- $\mathcal{T} = \{t_{i \to o}\}$ the set of **existing** transitions in the ECM between an input clip i and an output clip o.

The first important step, once those variables are defined, is to determine the parents of each output clips. This step corresponds to defining the groups that will be formed later in the optical circuit.

To do so, we loop over every element $o \in \mathcal{O}$:

we define $p_o = \{\iota \mid \iota \in \mathcal{I} \ \ \& \ \ t_{\iota \to o} \in \mathcal{T}\}$ the set of input clips ι parent of an output clip o. To construct p_o, on can loop over the input clips and verify that a transition exists between the marked input and output.

We create $\mathcal{P} = \{p_\theta \mid \theta \in \mathcal{O}\}$, the hyper-set of parents of output clips. This construction is important because we need to verify whether any set $p_\theta \in \mathcal{P}$ overlaps with another set $p_\vartheta \in \mathcal{P}$. If yes, we will have to introduce ancillary modes, if not, a specific transition order and grouping order is enough.

So let's loop over $p_\theta \in \mathcal{P}$ and another $p_\vartheta \in \mathcal{P}$:

If no elements are shared between p_θ and p_ϑ, or if $p_\theta \subset p_\vartheta$, or if $p_\vartheta \subset p_\theta$, then there is no need to introduce ancillary modes.

Else, if an element e or a set of element $\mathcal{E} = \{e\}$ is shared by p_θ and p_ϑ and $p_\theta \not\subset p_\vartheta$ and $p_\vartheta \not\subset p_\theta$, then we do :

- For each element $e \in \mathcal{E}$, we create an new clip (ancilla) ε.
- we create a new transition $a_{e\to\varepsilon}$
- if not created already, we create $\mathcal{A} = \{a_{e\to\varepsilon}\}$ the set of ancillary transitions.
- In p_ϑ we replace e by ε

The final step is to construct the circuit. The number of modes of the circuit will be determined by both the size of the ECM and the number of ancillas added during the previous step. The exact number of modes is given by

$$n_{modes} = i_{nput} + a_{ncilla} + o_{utput}$$

Each of these modes is attached a label i_x, a_x or o_x to distinguish them from each other. Every link, or transition, between mode is added via a Mach-Zehnder Interferometer, but the order of them in the circuit will matter and it requires dedicated attention.

We start the construction of the circuit by introducing the transitions in $\mathcal{A}$ to create the ancillary modes. Those transitions, as the other ones are fully tunable, allowing for an input clip of the ECM to transmit all, none or partial excitation to an associated output clip. For this step, we just take every transition $a_{e\to\varepsilon} \in \mathcal{A}$ and add an MZI linking between the modes $e\,(\in \mathcal{I})$ and $\varepsilon\,(\in \mathcal{A})$ with no consideration for the order, no grouping is performed yet.

Here comes the subtlety, for each $p_\theta \in \mathcal{P}$ **in the increasing size order**, we group the input/ancillary modes and link one of them to the output clip: For each $\iota \in p_\theta$ and its successor $S(\iota)$, S being the successor operation, we add an MZI between the modes $\iota \in \mathcal{I} \mid \mathcal{A}$ and $S(\iota) \in \mathcal{I} \mid \mathcal{A}$. If $S(\iota)$ does not exist, then ι is the last element of p_θ and we add an MZI between $\iota \in \mathcal{I} \mid \mathcal{A}$ and $\theta \in \mathcal{O}$.

Taking a random graph, see the given in Figure 38, we can test the performance of this procedure.

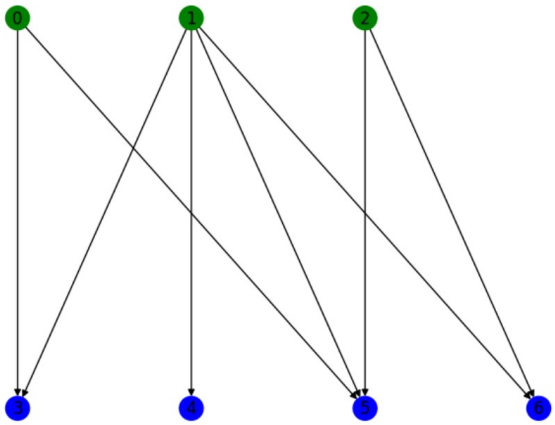


Figure 38: *Complex graph taken as an example to test the developed algorithm for the need of ancillas.*

For this graph, using the ancillary method, it is needed to introduce many ancillary modes. The clip 0 having 2 transitions, we need 1 ancillary mode, 3 for the clip 1 and 1 for the clip 2. All this adding up shifts the total number of modes from 7 to 12.

Using the procedure explained above, this number is down to 2 ancillary modes. We can follow the reasoning explained if we look at the parents of each output clip :

- $3 \to \{0,1\}$
- $4 \to \{1\}$
- $5 \to \{0,1,2\}$
- $6 \to \{1,2\}$

We notice that clip 1 is shared by $\{0,1\}$ and $\{1,2\}$ but neither of those sets is included in the other one, we create $1'$ such that $\{1,2\}$ becomes $\{1',2\}$. The same phenomenon can be observed for the clip 2 in $\{1',2\}$ and $\{0,1,2\}$. We then need to create an ancilla for $\{1',2\}$ to become $\{1',2'\}$.

The associated circuit is shown in Figure 39.

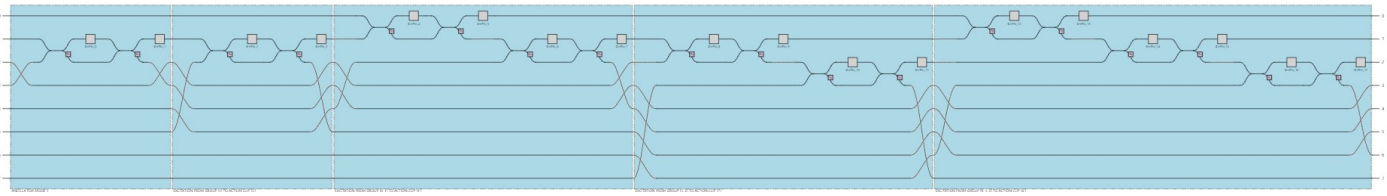


Figure 39: *Circuit associated to Figure 38 and generated using the clip grouping procedure.*

Application to Multi-Layer ECM

To scale this method to multi-layer ECMs, one could think about defining precise layers in the ECM such that the hypergraph contains several 2-layer graphs. Then applying the same method is direct.

However, the limitations in the number of modes needs to be taken into consideration. One very promising way to investigate for further research is to create a circuit which is as big as the biggest 2-layer sub-graph of the hypergraph and tune the MZI in such a way that all excitations from an upper layer is transmitted to its lower layer. Precisely tuning MZI could lead to such solution but will require further investigation in the tuning and specific procedure for this method.

We also have to take into account that an excitation transition can be transmitted from one layer to another layer with large distance between them. To solve this problem, we can either introduce sort of dedicated 'storage modes' or new transitions between single clips.

Further research will be dedicated to this problem.

Limitations for Multi-Photon Input

The method under discussion, while robust in many respects, does have its limitations. The most significant of these is its constraint to single excitation, or 1 photon input. This limitation becomes particularly evident when the method is applied to multi-photon inputs.

When both ancillary and grouping methods are employed in the context of multi-photon inputs, the results are far from satisfactory. This is primarily due to the uncontrollable interference effects that arise between the multiple photonic excitations. These interference effects make it impossible to favor any particular direction for the photons within the circuit or to manipulate the distribution of excitation.

VII Probability Recycling

Error mitigation is a field of research in computer science that aims at improving noisy computational results. Research in this field is very important for the application of quantum computing as its main limitation rely on the noise to which the measurements and computation are subjected.

In the specific context of photonic quantum computing, there are several primary sources of noise that pose challenges to the accuracy and reliability of computations. These include the loss of photons within the circuit, the failure of the source to emit single photons, and the failure in the detection of photons. Additionally, the 'jump' of a photon from one mode to another, as well as issues related to the tuning of the optical components and the timing to when all the photons are sent into the computer, also contribute to the overall noise in the system.

Probability recycling is a sophisticated method that falls under the broader category of quantum error correction. This method, which was introduced recently in[57], is specifically designed to mitigate the issue of photon loss. It takes into account all three cases of photon loss - those occurring inside the circuit, emission errors, and detection errors. These three types of errors can be mathematically treated as equivalent to missing photons in the output state, thereby allowing for a unified approach to their mitigation.

However, it's important to note that these errors are distinct from photon number resolution errors. The latter type of error arises due to the fact that Quandela's photon detectors do not differentiate between a single photon click and multiple photon clicks. In other words, they do not permit photon number resolution. This presents a unique challenge that requires a different approach to error mitigation.

In [57] they present compelling evidence that the technique of probability recycling significantly outperforms the method of post-selection of states in quantum computers that are subjected to high levels of noise. More precisely, this superiority becomes evident when the probability of losing a photon, denoted as η, is much greater than 0.5.

Post-selection, in this context, refers to the process of discarding output states that have experienced photon loss and retaining only those output states that possess the correct number of photons. These retained states are then used to compute an estimate of the output state probabilities. However, when the noise level η, exceeds 0.5, the number of post-selected output states can be drastically reduced. This scarcity of usable output states can lead to estimations of low fidelity, thereby compromising the accuracy of the computations.

Given these considerations, my objective is to implement the technique of probability recycling with the aim of potentially enhancing the outputs derived from computations performed on the actual Quantum Processing Unit (QPU). The anticipated improvement stems from the ability of probability recycling to effectively mitigate photon loss, thereby increasing the number of usable output states and, consequently, the fidelity of the estimations.

However, it's important to note that while the potential benefits of implementing probability recycling are promising, it has not yet been definitively proven that this algorithm can effectively handle other sources of noise and successfully avoid biases due to these noises. Therefore, a degree of caution and skepticism is warranted when considering the application of this technique.

As part of my ongoing research, I plan to implement this technique for two specific scenarios: the Prisoners Dilemma and the invasion game. The results obtained from these implementations will then be compared with those from perfect (SLOS) runs and noisy simulations with post-selection. This comparative analysis will provide valuable insights into the effectiveness of probability recycling in real-world applications and contribute to the broader understanding of error mitigation techniques in quantum computing. However, this analysis is not finished by the delivery date of this thesis and thus is not present in the following parts.

VII.1 Construction of Recycled Probabilities from Lossy Outputs

The construction of recycled probabilities is a crucial process that is based on the comprehensive analysis of lossy output states. This technique is extensively discussed in [57], which provides an illustrative example to elucidate the concept.

In this example, we consider a quantum circuit with six modes and three input photons. After conducting multiple experiments, we are able to compute an estimate of the output state probabilities. The post-selected estimate probability of an output state, denoted as:

$$|\phi\rangle = |111000\rangle$$

is calculated using the formula:

$$p_{PS}(|\psi\rangle) = \frac{n(|\psi\rangle)}{n(|3 \text{ photons}\rangle)}$$

Here, $p_{PS}(|\psi\rangle)$ represents the post-selected estimate probability of the state $|\psi\rangle$ and $n(|\psi\rangle)$ is the number of experimental runs in which we measure the state $|\psi\rangle$ in the output.

To mitigate the effects of photon loss, we compute the mitigated probability estimate using the following rule:

$$p_R(|\phi = 111000\rangle = \frac{n(|\phi - 1 \text{ photon}\rangle)}{n(|2 \text{ photons}\rangle)} * N = \frac{n(|110000\rangle) + n(|101000\rangle) + n(|011000\rangle)}{n(|2 \text{ photons}\rangle)} * N$$

In this equation, p_R is the recycled probability estimate and N is the normalization factor, which is given by:

$$N = \binom{4}{1}^{-1}$$

For a more formal representation, we can express the recycled probability estimate as:

$$p_R^k(s_l^n) = \frac{1}{\binom{m-n+k}{k}} \sum_{s_i^{n-k} \in \mathcal{L}(s_l^n)} p_{PS}(s_i^{n-k})$$

In this equation, we define n as the ideal number of photons and k as the number of lost photons. Thus, $p_R^k(s_l^n)$ is the probability estimated recycled from a k photon loss. The normalization factor also takes into account m, which is the number of modes of the circuit.

Detailing of the Algorithm

The process of reconstructing probabilities involves a precise algorithm, which is defined in the following steps:

Variable definition: The first step in the algorithm is to define two crucial variables, n and m. Here, n represents the number of photons in an ideal output, while m denotes the number of modes in the circuit. These variables play a significant role in the subsequent steps of the algorithm.

Pattern Generation: After defining the variables, the next step is to generate a pattern that corresponds to all possible states with the given number of photons and modes. This pattern serves as a reference for comparing the output states obtained from the experiment.

Experiment Execution: With the variables defined and the pattern generated, we proceed to run our experiment. The experiment yields noisy output states, which are then used for further computations.

Probability Computation: Using the results obtained from the experiment, we compute three key quantities: p_{PS}, P_R^{-1} and P_R^{-2}. These quantities are computed as explained in the previous section. In addition, we also compute the norm corresponding to the perfect states, denoted by N, using the formula:

$$N = \binom{m}{n}^{-1}$$

State Evaluation: For each state, we evaluate whether N is comprised between P_R^{-1} and P_R^{-2}. If it is, we consider N to be a good estimated value. However, if N does not fall within this range, we consider the maximum of the exponential fit between values at positions:

$$\{1 : p_R^{-1}, 2 : p_R^{-2}, 50 : N\}$$

This step ensures that we have a reliable estimate for each state.

Normalization and Final Probability Estimate: After evaluating each state, the final step is to normalize the results. This normalization process ensures that the probabilities are correctly scaled. Once the results are normalized, we obtain the final probability estimate.

This algorithm provides a systematic and rigorous approach to reconstruct all probabilities from noisy output states. It is a crucial tool in quantum computing, enabling us to mitigate the effects of noise and obtain more accurate results.

VII.2 Performances of Probability Recycling

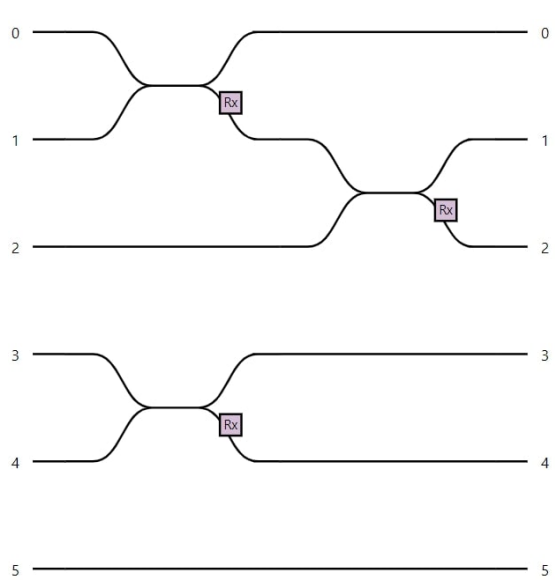
In order to test the probability recycling method presented above, we are going to test its efficiency by comparing for several circuits and several noise level, the results got from post selection and probability recycling. In order to do this we introduce the distance metric for the probability estimation of the output states. This distance is defined as

$$d(\mathcal{S}) = \sum_i |p(s_i)^\star - p(s_i)|$$

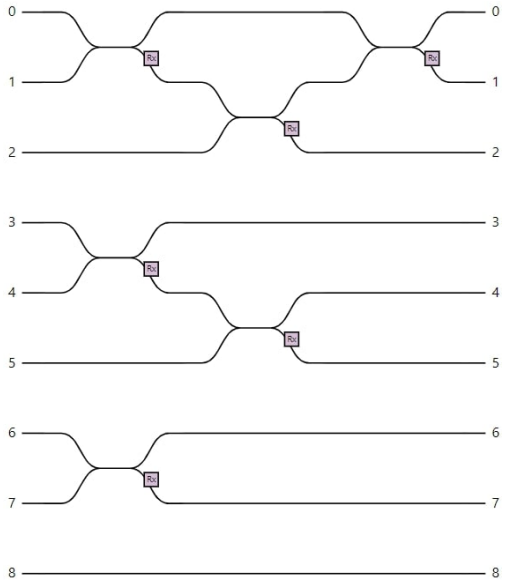
with $\mathcal{S}$ the output sample, $p(s_i)^\star$ the exact probability to measure the output state s_i and $p(s_i)$ the estimated probability.

The distance is then computed for loss levels from 0.001 to 0.999 and for different number of samples and different circuits.

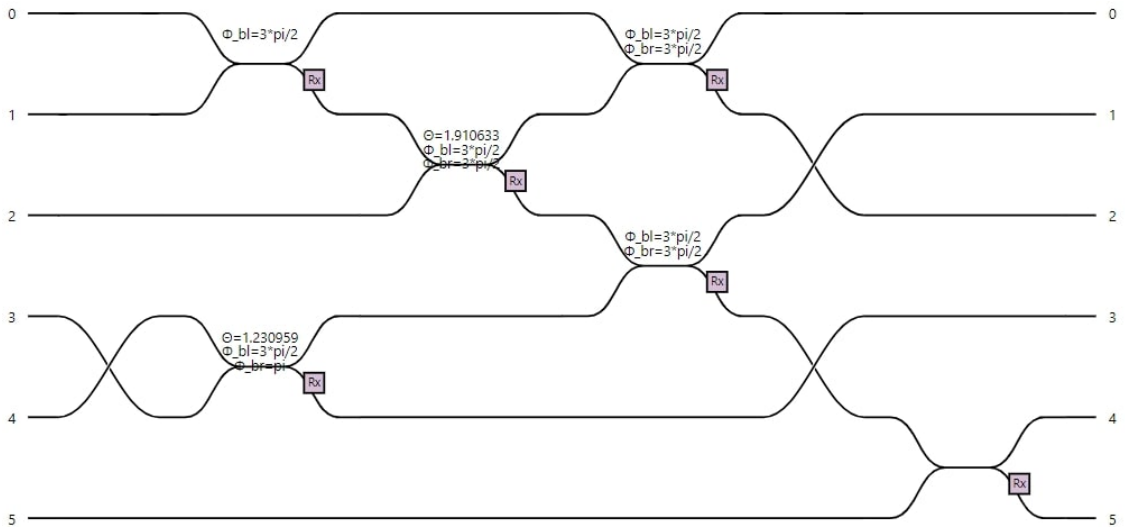
For the following analysis, we choose 3 circuits and associated input states to observe the effect of the disparity of output states and length of state.



(a) ***Simple circuit*** *with input state* $|1,0,0,1,0,1\rangle$



(b) ***Bigger circuit*** *with input state* $|1,0,0,1,0,0,1,0,1\rangle$



(c) ***Complex circuit*** *with input state* $|1,1,1,1,0,0\rangle$

Figure 40: *Circuits for the analysis of the probability recycling algorithm*

The idea of the 2 first circuits is to see the effect of the circuit size on the distance between the ideal and post-selected/mitigated probabilities. The last circuit is supposed to generate entangled states with post selection on modes 4 and 5. Other states are generated by this circuit and may contain spatial modes with more than 1 photon. This case is not taken into account by the probability recycling technique but is interesting to look at.

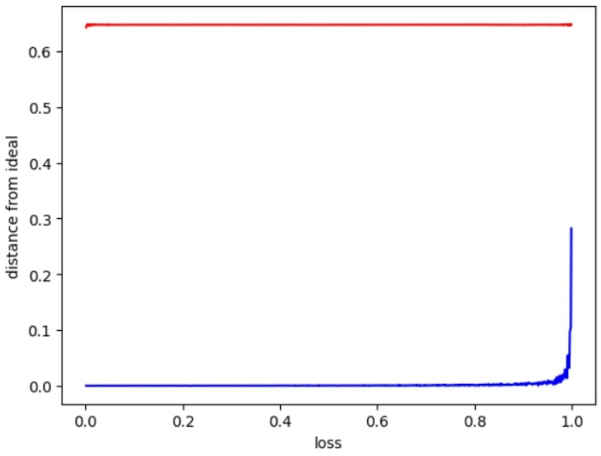
Simple Circuit

For this case, we plot the distance for the mitigated distribution and the post selected one using 10^8, 10^5 and 10^3 samples.

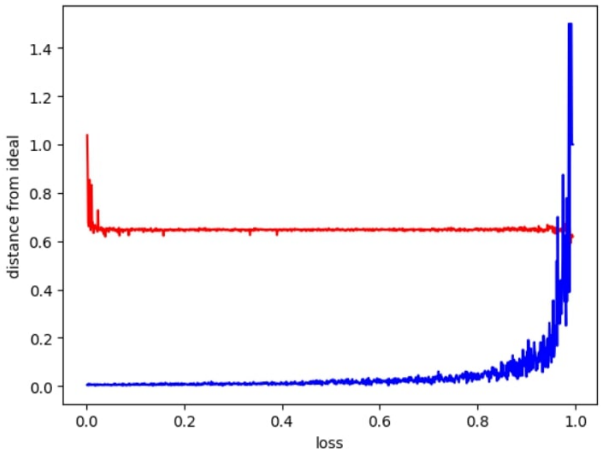
As expected, for a very small amount of noise, the post-selected probability corresponds almost exactly to the ideal probability as all states can be used in order to construct those probabilities. However, when the losses become bigger and bigger, fewer and fewer states are being post selected and thus, only in the case for which we get 10^8 samples we can keep a better probability estimate with the post-selected probabilities. In real life computing, losses are usually of the order of 0.9 and generating a big amount of samples is very expensive. As we use less and less samples, the mitigated

probabilities become of big interest, both for the better performances with high losses but also for the stability of its results with different noise levels, making the estimation of the viability of obtained results easier.

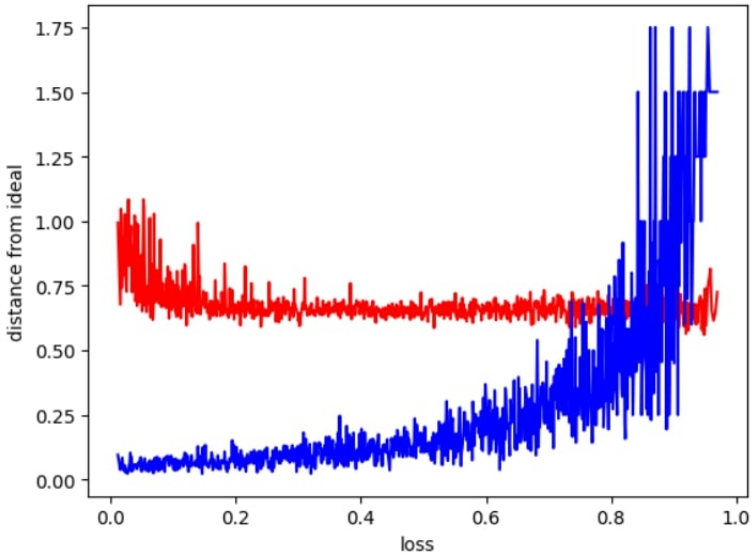
We remark as expected that introducing more photons and more modes results in a higher overall sensibility to noise. It can be observed for the post selected probability as it starts increasing earlier but we can also see that the mitigated probability is not maintained at the same distance as in the simple case. However, we still confirm that tendency of getting better results with mitigated probabilities for high losses.



(a) *samples* = 10^8*, this number of sample still favors the post-selected probability as the amount of valid samples stays sufficient to evaluate the output probability*



(b) *samples* = 10^5*. Already an advantage is observed for the probability recycling, but only for very high loss*



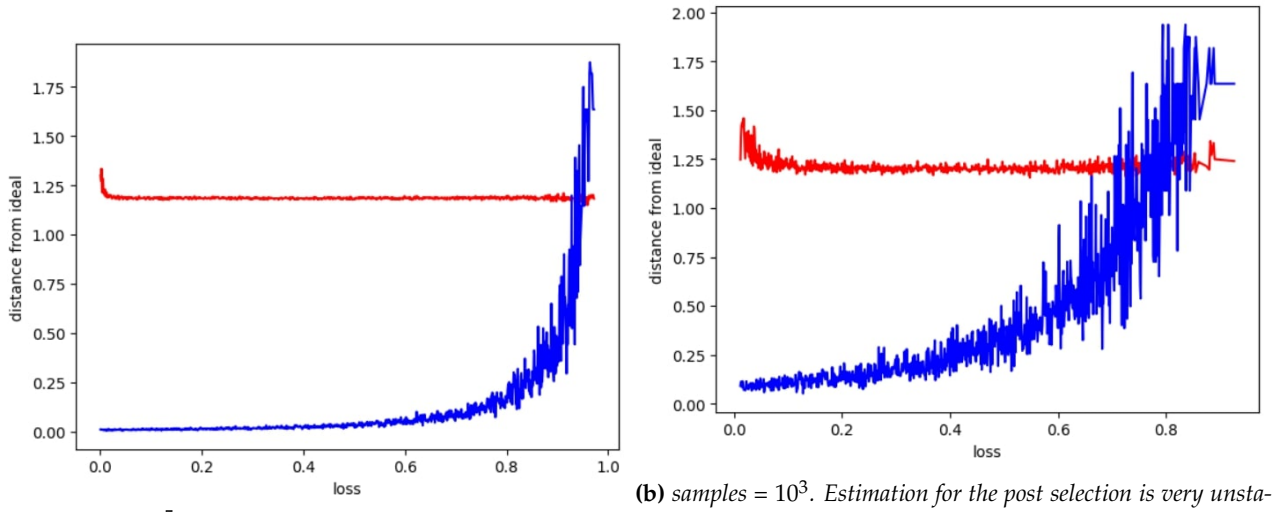
(c) *samples* = 10^3*. With few samples, it is way more efficient to use the probability recycling for loos over* 0.7

Figure 41: *distance from ideal for* ***blue:*** *post-selected probability and* ***red:*** *mitigated probability*

Bigger Circuit

For the bigger circuit, we compute the same probabilities but with 10^5 and 10^3 samples. The results are shown in Figure 40b.

We observe very similar results compared to the simple circuit case. The main difference is the starting point of the exponential decay of performance of the post-selection probability estimate. This can be explained by noticing that the use of more photons in the circuit implies a bigger sensitivity to noise. The mitigated probability stay stable as in the previous case but with a higher value, due to the bigger amount of possible output states.



(a) *samples* = 10^5. *The post selection probability stays a better estimate until losses of* 0.9

(b) *samples* = 10^3. *Estimation for the post selection is very unstable so unreliable. The mitigation is constant and advantageous for loss over* 0.7

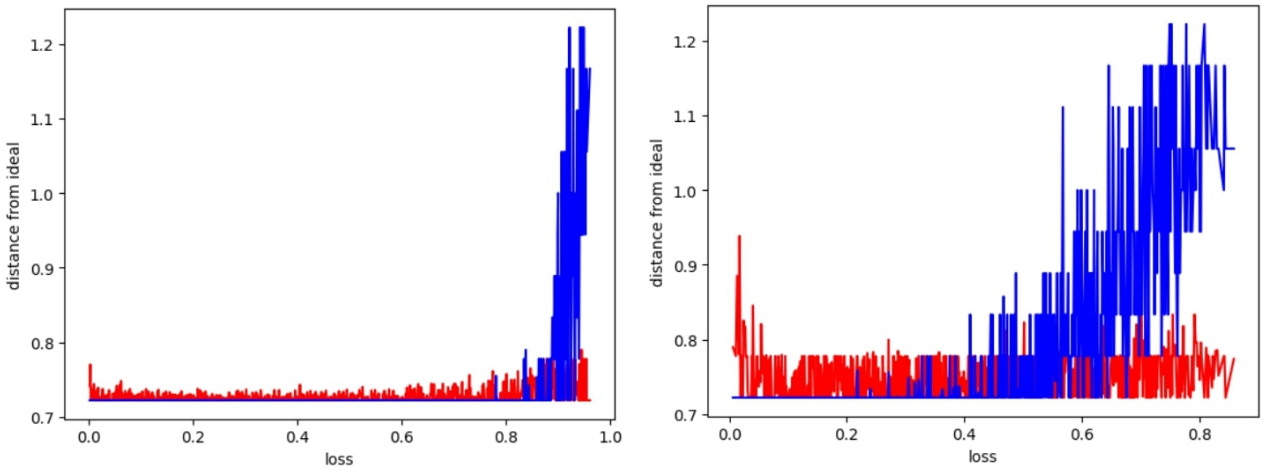
Figure 42: *distance from ideal for* ***blue:*** *post-selected probability and* ***red:*** *mitigated probability*

Complex Circuit

For the complex circuit, we expect the results to be different. The main reason being that both the post selected and mitigated estimate of the probability are computed without taking into account states for which an optical mode contains more than 1 photon. Thus the post-selected probability with no losses will not correspond to the ideal case.

In this case, there is no way to retrieve a perfect probability, either by using post selection or probability recycling. We observe in Figure 43 a more stable result for the post selection with low noise, but as the noise grows, mitigation becomes more interesting to use to estimate the output probability.

This shows that we could benefit from using mitigated probability estimate when performing computations on very noisy devices. This analysis will be the subject for further research, in order to determine the viability of this error mitigation method for noisy devices which do not only suffer from photon losses.



(a) *samples* = 10^5. *Post selection stays advantageous for loss going until* 0.9

(b) *samples* = 10^3. *Both estimates are very noisy. The advantage for mitigation appears for loss over* 0.5

Figure 43: *distance from ideal for* ***blue:*** *post-selected probability and* ***red:*** *mitigated probability*

VIII Conclusion

In this thesis, I explored the subject of quantum machine learning through the QOPS framework.

My initial work was to improve the implementation of this framework for the Prisoners dilemma use case. This improved implementation permitted to test the capabilities of the framework on several aspects. The main feature of QOPS is to reflect the nature of Projective Simulation using a quantum system. The QOPS framework was tested for the implementation of several instances of the prisoners dilemma. The different sets of strategies were classical strategy (C,D), simple Quantum strategy (C,D,Q), and General Unitary. The analysis comprised the ability of the framework to converge to the Nash Equilibrium strategies for all cases. It shows that QOPS efficiently learns from its environment, as its classical counterpart, while keeping the interpretability of PS.

This efficiency have been proven for different conditions, notably on Strong Linear Optics Simulator, indicating the validity of the model, but also on noisy simulator sim:altair, provided by Quandela cloud. The success of the model to learn in a noisy environment proves its resilience and stability in non-perfect cases.

This work is original and has led to the publication of a research paper : *Interpretable Quantum Reinforcement Learning with Entanglement: an Application using Single-Photons*.

This paper has not been published yet by the submission date of this thesis.

Elaborating on the QOPS framework, a second objective of my internship was to study the possibility of using a multi-photon state as input for a learning circuit. The long term goal was to develop a framework based on QOPS that utilizes multiple excitations for the input. After facing multiple issues due to the use of a plurality of photon, I managed to develop a framework as desired that we call meQOPS for multi-excitation Quantum Optical Projective Simulation.

The framework that have been developed is based on a novel approach of multi-photon treatment introduced by myself called Qudit Parallel Circuit Processing.

After the introduction of this technique, I applied meQOPS to the use case of the invasion game. This permitted to evaluate the robustness of the framework by using the Strong Linear Optics Simulator, provided by Perceval. I also studied the behavior of the framework with the introduction of noise in the model with the Simulator called sim:altair provided by Quandela cloud.

With the study of the different instances of the invasion game model, we conclude that the meQOPS framework performs under the simple constraint of the Binary Variable model, the Decoy Binary Variable model and the Generalized Multi-Variable model. This indicates that the framework is well adapted to treat multiple variables as input.

Moreover, the experiment was performed on the noisy simulator for the Binary Variable model and the Decoy Binary Variable model. The results show that the framework is also resilient to noise, which is a good feature for future experiments on real Quantum Processing Unit.

The experiment was not performed for the Generalized Multi-Variable model due to circuit size limitations. It would be interesting as a future work to introduce the feature of separable sub-circuits in order to perform bigger computation with a lower sensitivity to noise. This idea of separable sub-circuit was introduced by myself but not finished by the submission date of the thesis, thus not present here.

As a side project, I worked on the efficient translation between Directed Acyclic Graphs and Linear Optical Circuit. This side project appeared as a natural extension of my work on the implementation of the prisoner dilemma.

At the beginning of my thesis, a first implementation of the prisoners dilemma have already been made. However, the encoding of the ECM for this problem as an optical circuit was leading to the construction of a circuit with 16 modes. At that time, the maximum number of optical modes in a circuit, given by sim and qpu:ascella, was 12, thus making the implementation of this problem on noisy simulator and quantum processing unit impossible.

I developed a novel method for the translation between DAG and Linear Optical Circuits called *Clip Grouping* that allows for a massive reduction of the number of ancillary modes needing for the transcription to be made. In the case of the prisoners dilemma, this method allowed for the reduction of the circuit to implement from 16 modes to 12 modes. Only focusing on the individual ECMs, the associated circuits reduced from 6 modes to 4 modes.

In addition to the specific application of this method to the ECM of the prisoners dilemma, the method has been developed as a general way to efficiently translate from any kind of Directed Acyclic Graph to an optical circuit. This generalization comprises large Graphs, multi-layer Graphs and Graphs with ancillas. Some tests have been made for this method and show satisfying results. A more complete analysis is still to be performed for a complete characterization of the limitations of it.

This method will be later implemented for other cases of ECM and Neural Networks in the ase of Photonic Quantum Computing.

The final side project I have been working on is the implementation and test of the Probability Recycling method for Error Mitigation of photon loss. The project was initially supposed to be implemented for a direct use in the Invasion

Game use case. However, due to the progress made on the implementation of the Invasion Game, only a characterization of the method have been performed.

The probability recycling method have been already developed by the theory team from Quandela. My contribution to the work is the implementation of the method into the perceval code and the tests performed for different levels of noise.

This error mitigation method shows very promising results for applications with a significant level of noise. The reconstruction of probabilities becomes advantageous which compared to the already existing post processing method for high level of noise and low number of shots for each experiment. Those conditions actually reflects well the actual performance of the quantum processing unit. This makes the probability recycling method very interesting for further investigations. The implementation of this methods for the use cases of the prisoners dilemma and the invasion game are planned as future work but not completed by the submission date of this thesis.

Future Work

This thesis have permitted several advancements in the realization of experiments on photonic quantum computers and creation of algorithms/methods to improve those experiments. This work opens the possibility for new research.

The success of realization for the proof of concepts of the QOPS framework opens the scope for the implementation of this framework for real life applications. Those applications are especially focused on decision problems of multiple agents that needs to adjust their behavior depending on the others. One possible case for this application is the exploration of multiple drones that needs efficiently map a territory.

This can further extend to communication in logistic problems with automatized systems.

The development of the meQOPS framework has been successful, however, an in-dept analysis of its performances will be required in the future along with the development of optimization techniques specific to meQOPS.

A first optimization that can be studied is the formalization of separable sub-circuits that could potentially enlarge the actual capacities of this framework. Other specialized use cases for meQOPS can be studied in order to get closer to real world applications. Finally, one could study the behavior of this framework with entangled photon as input.

The two side projects, namely probability Recycling and Clip Grouping will both require a more in dept analysis and optimization before being integrated in several use cases for improving the overall performance of various algorithms.

Bibliography

[1] A. A. Melnikov, A. Makmal, V. Dunjko, and H. J. Briegel, "Projective simulation with generalization," *Scientific Reports*, vol. 7, no. 1, [Online]. Available: http://dx.doi.org/10.1038/s41598-017-14740-y.

[2] G. Franceschetto and A. Ricou, "Demonstration of quantum projective simulation on a single-photon-based quantum computer," 2024. [Online]. Available: https://arxiv.org/abs/2404.12729.

[3] P. A. LeMaitre, M. Krumm, and H. J. Briegel, "Multi-excitation projective simulation with a many-body physics inspired inductive bias," 2024.

[4] F. Flamini, M. Krumm, L. J. Fiderer, T. Müller, and H. J. Briegel, "Towards interpretable quantum machine learning via single-photon quantum walks," 2023.

[5] B. A. Bhuiyan, "An overview of game theory and some applications," *Philosophy and Progress*, vol. 59, no. 1-2, p. 111, 2016.

[6] K. L. Brown, W. J. Munro, and V. M. Kendon, "Using quantum computers for quantum simulation," *Entropy*, vol. 12, no. 11, pp. 2268–2307, Nov. 2010. [Online]. Available: http://dx.doi.org/10.3390/e12112268.

[7] I. M. Georgescu, S. Ashhab, and F. Nori, "Quantum simulation," *Reviews of Modern Physics*, vol. 86, no. 1, pp. 153–185, Mar. 2014. [Online]. Available: http://dx.doi.org/10.1103/RevModPhys.86.153.

[8] A. Steane, "Quantum computing," *Reports on Progress in Physics*, vol. 61, no. 2, pp. 117–173, Feb. 1998. [Online]. Available: http://dx.doi.org/10.1088/0034-4885/61/2/002.

[9] P. B. Upama, M. J. H. Faruk, M. Nazim, *et al.*, "Evolution of quantum computing: A systematic survey on the use of quantum computing tools," 2022. [Online]. Available: https://arxiv.org/abs/2204.01856.

[10] D. J. Brod, E. F. Galvão, A. Crespi, R. Osellame, N. Spagnolo, and F. Sciarrino, "Photonic implementation of boson sampling: A review," *Advanced Photonics*, vol. 1, no. 3, pp. 034 001–034 001, 2019.

[11] C. M. Kirsch and S. M. Lei, "Quantum advantage for all," *arXiv preprint arXiv:2111.12063*, 2021.

[12] T. S. Humble, H. Thapliyal, E. Munoz-Coreas, F. A. Mohiyaddin, and R. S. Bennink, "Quantum computing circuits and devices," 2018. [Online]. Available: https://arxiv.org/abs/1804.10648.

[13] P. Hauke, H. G. Katzgraber, W. Lechner, H. Nishimori, and W. D. Oliver, "Perspectives of quantum annealing: Methods and implementations," *Reports on Progress in Physics*, vol. 83, no. 5, p. 054 401, May 2020. [Online]. Available: http://dx.doi.org/10.1088/1361-6633/ab85b8.

[14] A. El-Kadi and R. Bondesan, "Quantum approximate optimisation for not-all-equal sat," 2024. [Online]. Available: https://arxiv.org/abs/2401.02852.

[15] N. Herrmann, D. Arya, M. W. Doherty, *et al.*, "Quantum utility – definition and assessment of a practical quantum advantage," in *2023 IEEE International Conference on Quantum Software (QSW)*, IEEE, Jul. 2023. [Online]. Available: http://dx.doi.org/10.1109/QSW59989.2023.00028.

[16] A. J. Daley, I. Bloch, C. Kokail, *et al.*, "Practical quantum advantage in quantum simulation," *Nature*, vol. 607, pp. 667–676, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:251132664.

[17] J. L. O'Brien, "Optical quantum computing," *Science*, vol. 318, no. 5856, pp. 1567–1570, Dec. 2007. [Online]. Available: http://dx.doi.org/10.1126/science.1142892.

[18] J. Lau, K. Lim, H. Shrotriya, and L. Kwek, "Nisq computing: Where are we and where do we go?" *AAPPS Bulletin*, vol. 32, Sep. 2022.

[19] R. MATSUMOTO and M. HAGIWARA, "A survey of quantum error correction," *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, vol. E104.A, Jun. 2021.

[20] E. Farhi, J. Goldstone, and S. Gutmann, *A quantum approximate optimization algorithm*, 2014. [Online]. Available: https://arxiv.org/abs/1411.4028.

[21] M. Cerezo, A. Arrasmith, R. Babbush, *et al.*, "Variational quantum algorithms," *Nature Reviews Physics*, vol. 3, no. 9, pp. 625–644, Aug. 2021. [Online]. Available: http://dx.doi.org/10.1038/s42254-021-00348-9.

[22] D. P. DiVincenzo, "The physical implementation of quantum computation," *Fortschritte der Physik*, vol. 48, no. 9–11, pp. 771–783, Sep. 2000. [Online]. Available: http://dx.doi.org/10.1002/1521-3978(200009)48:9/11%3C771::AID-PROP771%3E3.0.CO;2-E.

[23] J. Romero and G. Milburn, "Photonic quantum computing," 2024. [Online]. Available: https://arxiv.org/abs/2404.03367.

[24] I. L. Chuang and Y. Yamamoto, "Simple quantum computer," *Physical Review A*, vol. 52, no. 5, pp. 3489–3496, Nov. 1995. [Online]. Available: http://dx.doi.org/10.1103/PhysRevA.52.3489.

[25] P. Kok, W. J. Munro, K. Nemoto, T. C. Ralph, J. P. Dowling, and G. J. Milburn, "Linear optical quantum computing with photonic qubits," *Reviews of Modern Physics*, vol. 79, no. 1, pp. 135–174, Jan. 2007. [Online]. Available: http://dx.doi.org/10.1103/RevModPhys.79.135.

[26] R. Jozsa, "Entanglement and quantum computation," 1997. [Online]. Available: https://arxiv.org/abs/quant-ph/9707034.

[27] R. Okamoto, J. L. O'Brien, H. F. Hofmann, and S. Takeuchi, "Realization of a knill-laflamme-milburn controlled-not photonic quantum circuit combining effective optical nonlinearities," *Proceedings of the National Academy of Sciences*, vol. 108, no. 25, pp. 10 067–10 071, Jun. 2011. [Online]. Available: http://dx.doi.org/10.1073/pnas.1018839108.

[28] X. Zhou, D. W. Leung, and I. L. Chuang, "Methodology for quantum logic gate construction," *Physical Review A*, vol. 62, no. 5, Oct. 2000. [Online]. Available: http://dx.doi.org/10.1103/PhysRevA.62.052316.

[29] D. P. DiVincenzo, "Quantum gates and circuits," *Proceedings of the Royal Society of London. Series A: Mathematical, Physical and Engineering Sciences*, vol. 454, no. 1969, pp. 261–276, Jan. 1998. [Online]. Available: http://dx.doi.org/10.1098/rspa.1998.0159.

[30] D. M. Zajac, A. J. Sigillito, M. Russ, *et al.*, "Resonantly driven cnot gate for electron spins," *Science*, vol. 359, no. 6374, pp. 439–442, Jan. 2018. [Online]. Available: http://dx.doi.org/10.1126/science.aao5965.

[31] J. H. Plantenberg, P. de Groot, C. Harmans, and J. E. Mooij, "Demonstration of controlled-not quantum gates on a pair of superconducting quantum bits," *Nature*, vol. 447, pp. 836–839, 2007. [Online]. Available: https://api.semanticscholar.org/CorpusID:3054763.

[32] E. Knill, R. Laflamme, and G. Milburn, "A scheme for efficient quantum computation with linear optics," *Nature*, vol. 409, pp. 46–52, Feb. 2001.

[33] E. Knill, "Quantum gates using linear optics and postselection," *Physical Review A*, vol. 66, p. 052 306, 2002. [Online]. Available: https://api.semanticscholar.org/CorpusID:118104808.

[34] L. P. Kaelbling, M. L. Littman, and A. W. Moore, *Reinforcement learning: A survey*, 1996. [Online]. Available: https://arxiv.org/abs/cs/9605103.

[35] H. J. Briegel and G. D. las Cuevas, "Projective simulation for artificial intelligence," 2013.

[36] J. Mautner, A. Makmal, D. Manzano, M. Tiersch, and H. J. Briegel, "Projective simulation for classical learning agents: A comprehensive investigation," *New Generation Computing*, vol. 33, no. 1, pp. 69–114, Jan. 2015. [Online]. Available: http://dx.doi.org/10.1007/s00354-015-0102-0.

[37] W. L. Boyajian, J. Clausen, L. M. Trenkwalder, V. Dunjko, and H. J. Briegel, "On the convergence of projective-simulation–based reinforcement learning in markov decision processes," *Quantum Machine Intelligence*, vol. 2, no. 2, Nov. 2020. [Online]. Available: http://dx.doi.org/10.1007/s42484-020-00023-9.

[38] S. Zhu, I. Ng, and Z. Chen, *Causal discovery with reinforcement learning*, 2020. [Online]. Available: https://arxiv.org/abs/1906.04477.

[39] J. J. G. Luis, *Policy gradient rl algorithms as directed acyclic graphs*, 2021. [Online]. Available: https://arxiv.org/abs/2012.07763.

[40] D. Schmid and A. Sly, *On the number and size of markov equivalence classes of random directed acyclic graphs*, 2022. [Online]. Available: https://arxiv.org/abs/2209.04395.

[41] V. Saggio, B. E. Asenbeck, A. Hamann, *et al.*, "Experimental quantum speed-up in reinforcement learning agents," *Nature*, vol. 591, no. 7849, pp. 229–233, Mar. 2021. [Online]. Available: http://dx.doi.org/10.1038/s41586-021-03242-7.

[42] F. Flamini, A. Hamann, S. Jerbi, L. M. Trenkwalder, H. P. Nautrup, and H. J. Briegel, "Photonic architecture for reinforcement learning," *New Journal of Physics*, vol. 22, no. 4, p. 045 002, Apr. 2020. [Online]. Available: http://dx.doi.org/10.1088/1367-2630/ab783c.

[43] S. Li, Y. Xia, and Z. Xu, *Simultaneous perturbation stochastic approximation: Towards one-measurement per iteration*, 2022. [Online]. Available: https://arxiv.org/abs/2203.03075.

[44] J. C. Spall, "An overview of the simultaneous perturbation method for efficient optimization," *Johns Hopkins Apl Technical Digest*, vol. 19, pp. 482–492, 1998. [Online]. Available: https://api.semanticscholar.org/CorpusID:7988308.

[45] D. Carfi, A. Ricciardello, *et al.*, "Topics in game theory," *Applied Sciences-Monographs*, vol. 9, 2012.

[46] R. B. Myerson, "Refinements of the nash equilibrium concept," *International journal of game theory*, vol. 7, pp. 73–80, 1978.

[47] A. P. Flitney and D. Abbott, "An introduction to quantum game theory," *Fluctuation and Noise Letters*, vol. 2, no. 04, R175–R187, 2002.

[48] J. Eisert, M. Wilkens, and M. Lewenstein, "Quantum games and quantum strategies," *Phys. Rev. Lett.*, vol. 83, pp. 3077–3080, 15 Oct. 1999. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.83.3077.

[49] J. Du, X. Xu, H. Li, X. Zhou, and R. Han, *Playing prisoner's dilemma with quantum rules*, 2003. [Online]. Available: https://arxiv.org/abs/quant-ph/0301042.

[50] S. A. Fldzhyan, M. Y. Saygin, and S. P. Kulik, "Compact linear optical scheme for bell state generation," *Physical Review Research*, vol. 3, no. 4, p. 043 031, 2021.

[51] J. Eisert and M. Wilkens, "Quantum games," *Journal of Modern Optics*, vol. 47, pp. 2543–2556, 2000. [Online]. Available: https://api.semanticscholar.org/CorpusID:267804350.

[52] P. P. Angelov, E. A. Soares, R. Jiang, N. I. Arnold, and P. M. Atkinson, "Explainable artificial intelligence: An analytical review," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, vol. 11, no. 5, e1424, 2021.

[53] S. Ataman, *The quantum optical description of a double mach-zehnder interferometer*, 2014. [Online]. Available: https://arxiv.org/abs/1407.1704.

[54] Y. Wang, Z. Hu, B. C. Sanders, and S. Kais, "Qudits and high-dimensional quantum computing," *Frontiers in Physics*, vol. 8, p. 589 504, 2020.

[55] B. Seron, L. Novo, and N. J. Cerf, "Boson bunching is not maximized by indistinguishable particles," *Nature Photonics*, vol. 17, pp. 702–709, 2022. [Online]. Available: https://api.semanticscholar.org/CorpusID:247218409.

[56] B. Mondal and K. De, "An overview applications of graph theory in real field," *International Journal of Scientific Research in Computer Science, Engineering and Information Technology*, vol. 2, no. 5, pp. 751–759, 2017.

[57] J. Mills and R. Mezher, *Mitigating photon loss in linear optical quantum circuits: Classical postprocessing methods outperforming postselection*, 2024. [Online]. Available: https://arxiv.org/abs/2405.02278.

PLAGIARISM DECLARATION

I, Alexandre Ghislain BERGERAULT, certify that this document, untitled "Quantum Reinforcement Learning using Quantum Optical Projective Simulation" is an original work.

I certified that all sources used for this realization have been clearly referenced

I certify that have not copied or used ideas from books, papers or other source without precisely mentioning their origin.

I certify that this document have been produced on the basis of my work, in the context of my Master's Thesis at Quandela - Massy, France during the period March 2024 - July 2024.

Alexandre Ghislain BERGERAULT

At Gif-sur-Yvette, 91190, France

25/07/2024