Quantum speed-ups in reinforcement learning

PROCEEDINGS OF SPIE

SPIEDigitalLibrary.org/conference-proceedings-of-spie

Quantum speed-ups in reinforcement learning

Valeria Saggio, Beate Asenbeck, Arne Hamann, Teodor Strömberg, Peter Schiansky, et al.

Valeria Saggio, Beate E. Asenbeck, Arne Hamann, Teodor Strömberg, Peter Schiansky, Vedran Dunjko, Nicolai Friis, Nicholas C. Harris, Michael Hochberg, Dirk Englund, Sabine Wölk, Hans J. Briegel, Philip Walther, "Quantum speed-ups in reinforcement learning," Proc. SPIE 11806, Quantum Nanophotonic Materials, Devices, and Systems 2021, 118060N (1 August 2021); doi: 10.1117/12.2593720

Event: SPIE Nanoscience + Engineering, 2021, San Diego, California, United States

Quantum speed-ups in reinforcement learning

Valeria Saggio[a], Beate E. Asenbeck[a], Arne Hamann[b], Teodor Strömberg[a], Peter Schiansky[a], Vedran Dunjko[c], Nicolai Friis[d], Nicholas C. Harris[e], Michael Hochberg[f], Dirk Englund[g], Sabine Wölk[b,h], Hans J. Briegel[b,i], and Philip Walther[a,l]

[a]Vienna Center for Quantum Science and Technology (VCQ), Faculty of Physics, University of Vienna, Boltzmanngasse 5, A-1090 Vienna, Austria
[b]Institut für Theoretische Physik, Universität Innsbruck, Technikerstraße 21a, 6020 Innsbruck, Austria
[c]LIACS, Leiden University, Niels Bohrweg 1, 2333 CA Leiden, Netherlands
[d]Institute for Quantum Optics and Quantum Information - IQOQI Vienna, Austrian Academy of Sciences, Boltzmanngasse 3, A-1090 Vienna, Austria
[e]Lightmatter, 60 State Street, Boston, MA 02109, USA
[f]Nokia Corporation, New York, NY, USA
[g]Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, MA 02139, USA
[h]Deutsches Zentrum für Luft- und Raumfahrt e.V. (DLR), Institut für Quantentechnologien, Söflingerstr. 100, 89077 Ulm, Germany
[i]Fachbereich Philosophie, Universität Konstanz, Fach 17, 78457 Konstanz, Germany
[l]Christian Doppler Laboratory for Photonic Quantum Computer, Faculty of Physics, University of Vienna, Vienna, Austria

ABSTRACT

As the field of artificial intelligence is pushed forward, the question arises of how fast autonomous machines can learn. Within artificial intelligence, an important paradigm is reinforcement learning, where agents - learning entities capable of decision making - interact with the world they are placed in, called an environment. Thanks to these interactions, agents receive feedback from the environment and thus progressively adjust their behaviour to accomplish a given goal. An important question in reinforcement learning is how fast agents can learn to fulfill their tasks. To answer this question we consider a novel reinforcement learning framework where quantum mechanics is used. In particular, we quantize the agent and the environment and grant them the possibility to also interact quantum-mechanically, that is, by using a quantum channel for their communication. We demonstrate that this feature enables a speed-up in the agent's learning process, and we further show that combining this scenario with classical communication enables the evaluation of such an improvement. This learning protocol is implemented on an integrated re-programmable photonic platform interfaced with photons at telecommunication wavelengths. Thanks to the full tunability of the device, this platform proves the best candidate for the implementation of learning protocols, where a continuous update of the learning process is required.

Keywords: Quantum reinforcement learning, quantum speed-up, single photons, integrated optics

1. INTRODUCTION

The field of artificial intelligence (AI) is concerned with machines performing tasks that would normally require human intelligence. Of great interest within AI is reinforcement learning (RL), a paradigm that allows learning entities to improve their performance on the basis of obtained feedback.[1] In more detail, the framework of RL

Further author information: (Send correspondence to V.S.)
V.S.: E-mail: valeria.saggio@univie.ac.at

Quantum Nanophotonic Materials, Devices, and Systems 2021, edited by Cesare Soci, Matthew T. Sheldon, Mario Agio, Proc. of SPIE Vol. 11806, 118060N · © 2021 SPIE
CCC code: 0277-786X/21/$21 · doi: 10.1117/12.2593720

includes two essential elements: the agent - capable of decision-making - and its environment - the external world the agent is placed in. Whenever the agent makes the right decision (leading to an accomplishment of its task), the environment rewards its behaviour. This reward is used by the agent to increase the likelihood of performing well at the next round of interaction. In this way agents learn by reinforcement. RL is a type of dynamic learning that enables the development of intelligent agents capable of learning solely via interactions with their environment. A notable and deservedly celebrated example employing RL is the AlphaGo algorithm, able to outperform human skills by beating even the best human players at the game of Go.[2]

Over the past few years the idea of using quantum mechanics to boost the performance of RL protocols has been investigated. Noting that the internal structure of both the agent and the environment can be quantized, four different settings arise considering the agent-environment interaction: classical-classical (CC), classical-quantum (CQ), quantum-classical (QC), and quantum-quantum (QQ). The first scenario features no quantization and is therefore concerned with what briefly explained so far. In the CQ case the agent's structure is kept classical but the environment undergoes internal quantization. This leads to classical learning techniques aiding in quantum tasks, for example the design of new quantum experiments[3] or quantum control. The QC case is concerned with the quantization of the agent, which interacts classically with a classical environment. It was shown that this case leads to a quadratic speed-up in the agent's decision-making process.[4,5] This allows the agent to output moves faster, but does not lead to a reduction of the number of interaction steps required to fulfill the task. This last case can be achieved only in a fully quantum case, where both the agent and environment undergo quantization. In the following this fully-quantum case will be described, as it lays the foundation for the first demonstration of a quantum speed-up in the agent's learning time.[6]

2. QUANTIZING REINFORCEMENT LEARNING

In this section we will first describe RL in more formal terms, and will then present how to quantize this learning paradigm.

As already mentioned, two entities are essential to describe RL: an agent, provided with a goal to accomplish, and its environment. The idea of how RL works is conceptually depicted in Fig. 1. The key point is the agent-

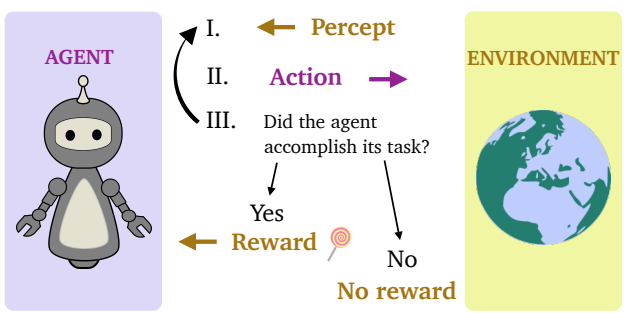

Figure 1. In RL an agent interacts with an environment and receives feedback signals that reinforce its learning process. Assuming the agent is provided with a certain task to fulfill, the environment issues an initial percept that triggers an action from the agent. If the output action leads to the accomplishment of the task, the environment sends the agent a reward that is used to learn. Otherwise, no reward is sent. In either case, the steps are repeated until the agent maximizes its obtained reward.

environment interaction, which can be described as a Markov decision process where certain moves called actions a_i performed by the agent (belonging to a discrete set $\mathcal{A}$), and certain states s_i of the environment (belonging to a discrete set $\mathcal{S}$) can be defined. Here the subscript i labels the time step. The agent can sense the environment's state s_i through some perceived information called a percept $c(s_i)$, and output a certain action a_i accordingly. After this step the environment will find itself in a new state s_{i+1}. Percepts trigger an action output according to the agent's policy $\pi(a_i|c(s_{i-1}))$, which defines a mapping from percepts to actions and therefore represents the control strategy of the agent. Here we assume a fully observable environment, which means that the percepts $c(s_i)$ coincide with the states s_i. Therefore, the agent's policy can be written as $\pi(a_i|s_{i-1})$. Whenever the agent

outputs the correct action a_i, the environment issues a scalar reinforcement signal called a reward r. If the agent does not perform well, no reward is given instead. In either case, after the environment evaluates the action of the agent, another round of interaction is started again. At every round, whenever a reward is given, the agent adjusts its behaviour - i.e. updates its policy π - in order to obtain maximal reward.

To construct a quantized RL framework, we start with promoting percepts s_i, actions a_i and rewards r to quantum states $|s_i\rangle$, $|a_i\rangle$ and $|r\rangle$, respectively. The percept set $\mathcal{S}$ and the action set $\mathcal{A}$ are now defined as Hilbert spaces $\mathcal{H}_\mathcal{S} = span\{|s_i\rangle\}$ and $\mathcal{H}_\mathcal{A} = span\{|a_i\rangle\}$, respectively, and form orthonormal bases.[7] This first step implies that agent and environment can now exchange quantum states, prepared in arbitrary quantum superpositions. Formally, the interaction between the agent and the environment can be described in a communication framework where the two entities use a channel for signal exchanges. Fig. 2 clarifies the difference between a classical and a quantum framework. In the classical case (Fig. 2a) percepts, actions and rewards can only belong to a fixed

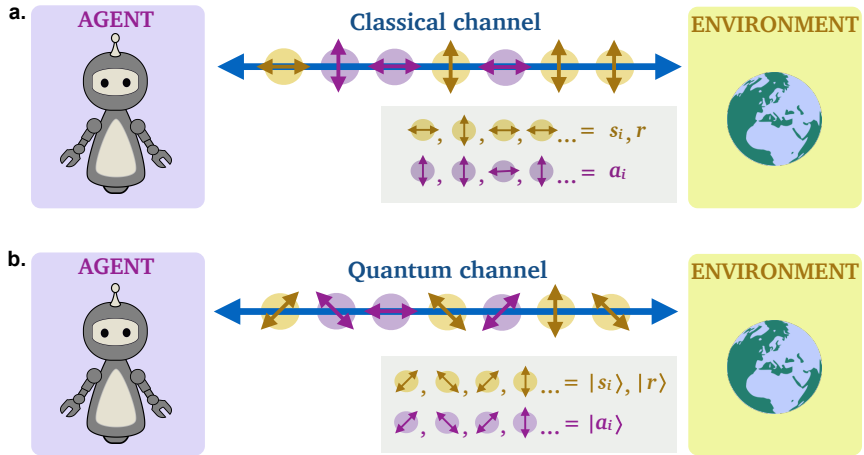

Figure 2. a. Agent and environment using a classical channel for their communication. Here signals can only be exchanged in a fixed preferred basis, for example horizontal/vertical photon polarization. b. Agent and environment using a quantum channel for their communication. Here signals can be exchanged in arbitrary quantum superpositions.

discrete alphabet, corresponding to a fixed basis when taking into account quantum systems (for example, the horizontal/vertical polarization of single photons). In this case, a classical channel is used for sending and receiving the signals. In the quantum case (Fig. 2b) the communication is no longer limited to a fixed basis, but allows for exchange of quantum percepts, actions and rewards in arbitrary quantum superpositions.

To prove the quantum speed-up in the agent's learning time, a few assumptions have to be made. First, we consider the so-called luck-favoring settings, where agents that initially receive many rewards by chance alone (lucky) will perform better than agents who initially obtain less rewards (unlucky).[7] Moreover, we consider the so-called deterministic strictly epochal environments (DSE), also referred to episodic instead of epochal.[1] In more detail, an environment having its initial state s_0 fixed and the percept s_i completely defined by all previous states and actions is called deterministic. This implies that the agent's policy $\pi(a_i|s_{i-1})$ can be written as $\pi(a_i|a_1, ..., a_{i-1})$. In addition, dealing with epochal environments means considering an interaction between agent and environment structured into epochs. Epochs consist of strings of percepts $\vec{s} = \{s_0, ..., s_{L-1}\}$ and actions $\vec{a} = \{a_1, ..., a_L\}$, both of fixed length L. During one epoch a fixed percept s_0 is given and L pairs of percepts and actions are exchanged. At the end of the epoch, the environment issues a reward r. After L percept-action pairs are exchanged, a new independent epoch is started again. Under these assumptions,[8] it is possible to define an environment whose behaviour can be described via a unitary U_E on the action and reward registers A and R as

$$U_\mathrm{E}|\vec{a}\rangle_\mathrm{A}|0\rangle_\mathrm{R} = \begin{cases} |\vec{a}\rangle_\mathrm{A}|1\rangle_\mathrm{R} & \text{if } r(\vec{a}) > 0 \\ |\vec{a}\rangle_\mathrm{A}|0\rangle_\mathrm{R} & \text{if } r(\vec{a}) = 0 \end{cases} . \tag{1}$$

Here a qubit ($|0\rangle_\mathrm{R}$ and $|1\rangle_\mathrm{R}$) is used to encode the reward and $r(\vec{a}) > 0$ and $r(\vec{a}) = 0$ indicate that the action sequence $\vec{a}$ is rewarded or non-rewarded, respectively. Therefore, the environment flips the reward state whenever

the agent couples out rewarded action sequences; else, the reward state is left unchanged. In this way rewarded sequences are identified by the environment.

Epochs can be classical or quantum. We will see how a speed-up in the learning time and its evaluation are possible if the agent alternately plays quantum and classical epochs.

In a classical epoch the agent determines the state $|\vec{a}\rangle_{\mathrm{A}}|0\rangle_{\mathrm{R}}$, where $|\vec{a}\rangle$ is obtained by sampling from $p(\vec{a})$, a classical probability distribution that is determined by the policy π. At the end of the epoch, the agent receives a reward r with a success probability

$$p_{\mathrm{win}} = \sin^2(\xi) = \sum_{\{\vec{a}|r(\vec{a})>0\}} p(\vec{a}), \tag{2}$$

where $\xi \in [0, 2\pi]$, and updates its policy according to a rule based on an RL model called projective simulation.[9]

A quantum epoch consists instead of the following three steps:

1. The agent prepares the state $|\psi\rangle_{\mathrm{A}}|-\rangle_{\mathrm{R}}$ and sends it to the environment. Here $|\psi\rangle_{\mathrm{A}} = \sum_{\vec{a}} \sqrt{p(\vec{a})}\,|\vec{a}\rangle_{\mathrm{A}} = \cos(\xi)|\ell\rangle_{\mathrm{A}} + \sin(\xi)|w\rangle_{\mathrm{A}}$, where $|w\rangle_{\mathrm{A}}$ and $|\ell\rangle_{\mathrm{A}}$ are superpositions of all winning (rewarded) and losing (non-rewarded) action sequences, respectively, and $|-\rangle_{\mathrm{R}} = (|0\rangle_{\mathrm{R}} - |1\rangle_{\mathrm{R}})/\sqrt{2}$.

2. The environment acts on $|\psi\rangle_{\mathrm{A}}|-\rangle_{\mathrm{R}}$ via U_{E} from Eq. (1), thus leading to

$$U_{\mathrm{E}}|\psi\rangle_{\mathrm{A}}|-\rangle_{\mathrm{R}} = [\cos(\xi)|\ell\rangle_{\mathrm{A}} - \sin(\xi)|w\rangle_{\mathrm{A}}]|-\rangle_{\mathrm{R}}. \tag{3}$$

   In this way it flips the sign of the winning state. It eventually sends the resulting state to the agent.

3. The agent performs a so-called reflection $U_{\mathrm{R}} = 2\,|\psi\rangle\langle\psi|_{\mathrm{A}} - \mathbb{1}_{\mathrm{A}}$ over the initial state $|\psi\rangle_{\mathrm{A}}$.

The implementation of these steps leads to an amplitude amplification of the winning state equal to the one that would be obtained in a Grover search for the winning states. In particular, the success probability to find rewarded action sequences $\sin^2(\xi)$ - given in Eq. (2) - increases to $\sin^2(3\xi)$.[10]

Quantum epochs allow for an increase of the success probability, however they do not reveal the reward for the chosen action sequence, and therefore cannot lead to an update of the agent's policy. The reward must be obtained classically, that is, by playing a classical test epoch where the obtained action sequence is used as input. By playing both a quantum and a classical test epoch, the success probability is first increased and the reward is eventually measured, leading to an update of the internal policy. In this sense, agents that alternately play quantum and classical epochs - also called hybrid agents - can accomplish their task of finding winning action sequences faster, and can therefore learn faster than agents limited to playing only classical epochs.

3. THE EXPERIMENTAL IMPLEMENTATION

This section describes the experimental demonstration of the quantum-enhanced RL protocol presented above. We used a particular type of integrated photonic processor, compact and fully programmable.[11] Especially thanks to its programmability, this device is particularly suitable for the implementation of learning tasks, which generally require periodic updates. Moreover, the processor interfaces with photons at telecommunication wavelengths, which feature very low loss compared to the more standard wavelength of 800 nm, and therefore would allow an implementation of learning tasks even over long distances.

An image of the processor is given in Fig. 3a. It is composed of 26 waveguides interconnected to form 88 Mach-Zehnder interferometers (MZIs). Fig. 3b shows one MZI equipped with two programmable phase shifters, one internal allowing for a scan of the output distribution over $\theta \in [0, 2\pi]$ and one external defining the relative phase $\phi \in [0, 2\pi]$ between the two output modes. Such an MZI acts as a tunable beam splitter, implementing the unitary

$$U_{\theta,\phi} = \begin{pmatrix} e^{i\phi}\sin\frac{\theta}{2} & e^{i\phi}\cos\frac{\theta}{2} \\ \cos\frac{\theta}{2} & -\sin\frac{\theta}{2} \end{pmatrix}. \tag{4}$$

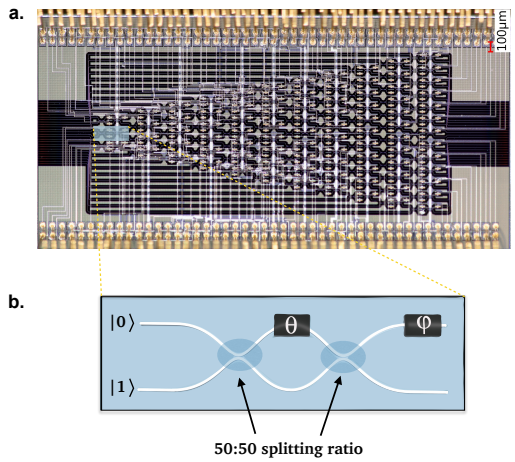

Figure 3. a. Image of the processor showing its layout, where the arrangement of the 88 MZIs in a trapezoidal configuration is visible. b. Single MZI equipped with an internal and an external tunable phase shifter.

The information, carried by single photons, is spatially encoded onto two orthogonal modes, $|0\rangle = (1,0)^{\mathrm{T}}$ and $|1\rangle = (0,1)^{\mathrm{T}}$, constituting the computational basis.

Quantum and classical epochs are implemented on this processor. To do this, the first step is to couple single photons into the chip. Single photons at telecommunication wavelength are produced in pair by a single-photon source pumped by continuous-wave laser light, as shown in Fig. 4. One photon from the pair is coupled into the

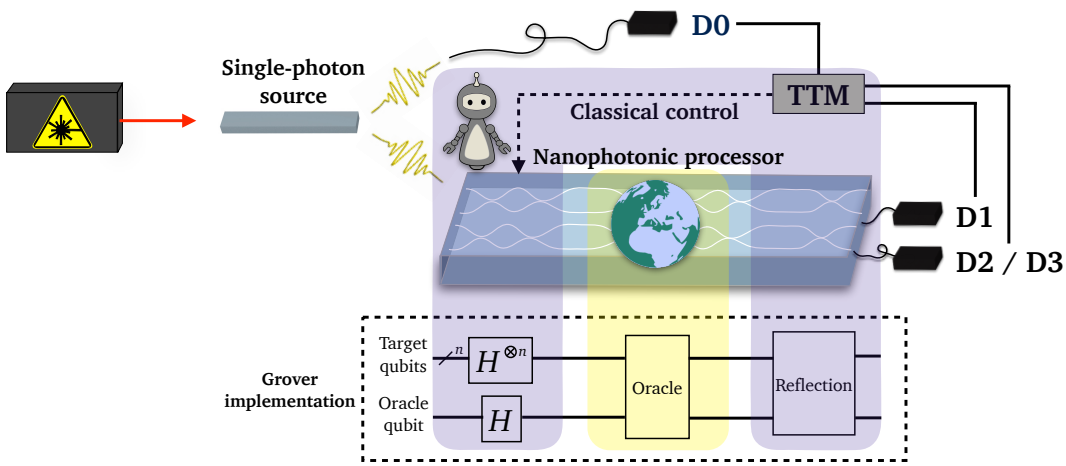

Figure 4. Laser light at 789.75 nm pumps a single-photon source, which produces pairs of single photons at telecommunication wavelength. One photon from the pair is sent into the processor to perform the computation, while the other one is used to herald its presence. A TTM records coincidence events between clicks in D0 and D1/D2/D3. Specific regions of the processor are assigned to the agent and the environment, which implement the classical or quantum strategy. In the quantum strategy, a Grover search for the winning action sequences is performed. Furthermore, the agent is equipped with a classical control part that is used for policy update.

processor to perform the desired computation, and can trigger clicks in single-photon detectors D1, D2 or D3 (according to the specific circuit implementation). The second photon from the pair is sent to a single-photon detector D0 for detecting the presence of its twin. Clicks between D0 and D1/D2/D3 that fall in a temporal window of 1.3 ns are meaningful detection events indicated as the coincidence events. They are recorded with a time tagging module (TTM), which outputs the temporal information about the photons. Different areas of the processor are assigned to the agent and the environment, which can thus either implement a classical epoch

or play a Grover-like search following the quantum steps 1-3 listed in the previous section. The policy update is implemented via a feedback loop over which the agent has full control.

For implementing the experiment, all possible action sequences are grouped into winning and losing sequences $|\text{win}\rangle_\text{A}$ and $|\text{lose}\rangle_\text{A}$, and the states $|1\rangle_\text{A}$ and $|0\rangle_\text{A}$ are used to encode them, respectively. Another qubit ($|1\rangle_\text{R}$, $|0\rangle_\text{R}$) encodes the reward. This results in a four-level system ($|0_\text{A}1_\text{R}\rangle$, $|0_\text{A}0_\text{R}\rangle$, $|1_\text{A}0_\text{R}\rangle$, $|1_\text{A}1_\text{R}\rangle$) represented by four spatial modes in the processor. Figs. 5a and b show these four spatial modes and depict how a classical and a quantum epoch are implemented. In both cases, a single photon is input in the spatial mode $|0_\text{A}0_\text{R}\rangle$.

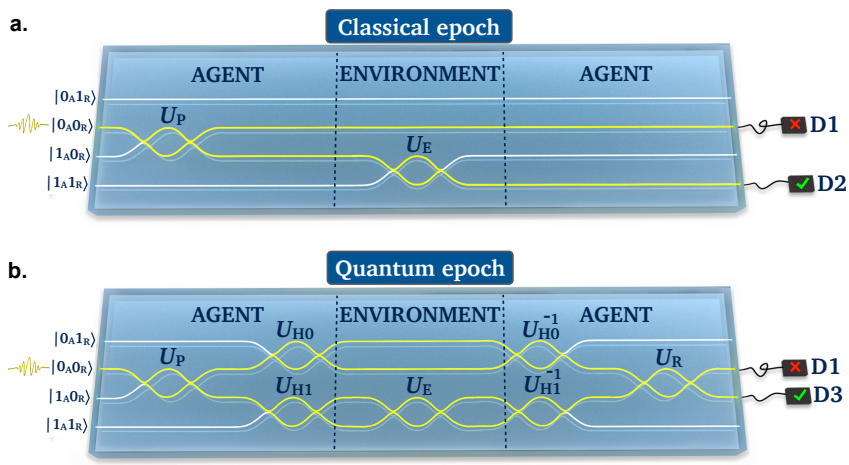

Figure 5. a. Implementation of a classical epoch. b. Implementation of a quantum epoch. The yellow waveguides highlight the photon's possible paths and the straight waveguides represent Identity gates.

In a classical epoch, the agent performs the unitary U_P, which transforms the state $|0_\text{A}0_\text{R}\rangle$ into

$$|\psi\rangle = \sqrt{p_\text{win}}|1_\text{A}0_\text{R}\rangle + \sqrt{1-p_\text{win}}|0_\text{A}0_\text{R}\rangle, \tag{5}$$

or equivalently $|\psi\rangle = \sin(\xi)|1_\text{A}0_\text{R}\rangle + \cos(\xi)|0_\text{A}0_\text{R}\rangle$, parametrising p_win with $\sin^2(\xi)$ as shown in Eq. (2). In this way, a superposition of the winning and losing action sequences is created. The success probability p_win is initially set to 0.01 (this corresponds to considering a single winning action sequence out of 100). Next, the environment swaps the photon's mode in the case of the winning sequence - thus issuing a reward - while it leaves the losing sequence unchanged. The photon will eventually be found either in detector D2 with probability $p_\text{win} = \sin^2(\xi)$ (initially set to 0.01) or in detector D1 with probability $1 - 0.01 = 0.99$. If a coincidence event is found between D2 and D0, the agent obtains a reward and can therefore update p_win and consequently its policy π. If a coincidence event is detected between D1 and D0, no reward is obtained and therefore no update takes place. In either case, another epoch is played until the success probability is maximized. After the agent finds j rewards, its success probability reads

$$p_{\text{win}_j} = \frac{1+2j}{100+2j}, \tag{6}$$

which represents the specific update rule for this implementation. Playing only classical epochs constitutes the so-called classical strategy.

In a quantum epoch, analogously to the classical case, the agent first prepares the superposition state in Eq. (5) via U_P. Then two unitaries U_H0 and U_H1 are used to rotate the reward state $|0\rangle_\text{R}$ to $|-\rangle_\text{R} = \frac{|0\rangle_\text{R} - |1\rangle_\text{R}}{\sqrt{2}}$. Now it is the turn of the environment, which acts as an oracle via U_E and flips the sign of the winning sequence. The losing sequence is left unchanged. Next the agent reverses the two unitaries U_H0 and U_H1 and performs the reflection U_R over the initial distribution. These steps follow the three theoretical steps given in the previous section. Whenever D3 clicks (in coincidence with D0), the winning action sequence is detected with an amplified probability $p_\text{win}(3-4p_\text{win})^2$, or equivalently $\sin^2(3\xi)$, which corresponds to the success probability after one Grover step.

However, no information about the reward is gained at this stage (the reward still finds itself in the state $|0\rangle_{\mathrm{R}}$). This information is obtained classically by using the detected action sequence as input state for a classical test epoch. Whenever a reward is obtained, the same update rule as in the classical strategy is used. Note that for practical reasons the classical test epoch has been implemented in software only. Alternating quantum and classical test epochs constitutes the so-called quantum strategy, which is relevant for agents with the goal of learning faster.

However, any Grover-like algorithm experiences a drop in amplitude amplification after reaching the optimal point (that maximizes the probability of finding the desired state). For this reason, agents implement the quantum strategy only until they receive on average more rewards than with the classical strategy, and start playing a fully classical strategy from that point on. The probability p_{win} at which the switch occurs can be found by considering the average reward η for both the quantum and classical strategies, and finding the value of p_{win} at which the classical strategy becomes more advantageous than the quantum strategy. The average reward η in the quantum strategy is

$$\eta_{\mathrm{Q}} = \frac{p_{\mathrm{win}}(3 - 4p_{\mathrm{win}})^2}{2}, \tag{7}$$

where the numerator is the amplified success probability after one Grover step. The factor 2 in the denominator accounts for the fact that two epochs (quantum and classical test) must be played to obtain a reward. In the classical case, one has instead

$$\eta_{\mathrm{C}} = p_{\mathrm{win}}. \tag{8}$$

It is easy to see that the agents only benefit from a quantum strategy as long as $p_{\mathrm{win}} < 0.396$. Whenever they reach a probability of 0.396, indicated by P from now on, they switch to a completely classical strategy. This (quantum + classical) strategy will be referred to as combined strategy. By playing a combined strategy, such hybrid agents can always gain an advantage over their purely classical counterparts.

4. RESULTS

Here we show the experimental comparison between the classical and combined strategies. During the classical strategy, outcomes 1 and 0 (corresponding to the rewarded and non-rewarded behaviour, respectively) are recorded at the end of each epoch, forming a sequence of length equal to the number of played epochs. In the quantum strategy, a sequence of half length is obtained instead. A fair comparison between the two strategies is guaranteed by averaging the reward obtained in the quantum strategy over the two quantum and classical test epochs. For both strategies, the obtained reward is eventually averaged over many different agents playing independently of one another. A comparison between the combined and the classical strategies is shown in Fig. 6. The theoretical data, represented by the solid lines, is simulated for n = 10,000 agents, while n=165 agents have been used to obtain the experimental data. The orange and green curves represent the average reward η in the classical and combined case, respectively, versus the number of even epochs. The initial steep part of the green curve, better visualized in the inset, shows the quantum improvement originating from the use of amplitude amplification, in comparison with a purely classical strategy (orange curve). As soon as $p_{\mathrm{win}} = 0.396 = P$ is reached, the agents switch to a classical strategy. In this way, agents playing a combined strategy always perform better than purely classical agents. The improvement in the learning time can now be quantified. Defining the learning time $\langle T \rangle$ as the average number of epochs necessary to achieve a certain predefined success probability P_{L} (smaller than P), a reduction of 63% is achieved choosing $P_{\mathrm{L}} = 0.37$. In more detail, a learning time of $\langle T \rangle_{\mathrm{C}} = 270$ in the classical strategy is reduced to $\langle T \rangle_{\mathrm{Q}} = 100$ when a quantum (or combined) strategy is played. These values are in good agreement with the theoretical values $T_{\mathrm{C}}^{\mathrm{theory}} = 293$ and $T_{\mathrm{Q}}^{\mathrm{theory}} = 97$, considering small experimental imperfections (see Appendix A for more details about the learning time and the theoretical values). In general, a quadratic speed-up in the learning time can be obtained if an arbitrary number of Grover iterations can be played.[8]

5. CONCLUSIONS

We have investigated the benefits of using quantum mechanics in combination with a fully quantized RL framework, where the agent and environment can interact quantum-mechanically. In particular, we have demonstrated

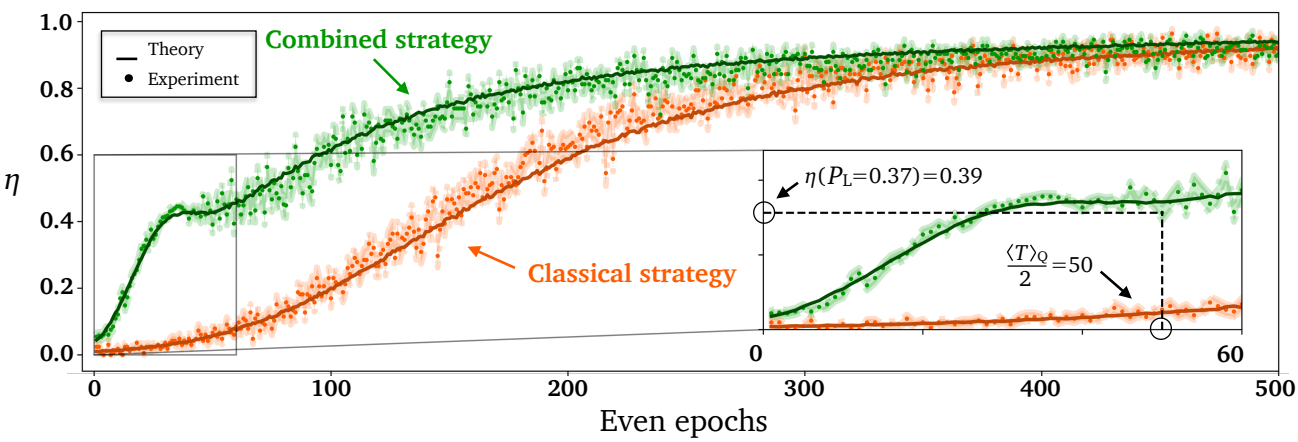

Figure 6. Behaviour of the average reward η for the classical and combined strategies. The orange (green) solid curve reproduces the theoretical simulations of the average reward η for the classical (combined) strategy. The dots correspond to experimental data, to which the standard errors are associated (shaded regions). The inset shows the region where the switch from the quantum to the classical strategy occurs. The predefined success probability $P_L = 0.37$ defines a learning time of $\langle T\rangle_Q = 100$.

for the first time that a speed-up in the learning time is possible. The experiment was carried out using an integrated photonic platform interfaced with telecommunication-wavelength photons. Recently, the development of fully programmable integrated photonic circuits[12] has been considered very promising for the implementation of RL protocols, which require a continuous update process. Additionally, interfacing this device with telecommunication-wavelength photons opens the possibility to integrate RL speed-ups in future quantum networks, where fast feedback loops over long distances would be required. Dealing with a combinatorially large space of action sequences of arbitrary length would require multiple photons for future scaled-up implementations.

APPENDIX A. THE LEARNING TIME

After the agent has observed j rewards, indicating by t_j the number of epochs to find the next rewarded action sequence $\vec{a}_{j+1}$, its learning time T can be written as

$$T = \sum_{j=0}^{J-1} t_j, \tag{9}$$

where J is the number of rewards needed to reach a certain predefined success probability P_L (smaller than $P = 0.396$). In a purely classical case, the number of epochs necessary to find the sequence $\vec{a}_{j+1}$ is, on average,

$$\langle t_j\rangle_C = \frac{1}{p_{\mathrm{win}_j}}. \tag{10}$$

If an arbitrary number of Grover steps can be implemented, an agent playing quantum-mechanically instead needs, on avergae, a number of epochs equal to[10,13]

$$\langle t_j\rangle_Q = \frac{\alpha}{\sqrt{p_{\mathrm{win}_j}}}, \tag{11}$$

where α is a constant that depends on the number of epochs needed to create one oracle query[7] and on whether p_{win} is known. Therefore, the average learning time in the case where quantum amplification is used reads

$$\langle T\rangle_Q = \sum_{j=0}^{J-1}\langle t_j\rangle_Q = \sum_{j=0}^{J-1}\frac{\alpha}{\sqrt{p_{\mathrm{win}_j}}} \leq \alpha\sqrt{J}\sqrt{\sum_{j=0}^{J-1}\frac{1}{p_{\mathrm{win}_j}}} = \alpha\sqrt{J\ \langle T\rangle_C}, \tag{12}$$

where Eqs. (9), (10), and the Cauchy-Schwarz inequality have been used. It is therefore clear that a quadratic advantage in the learning time is achieved if an arbitrary number of Grover iterations are played.

However, near-term quantum devices only allow for limited coherent evolutions in general, and are therefore limited to a maximal number $k_{\max}$ of Grover steps. The learning time can be thus defined as

$$\langle T \rangle_{\mathrm{Q}} = \sum_{j=0}^{J-1} \frac{\alpha_{\mathrm{o}} k_{\max} + 1}{\sin^2[(1 + 2k_{\max})\xi_j]}, \tag{13}$$

where α_{o} is the number of epochs required to create an oracle query and the denominator is the amplified success probability after $k_{\max}$ Grover steps. The term $+1$ in the numerator accounts for the classical test epoch. From Eq. (13), considering $k_{\max} \gg 1$ and $(1 + 2k_{\max})\xi_J \ll \pi/2$, and using $\sin(x) \approx x$ for $x \ll 1$, it is easy to see that

$$\langle T \rangle_{\mathrm{Q}} \approx \alpha_{\mathrm{o}} \frac{\langle T \rangle_{\mathrm{C}}}{4k_{\max}}, \tag{14}$$

which defines a linear improvement.

The theoretical values of the learning time $T_{\mathrm{C}}^{\mathrm{theory}}$ and $T_{\mathrm{Q}}^{\mathrm{theory}}$ given in the Results section are found by first extracting the number of rewards J needed to obtain $P_{\mathrm{L}} = 0.37$ from Eq. (6) (setting p_{win_j} to 0.37). A value of $j = J = 29$ is thus obtained. Now Eq. (10) together with (9), and Eq. (13) are used to obtain the classical and quantum learning time, respectively. In Eq. (13) $\alpha_{\mathrm{o}} = 1$ in our case. One thus finds $\langle T \rangle_{\mathrm{C}}^{\mathrm{theory}} = 293$ and $\langle T \rangle_{\mathrm{Q}}^{\mathrm{theory}} = 97$.

ACKNOWLEDGMENTS

The authors thank Lee A. Rozema, Irati Alonso Calafell, and Philipp Jenke for help with the detectors. A.H. acknowledges support from the Austrian Science Fund (FWF) through the project P 30937-N27. V.D. acknowledges support from the Dutch Research Council (NWO/OCW), as part of the Quantum Software Consortium programme (project number 024.003.037). N.F. acknowledges support from the Austrian Science Fund (FWF) through the project P 31339-N27. H.J.B. acknowledges support from the Austrian Science Fund (FWF) through SFB BeyondC F71. P.W. acknowledges support from the research platform TURIS, the European Commission through ErBeStA (No. 800942), HiPhoP (no. 731473) and UNIQORN (no. 820474), from the Austrian Science Fund (FWF) through CoQuS (W1210-4), BeyondC (F7113-N48) and NaMuG (P30067-N36), AFOSR via QAT4SECOMP (FA2386-17-1-4011), and Red Bull GmbH. The MIT portion of the work was supported in part by AFOSR award FA9550-16-1-0391.

REFERENCES

[1] Sutton, R. S. and Barto, A. G., [*Reinforcement Learning: An Introduction*], MIT press, Cambridge (1998).

[2] Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., Van den Driessche, G., Graepel, T., and Hassabis, D., "Mastering the game of Go without human knowledge," *Nature* **550**, 354–359 (2017).

[3] Melnikov, A. A., Poulsen Nautrup, H., Krenn, M., Dunjko, V., Tiersch, M., Zeilinger, A., and Briegel, H. J., "Active learning machine learns to create new quantum experiments," *Proc. Natl. Acad. Sci. U.S.A.* **115**, 1221–1226 (2018).

[4] Paparo, G. D., Dunjko, V., Makmal, A., Martin-Delgado, M. A., and Briegel, H. J., "Quantum Speedup for Active Learning Agents," *Phys. Rev. X* **4**, 031002 (2014).

[5] Sriarunothai, T., Wölk, S., Giri, G. S., Friis, N., Dunjko, V., Briegel, H. J., and Wunderlich, C., "Speeding-up the decision making of a learning agent using an ion trap quantum processor," *Quantum Science and Technology* **4**(1), 015014 (2018).

[6] Saggio, V., Asenbeck, B. E., Hamann, A., Strömberg, T., Schiansky, P., Dunjko, V., Friis, N., Harris, N. C., Hochberg, M., Englund, D., et al., "Experimental quantum speed-up in reinforcement learning agents," *Nature* **591**(7849), 229–233 (2021).

[7] Dunjko, V., Taylor, J. M., and Briegel, H. J., "Quantum-enhanced machine learning," *Physical review letters* **117**(13), 130501 (2016).

[8] Hamann, A., Dunjko, V., and Wölk, S., "Quantum-accessible reinforcement learning beyond strictly epochal environments," *arXiv preprint arXiv:2008.01481* (2020).

[9] Briegel, H. J. and De las Cuevas, G., "Projective simulation for artificial intelligence," *Scientific reports* **2**(1), 1–16 (2012).

[10] Grover, L. K., "Quantum mechanics helps in searching for a needle in a haystack," *Phys. Rev. Lett.* **79**, 325 (Jul 1997).

[11] Harris, N. C., Steinbrecher, G. R., Prabhu, M., Lahini, Y., Mower, J., Bunandar, D., Chen, C., Wong, F. N., Baehr-Jones, T., Hochberg, M., et al., "Quantum transport simulations in a programmable nanophotonic processor," *Nature Photonics* **11**(7), 447 (2017).

[12] Bogaerts, W., Pérez, D., Capmany, J., Miller, D. A., Poon, J., Englund, D., Morichetti, F., and Melloni, A., "Programmable photonic circuits," *Nature* **586**(7828), 207–216 (2020).

[13] Boyer, M., Brassard, G., Høyer, P., and Tapp, A., "Tight bounds on quantum searching," *Fortschritte der Physik: Progress of Physics* **46**(4-5), 493–505 (1998).