Cyber-physical defense in the quantum Era

Michel Barbeau, Joaquin Garcia-alfaro

Cyber-Physical Defense in the Quantum Era

Michel Barbeau*[1] and Joaquin Garcia-Alfaro*[2]

[1]Carleton University, School of Computer Science, Canada.
[2]Institut Polytechnique de Paris, Télécom SudParis, France.
*Correspondence to barbeau@scs.carleton.ca and garcia_a@telecom-sudparis.eu

ABSTRACT

Networked-Control Systems (NCSs), a type of cyber-physical systems, consist of tightly integrated computing, communication and control technologies. While being very flexible environments, they are vulnerable to computing and networking attacks. Recent NCSs hacking incidents had major impact. They call for more research on cyber-physical security. Fears about the use of quantum computing to break current cryptosystems make matters worse. While the quantum threat motivated the creation of new disciplines to handle the issue, such as post-quantum cryptography, other fields have overlooked the existence of quantum-enabled adversaries. This is the case of cyber-physical defense research, a distinct but complementary discipline to cyber-physical protection. Cyber-physical defense refers to the capability to detect and react in response to cyber-physical attacks. Concretely, it involves the integration of mechanisms to identify adverse events and prepare response plans, during and after incidents occur. In this paper, we make the assumption that the eventually available quantum computer will provide an advantage to adversaries against defenders, unless they also adopt this technology. We envision the necessity for a paradigm shift, where an increase of adversarial resources because of quantum supremacy does not translate into higher likelihood of disruptions. Consistently with current system design practices in other areas, such as the use of artificial intelligence for the reinforcement of attack detection tools, we outline a vision for next generation cyber-physical defense layers leveraging ideas from quantum computing and machine learning. Through an example, we show that defenders of NCSs can learn and improve their strategies to anticipate and recover from attacks.

Keywords: Cyber-Physical System, Cyber-Physical Attack, Networked-Control System, Quantum Machine Learning, Artificial Intelligence, Machine Learning, Quantum Computing, Quantum Information.

1 Introduction

Networked-Control Systems (NCSs) integrate computation, communications and physical processes. Their design involves fields such as computer science, automatic control, networking and distributed systems. Physical resources are orchestrated building upon concepts and technologies from these domains. In a Networked-Control System (NCS), the focus is on remote control, which means steering at distance a dynamical system according to requirements. Determined according to a target behavior, feedback and corrective control actions are transported over a communication network.

In a NCS, networks and systems, sometimes called the plant, represent observable and controllable physical resources. The sensors correspond to observation apparatus. The actuators represent an abstraction of devices enabling the control of the networked system. Using signals produced by the sensors, the controller generates commands to the actuators. The coupling of the controller with actuators and sensors happens through a communications network. In contrast to a classical feedback-control system, NCSs provide remote control.

NCSs are flexible, but vulnerable to computer and network attacks. Adversaries build upon their knowledge about dynamics, feedback predictability and countermeasures, to perpetrate attacks with severe implications[1–3]. When industrial systems and national infrastructures are victimized, consequences are catastrophic for businesses, governments and society[4]. A growing number of incidents have been documented. Representative instances are listed in Box 1.

Attacks can be looked into from several point of views[5]. We can consider attacks in relation to an adversary knowledge about a system and its defenses. In addition, we can consider attacks with respect to the criticality of disrupted resources. For example, a Denial of Service (DoS) attack targeting an element that is crucial to operation[6]. Besides, we can take into account the ability of an adversary to analyze signals, such as sensor outputs. This may enable sophisticated attacks impacting system integrity or availability. Moreover, there are incidents caused by human adversarial actions. For instance, they may forge feedback for disruption purposes. NCSs must be capable of handling security beyond breach. In other words, they must assume that cyber-physical attacks will happen. They should be equipped with cyber-physical defense tools. For instance, response management tools must assure that crucial operational functionality are be properly accomplished and cannot be

stopped. For example, the cooling service of a nuclear plant reactor or safety control of an autonomous navigation system are crucial functionalities. Other less important functionalities may be temporarily stopped or partially completed, such as a printing service. It is paramount to assure that defensive tools provide appropriate responses, to rapidly take back control when incidents occur.

That being said, the quantum paradigm will render obsolete a number of cyber-physical security technologies. Solutions that are assumed to be robust today will be deprecated by quantum-enabled adversaries. For example, adversaries that will capable of brute-forcing and taking advantage of the upcoming quantum computing power. Disciplines, such as cryptography, are addressing this issue. Novel post-quantum cryptosystems are facing the quantum threat. Other fields, however, have overlooked the eventual existence of quantum-enabled adversaries. Cyber-physical defense, a discipline complementary to cryptography, is a proper example. It uses artificial intelligence mainly to detect anomalies and anticipate adversaries. Hence, it enables NCSs with capabilities to detect and react in response to cyber-physical attacks. More concretely, it involves the integration of machine learning to identify adverse events and prepare response plans, while and after incidents occur. An interesting question is the following. How a defender will face a quantum-enabled adversary? How can a defender use the quantum advantage to anticipate response plans? How to ensure cyber-physical defense in the quantum era? In this paper, we investigate these questions. We develop foundations of a quantum machine learning defense framework. Through an illustrative example, we show that a defender can leverage quantum machine learning to address the quantum challenge. We also highlight some recent methodological and technological progress in the domain and remaining issues.

Box 1 - Representative cyber-physical attacks documented in the media.

Espionage and sabotage of critical facilities, such as US data breach in 2021 due to the SolarWinds attack or attempts of Saudi Aramco cyber-sabotage of oil-processing facilities in 2020. Similar problems are spanning worldwide.	**Adversarial actions in this scenario**, include USB injection of corrupted software binaries, drive-by-download malware installation, spear phishing-based design of websites, and traditional social engineering manipulation of critical infrastructure employees.
Remote control of navigation systems, including successful hacking of autonomous cars and avionic systems. Studies and general concern started with a malware that infected over sixty thousand computers of an Iranian nuclear facility.	**Adversarial actions** include the use of infection vectors (e.g., USB drives), corrupted updates and patches, radio frequency jamming, radio frequency spoofing, and software binary manipulations.
Disruptions of large-scale industries have been appointed by the Federal Office for Information Security of Germany as a serious concern to European factory and industrial markets. Similar threats affect drones and smart cities, as well.	**Adversarial actions** include the use of GNSS (Global Navigation Satellite Systems) attacks, e.g., jamming of signals, spoofing and hijacking of communications to downgrade communications to insecure modes (e.g., from encrypted to plain-text communications).

The remaining sections are organized as follows. Section 2 reviews related work. Section 3 develops our approach, exemplified with a proof-of-concept. Section 4 discusses the generalization of the approach and open problems. Section 5 concludes the paper.

2 Related Work

Protection is one of the most important branches of cybersecurity. It mainly relies on the implementation of state-of-the-art cryptographic protocols. They mainly comprise the use of encryption, digital signatures and key agreement. The security of some cryptographic families are based on computational complexity assumptions. For instance, public key cryptography builds upon factorization and discrete logarithm problems. They assume the lack of efficient solutions that break them in polynomial time. However, quantum enabled adversaries can invalidate these assumptions. They put those protocols at risk[7,8]. At the same time, the availability of quantum computers from research to general purpose applications led to the creation of new cybersecurity disciplines. The most prominent one is Post-Quantum Cryptography (PQC). It is a fast growing research topic aiming to develop new public key cryptosystems resistant to quantum enabled adversaries.

The core idea of PQC is to design cryptosystems whose security rely on computational problems that cannot be resolved by quantum adversaries in admissible time. Candidate PQC families include code-based[9], hash-based[10], multivariate[11], lattice-based[12,13] and supersingular isogeny-based[14] cryptosystems. Their security is all based on mathematical problems that are believed to be hard, even with quantum computation and communications resources[15]. Furthermore, PQC has led to new research directions driven by different quantum attacks. For instance, quantum-resistant routing aims at achieving a secure and sustainable quantum-safe Internet[16].

Besides, quantum-enabled adversaries can disrupt the operation of classical systems. For example, they can jeopardize availability properties by perpetrating brute-force attacks. Solidifying the integrity and security of the quantum Internet is of chief importance. Solutions to these challenges are being developed and published in the quantum security literature using multilevel security stacks. They involve the combination of quantum and classical security tools[17]. Cybersecurity researchers emphasized the need for more works on approaches mitigating the impact of such attacks[18]. Following their detection, adequate response to attacks is a problem that seems to have received little attention. Specially when we are dealing with quantum enabled adversaries. Intrusion detection, leveraging artificial intelligence and machine learning, is the most representative category of the detection and reaction paradigm.

The detection and reaction paradigm uses adversarial risk analysis methodologies, such as attack trees[20] and graphs[21]. Attacks are represented as sequences of incremental steps. The last steps of sequences correspond to events detrimental to the system. In other words, an attack is considered successful when the adversary reaches the last step. The cost for the adversary is quantified in terms of resource investment. It is generally assumed that with infinite resources, an adversary reaches an attack probability success of one. For instance, infinite resources can mean usage of brute-force[22]. An adversary that increases investment, such as time, computational power or memory, also increases the success probability of reaching the last step of an attack. Simultaneously, this reduces the likelihood of detection by defenders. Analysis tools may help to explore the relation between adversary investment and attack success probability[23]. Figure 1 schematically depicts the idea. The horizontal axis represents the cost of the adversary in terms of resource investment. The vertical axis represents the success probability of the attack. We depict three scenarios. The blue curve involves a classical adversary with classical resources and a relatively low probability of attack success. The red curve corresponds to a quantum-enabled adversary, classical defender scenario. The adversary has the quantum advantage with relatively high probability of attack success. The black curve represents a balanced situation, where both the adversary and defender have quantum resources. Every curve models a Cumulative Distribution Function (CDF) corresponding the probability of success versus the adversary resource investment. Distribution functions such as Rayleigh[24] and Rician[25] are commonly used in the intrusion detection literature for this purpose. Their parameters can be estimated via empirical penetration testing tools[26]. Without empowering defenders with the same quantum capabilities, an increase of adversarial resources always translate into a higher likelihood of system disruption. In the sequel, we discuss how to equip defenders with quantum resources such that a high attack success probability is not attainable anymore.

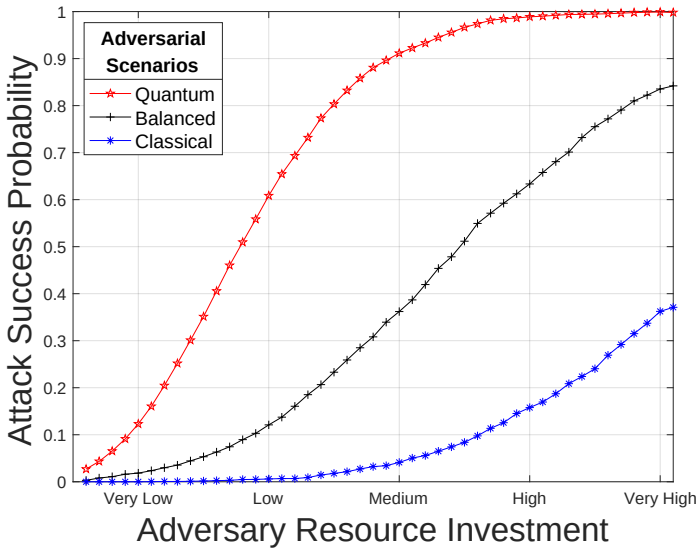


Figure 1. Attack success probability vs. adversary investment. We consider three adversarial scenarios. Classical (blue curve), where the resources of the adversary are lower than the resources of the defender. Balanced (black), where the resources of the defender are proportional to those of the adversary. Quantum (red), where the resources of the adversary are higher than those of the defender. Simulation code is available at our companion website, in the Matlab folder[19].

3 Cyber-Physical Defense using Quantum Machine Learning

Machine Learning (ML) is about data and, together with clever algorithms, building experience such that next time the system does better. The relevance of ML to computer security in general has already been given consideration. Chio and Freeman[27] demonstrated general applications of ML to enhance security. A success story is the use of ML to control spam emails metadata, source reputation, user feedback and pattern recognition are combined to filter out junk emails. Furthermore, there is an evolution capability. The filter gets better with time. This way of thinking is relevant to Cyber-Physical System (CPS) security because its defense can learn from attacks and make the countermeasures evolve. Focusing on CPS-specific threats, as an example pattern recognition can be used to extract from data the characteristics of attacks and to prevent them in the future. Because of its ability to generalize, ML can deal with adversaries hiding by varying the exact form taken by their attacks. Note that perpetrators can adopt as well the ML paradigm to learn defense strategies and evolve attack methods. The full potential of ML for CPS security has not been fully explored. The way is open for the application of ML in several scenarios. Hereafter, we focus on using Quantum Machine Learning (QML) for cyber-physical defense.

QML, i.e., the use of quantum computing for ML[28], has potential because the time complexity of tasks such as classification is independent of the number of data points. Quantum search techniques are data size independent. There is also the hope that the quantum computer can learn things that the classical computer is incapable of, due to the fact that the former has properties that the latter does not have, notably entanglement. At the outset, however, we must admit that a lot remains to be discovered.

QML is mainly building on the traditional quantum circuit model. Schuld and Killoran investigated the use of kernel methods[29], employed for system identification, for quantum ML. Encoding of classical data into a quantum format is involved. A similar approach has been proposed by Havlíček et al.[30]. Schuld and Petruccione[31] discuss in details the application of quantum ML over classical data generation and quantum data processing. A translation procedure is required to map the classical data, i.e., the data points, to quantum data, enabling quantum data processing, such as quantum classification. However, there is a cost associated with translating classical data into the quantum form, which is comparable to the cost of classical ML classification. This is right now the main barrier. The approach resulting in real gains is quantum data generation and quantum data processing, since there is no need to translate from classical to quantum data. Quantum data generation requires quantum sensing.

Successful implementation of this approach will grant a quantum advantage, to the adversary or CPS defenders. There are alternatives to doing QML with traditional quantum circuits. Use of tensor networks[32], a general graph model, is one of them[33].

Next, we develop an example that illustrates the potential and current limitations of quantum ML, using variational quantum circuits[31,34,35], for solving cyber-physical defense issues.

3.1 Approach

Let us consider the adversarial model represented in Figure 2. There is a controller getting data and sending control signals through networked sensors and actuators to a system. An adversary can intercept and tamper signals exchanged between the environment and controller, in both directions. Despite the perpetration of attacks, the controller may still have the ability to monitor and steer the system. This is possible using redundant sensors and actuators attack detection techniques. This topic has been addressed in related work[36]. Furthermore, we assume that:

1. the controller has options and can independently make choices,
2. the adversaries have options and can independently make choices and
3. the consequences of choices made by the controller, in conjunction with those made by adversaries, can be quantified, either by a penalty or a reward.

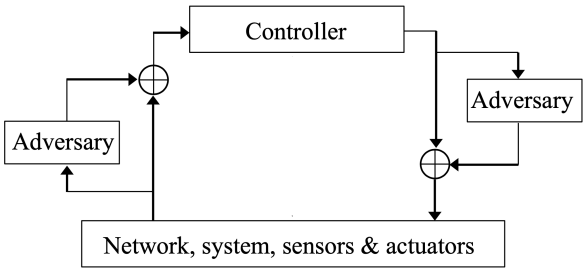


Figure 2. Adversarial model.

To capture these three key assumptions, we use the Markov Decision Process (MDP) model[37,38]. The controller is an agent evolving in a world comprising everything else, including the network, system and adversaries. At every step of its evolution, the agent makes a choice among a number of available actions, observes the outcome by sensing the state of the world and quantifies the quality of the decision with a numerical score, called reward. Several cyber-physical security and resilience issues lend themselves well to this way of seeing things.

The agent and its world are represented with the MDP model. The quantum learning part builds upon classical Reinforcement Learning (RL). The work on QML uses the feature Hilbert spaces of Schuld and Killoran[29], relying on classical kernel methods. Classical RL, such as Q-learning[39,40], assumes that the agent, i.e., the learner entity, evolves in a deterministic world. The evolution of the agent and its world is also formally modeled by the MDP. A RL algorithm trains the agent to make decisions such that a maximum reward is obtained. RL aims at optimizing the expected return of a MDP. The objectives are the same with QML. We explain MDP modeling and the quantum RL part in the sequel.

3.1.1 MDP Model

A MDP is a discrete time finite state-transition model that captures random state changes, action-triggered transitions and states-dependent rewards. A MDP is a four tuple (S,A,P_a,R_a) comprising a set of n states $S=\{0,1,\ldots,n-1\}$, a set of m actions $A=\{0,1,\ldots,m-1\}$, a transition probability function P_a and a reward function R_a. The evolution of a MDP is paced by a discrete clock. At time t, the MDP is in state s_t, such that $t=0,1,2,\ldots$. The MDP model starts in an initial state $s_0=0$. The transition probability function, denoted as

$$P_a(s,s')=Pr[s_{t+1}=s'|s_t=s,a_t=a] \tag{1}$$

defines the probability of making a transition to state s_{t+1} equal to s' at time $t+1$, when at time t the state is $s_t=s$ and action a is performed. The reward function $R_a(s,s')$ defines the immediate reward associated with the transition from state s to s' and action a. It has domain $S\times S\times A$ and co-domain $\mathbb{R}$.

Box 2 - Quantum computing basics.

With quantum computing, the basic unit of information is the quantum bit or qubit. A qubit is a binary unit of data that is simultaneously a zero and a one until the end of its life when the qubit is measured, which ends either in state zero or one. There is a probability associated with each of these two outcomes. In the ket notation, a qubit is represented by the pair

$$q=a\,|0\rangle+b\,|1\rangle$$

The symbols $|0\rangle$ and $|1\rangle$, pronounced ket zero and ket one, denote the quantum states zero and one. The parameters a and b are called probability amplitudes. Raised to the power of two, i.e., a^2 and b^2, they respectively correspond to the probability of measuring the qubit in state zero or state one. The plus sign does not represent arithmetic addition. Rather, an expression with the plus sign should be interpreted as a superposition of its operands, in this case the quantum states $|0\rangle$ and $|1\rangle$. Superposition means that a qubit is both a zero and a one at the same time. Quantum computations are done by gates. For instance the Hadamard find many applications. It can calculate the arithmetic sum of the probability amplitudes $a+b$ and their difference $a-b$.

Several qubits can be grouped together to represent a complex problem. For instance, a two-qubit quantum state q_1q_0, where q_1 is equal to $a_1\,|0\rangle+b_1\,|1\rangle$ and q_0 is equal to $a_0\,|0\rangle+b_0\,|1\rangle$, corresponds to the superposition:

$$a_1a_0\,|00\rangle+a_1b_0\,|01\rangle+b_1a_0\,|10\rangle+b_1b_0\,|11\rangle$$

Interestingly, information can be contained in binary combinations but also in probability amplitudes of ket terms. Quantum machine learning leverages both forms of information representation.

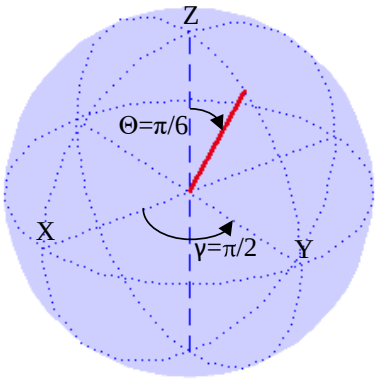
Qubits may be entangled, that is, related together such that they read in a coherent way. This means that some of the reading outcomes are made more probable than others. Besides, some reading outcomes can be made entirely non-probable.

Box 3 - Bloch sphere representation of a qubit.

A qubit can be graphically represented as a point on the surface of a unit radius Bloch sphere[a]. In this example, the complementary angle of the latitude, i.e., the colatitude θ is equal to $\pi/6$. The longitude γ is equal to $\pi/2$. A radial line is drawn from the origin to the point on the sphere surface representing the qubit. The point is defined by spherical coordinates. The two probability amplitudes are defined by two angles, namely, θ and γ. Angle θ is the colatitude. Angle γ is the longitude. In the Bloch sphere framework, a qubit is defined by the superposition

$$|\psi\rangle = \cos\frac{\theta}{2}|0\rangle + e^{j\gamma}\sin\frac{\theta}{2}|1\rangle.$$

With θ equal to zero, the qubit is $|0\rangle$. With θ equal to π, the qubit is $|1\rangle$. The Bloch sphere representation highlights the difference between a classical bit and a qubit. A classical bit can only be one of two points on the sphere, the north pole (0) or the south pole (1). A qubit can be any point on the sphere.



[a] Produced using the Hydrogenic Wavefunction Visulization Tool.

Box 4 - Q-learning RL.

Q-learning[39] RL builds upon the idea of associating rewards to actions and states. A policy, a function π with domain state set S and co-domain action set A, associates a recommended action to every state. Assuming an agent is following a policy π, every single state $s \in S$ has value $V_\pi(s)$ recursively defined as:

$$V_\pi(s) = \sum_{s' \in S} P_{\pi(s)}(s,s') \cdot \left[R_{\pi(s)}(s,s') + \gamma V_\pi(s')\right]$$

After the execution of the policy determined action $\pi(s)$, s' denotes the successor state of s. $P_{\pi(s)}(s,s')$ represents the probability of s' executing action $\pi(s)$. Under policy π, the evaluation of $V_\pi(s)$ denotes the value of state s. The reward obtained executing action a equal to $\pi(s)$ in state s is $R_a(s,s')$, or $R_{\pi(s)}(s,s')$. Constant γ in $[0,1]$ is a discounting factor, weighting the long-term reward less than the short term one. The goal of RL is to find a policy that makes the agent obtain the best possible reward. Best possible reward is achieved when the world goes through the most valued states. The optimal policy is such that for every state s

$$V_\pi(s) = \max_a \left(\sum_{s' \in S} P_a(s,s') \cdot \left[R_a(s,s') + \gamma V_\pi(s')\right] \right).$$

For obtaining the most rewarding policy, Q-learning uses the concept of Q-value. In reference to a policy π, it is a function Q with domain $S \cdot A$ and co-domain $\mathbb{R}$, defined as

$$Q_\pi(s,a) = \sum_{s' \in S} P_a(s,s') \cdot \left[R_a(s,s') + \gamma V_\pi(s')\right].$$

The optimization is accomplished through a sequence of epochs $t = 0, 1, \ldots, n$. The Q-learning algorithm is, at epoch t for the pair (s,a), where $s \in S$ is the current state and $a \in A$ is the executed action,

$$Q_t(s,a) = (1-\alpha)Q_{t-1}(s,a) + \alpha\left[R_a(s,s') + \gamma V_{t-1}(s')\right]$$

with learning factor α in $[0,1]$. For every other action pair (s,a), where $s \in S$ is not the current state or $a \in A$ is not the executed action, $Q_t(s,a)$ is equal to $Q_{t-1}(s,a)$. At epoch $t-1$, the value of state s is

$$V_{t-1}(s) = \max_a Q_{t-1}(s,a)$$

For all pairs (s,a), $Q_0(s,a)$ is set to null. It has been established[40] that Q_t converges to the optimal policy when n approaches infinity. That is, when t tends to infinity we have that Q_t tends to Q_π with the optimal policy π.

3.1.2 Quantum Reinforcement Learning

In this section we present our cyber-physical defense approach. A reader unfamiliar with quantum computing may first read Boxes 2 and 3, for a short introduction to the topic. At the heart of the approach is the concept of variational circuit. Bergholm et al.[41] interpret such a circuit as the quantum implementation of a function $f(\psi,\Theta):\mathbb{R}^m \rightarrow \mathbb{R}^n$. That is, a two argument function from a dimension m real vector space to a dimension n real vector space, where m and n are two positive integers. The first argument ψ denotes an input quantum state to the variational circuit. The second argument Θ is the variable parameter of the variational circuit. Typically, it is a matrix of real numbers. During the training, the numbers in the matrix are progressively tuned, via optimization, such that the behavior of the variational circuit eventually approaches a target function. In our cases, this function is the optimal policy π, in the terminology of Q-learning (see Box 4).

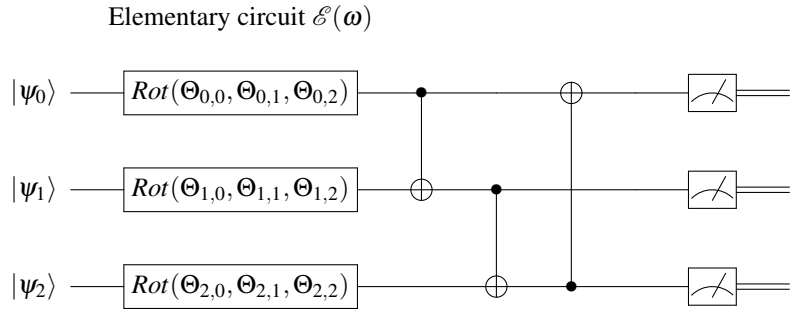


Figure 3. Three-qubit variational circuit layer $W(\Theta)$, where Θ is a three by three matrix of rotation angles.

As an example, an instance of the variational circuit design of Farhi and Neven[42] is pictured in Figure 3. In this example, both m and n are three. It is a m-qubit circuit. A typical variational circuit line comprises three stages: an initial state, a sequence of gates and a measurement device. In this case, for $i = 0,1,2$, the initial state is $|\psi_i\rangle$. The gates are a parameterized rotation and a CNOT. The measurement device is represented on the extreme right box, with a symbolic measuring dial. The circuit variable parameter Θ is a three by three matrix of rotation angles. For $i = 0,1,\ldots,m-1$, the gate $Rot(\Theta_{i,0},\Theta_{i,1},\Theta_{i,2})$ applies the x, y and z-axis rotations $\Theta_{i,0}$, $\Theta_{i,1}$ and $\Theta_{i,2}$ to qubit $|\psi_i\rangle$ on the Bloch sphere (see Box 3 for an introduction to the Bloch sphere concept). The three rotations can take qubit $|\psi_i\rangle$ from any state to any state. To create entanglement between qubits, qubit with index i is connected to qubit with index $i+1$, modulo m, using a CNOT gate. A CNOT gate can be interpreted as a controlled XOR operation. The qubit connected to the solid dot end, of the vertical line, controls the qubit connected to the circle embedding a plus sign. When the control qubit is one, the controlled qubit is XORed.

In our approach, quantum RL uses and train a variational circuit. The variational circuit maps quantum states to quantum actions, or action superpositions. The output of the variational circuit is a superposition of actions. During learning, the parameter Θ of the variational circuit is tuned such that the output of that variational collapses to actions that are proportional to their goodness, that is, the rewards they provide to the agent.

The training process can be explained in reference to Q-learning. For a brief introduction to Q-learning, see Box 4. The variational circuit is a representation of the policy π. Let $W(\Theta)$ be the variational circuit. W is called a variational circuit because it is parameterized with the matrix of rotation angles Θ. The RL process tunes the rotation angles in Θ. Given a state $s \in S$, an action $a \in A$ and epoch t, the probability of measuring value a in the quantum state A that is the output of the system

$$\mathrm{A} = W(\Theta)\,|s\rangle$$

is proportional to the ratio

$$p_{t,s,a} = \frac{Q_t(s,a)}{\sum_{i\in A} Q_t(s,i)}. \tag{2}$$

The matrix Θ is initialized with arbitrary rotations Θ_0. Starting from the initial state s_0, the following procedure is repeatedly executed. At the t^{th} epoch, random action a is chosen from set A. In current state s, the agent applies action a causing the world to make a transition. World state s' is observed. Using a, s and s', the Q-values (Q_t) are updated. For every other action pair (s,a), where $s \in S$ is not the current state or $a \in A$ is not the executed action, probability $p_{t,s,a}$ is also updated according to Equation (2). Using Θ_{t-1} and the $p_{t,s,a}$ probabilities, the variational circuit parameter is updated and yields Θ_t.

The variational circuit is trained such that under input state $|s\rangle$, the measured output in the system $\mathrm{A} = W(\Theta)|s\rangle$ is a with probability $p_{t,s,a}$. Training of the circuit can be done with a gradient descent optimizer[41]. Step-by-step, the optimizer minimizes the distance between the probability of measuring $|a\rangle$ and the ratio $p_{t,s,a}$, for a in A.

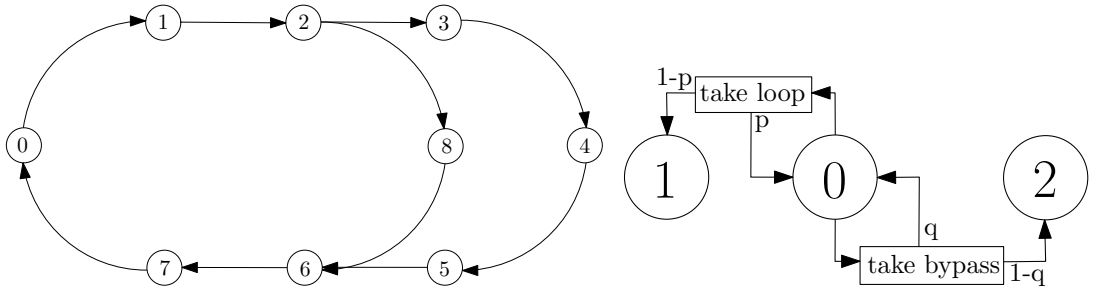
The variational circuit $W(\Theta)$ is trained on the probabilities of the computational basis members of A, in a state s. Quantum RL repeatedly updates Θ such that the evaluation of $\mathrm{A} = W(\Theta)|s\rangle$ yields actions with probabilities proportional to the rewards. That is, the action action recommended by the policy is $\arg\max_{a \in A} \mathrm{A}$, i.e., the row-index of the element with highest probability amplitude.

Since $W(\Theta)$ is a circuit, once trained it can be used multiple times. Furthermore, with this scheme the learned knowledge Θ, which are rotations, can be easily stored or shared with other parties. This RL scheme can be implemented using the resources of the PennyLane software[41]. An illustrative example is discussed in the next subsection.

3.2 Illustrative Example

In this section, we illustrate our approach with an example. We model the agent and its world with the MDP model. We define the attack model. We explain the quantum representation of the problem. We demonstrate enhancement of resilience leveraging quantum RL.

3.2.1 Agent and Its World



(a) Configuration (b) MDP

Figure 4. (a) Discrete two-train configuration. (b) MDP representation of the agent in its world. Circles represent states. Arrows represent action-triggered transitions.

Let us consider the discrete two-train configuration of Figure 4(a). Tracks are broken into sections. We assume a scenario where Train 1 is the agent and Train 2 is part of its world. There is an outer loop visiting points 3, 4 and 5, together with a bypass from point 2, visiting point 8 to point 6. Traversal time is uniform across sections. The normal trajectory of Train 1 is the outer loop, while maintaining a Train 2-Train 1 distance greater than one empty section. For example, if Train 1 is at point 0 while Train 2 is at point 7, then the separation distance constraint is violated. The goal of the adversary is to steer the system in a state where the separation distance constraint is violated. When a train crosses point 0, it has to make a choice: either traverse the outer loop or take the bypass. Both trains can follow any path and make independent choices, when they are at point 0.

In the terms of RL, Train 1 has two actions available: take loop and take bypass. The agent gets k reward points for a relative Train 2-Train 1 distance increase of k sections with Train 2. It gets $-k$ reward points, i.e., a penalty, for a relative Train 2-Train 1 distance decrease of k sections with Train 2. For example, let us assume that Train 1 is at point 0 and that Train 2 is at point 7. If both trains, progressing a the same speed, take the loop or both decide to take the bypass, then there is no relative distance change. The agent gets no reward. When Train 1 decides to take the bypass and Train 2 decides to take the loop, the agent gets two reward points, at return to point zero (Train 2 is at point five). When Train 1 decides to take the loop and Train 2 decides to take the bypass, the agent gets four reward points, at return to point zero (Train 2 is at point one, Train 2-Train 1 distance is five sections).

The corresponding MDP model is shown in Figure 4(b). The state set is $S = \{0,1,2\}$. The action set is $A = \{a_0 = \text{take loop}, a_1 = \text{take bypass}\}$. The transition probability function is defined as $P_{a_0}(0,0) = p$, $P_{a_0}(0,1) = 1-p$, $P_{a_1}(0,0) = q$ and $P_{a_1}(0,2) = 1-q$. The reward functions is defined as $R_{a_0}(0,0) = 0$, $R_{a_0}(0,1) = 4$, $R_{a_1}(0,0) = 0$ and $R_{a_1}(0,2) = 2$. This is interpreted as follows. In the initial state 0 with a one-section separation distance, the agent selects an action to perform: take loop or take bypass. Train 1 performs the selected action. When selecting take loop, with probability p the environment goes back to state 0 (no reward) or with probability $1-p$ it moves to state 1, with a five-section separation distance (reward is four). When selecting take bypass, with probability q the environment goes back to state 0 (no reward) or with probability $1-q$ it moves state 2, with a three-section separation distance (reward is two). The agent memorizes how good it has been to perform a selected action.

As shown in this example, multiple choices might be available in a given state. A MDP is augmented with a policy. At any given time, the policy tells the agent which action to pick such that the expected return is maximized. The objective of RL is finding a policy maximizing the return. Q-learning captures the optimal policy into a state-action value function $Q(s,a)$, i.e., an estimate of the expected discounted reward for executing action a in state s[39,40]. Q-learning is an iterative process. $Q_t(s,a)$ is the state-action at the t^{th} episode of learning.

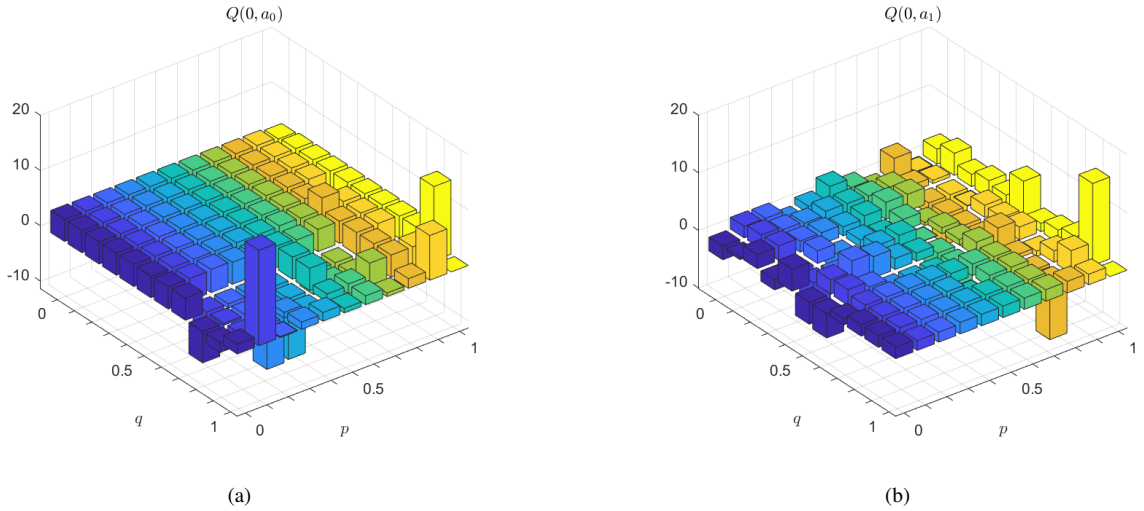


Figure 5. Q-values for actions a_0 and a_1, for values of probabilities p and q ranging from zero to one, in steps of 0.1.

Figure 5 plots side by side the Q-values for actions a_0 and a_1, for values of probabilities p and q ranging from zero to one, in steps of 0.1. As a function of p and q, on which the agent has no control, the learned policy is that in state zero should pick the action among a_0 and a_1 that fields the maximum Q-value, which can be determined from Figure 5. This figure highlights the usefulness of RL, even for such a simple example the exact action choice is by far not always obvious. However, RL tells what this choice should be.

The example is simple enough so that a certain number of cases can be highlighted. When probabilities p and q tend to one, it means that the adversary is more likely to behave as the agent. Inversely, when p and q tend to null, the adversary is likely to make a different choice from that of the agent. Such a bias can be explained by the existence of an insider that leaks information to the adversary when the agent makes its choice at point 0. In the former case, the agent is trapped in a risky condition. In the latter case, the adversary is applying its worst possible strategy. When p and q are both close to 50%, the adversary is behaving arbitrarily. On the long term, the most rewarding action for the agent is to take the loop. It is of course possible to update the policy according to a varying adversarial behavior, i.e., changing values for p and q. In following, we address this RL problem with a quantum approach.

3.2.2 Quantum Representation

The problem in the illustrative example of Figure 4 comprises only one state (0) where choices are available. A binary decision is taken in that state. The problem can be solved by a single qubit variational quantum circuit. The output of the circuit is a single qubit with the simple following interpretation. $|0\rangle$ is action take loop, while $|1\rangle$ is action take bypass.

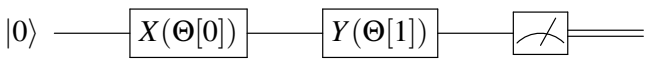
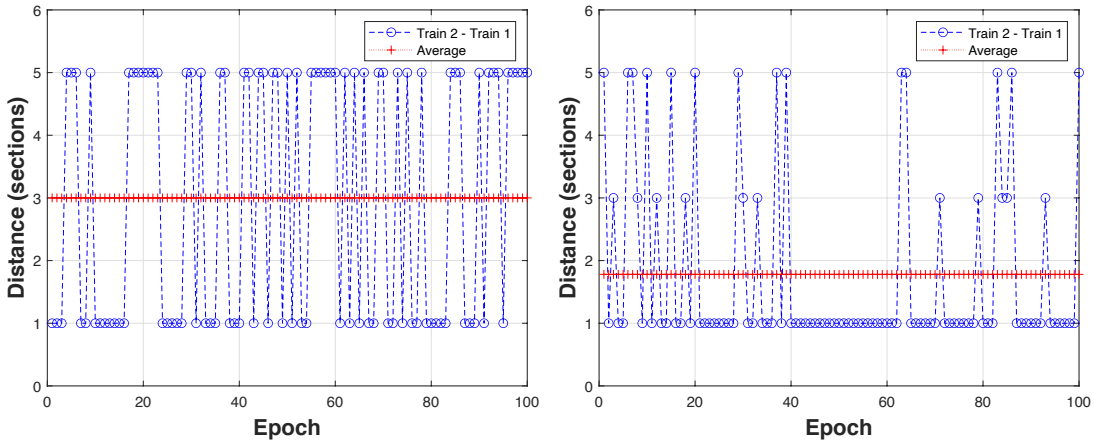


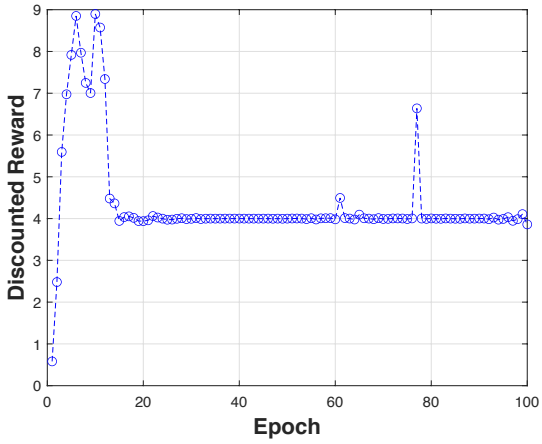
Figure 6. Single-qubit variational circuit $W(\Theta)$.

For this example, we use the variational quantum circuit pictured in Figure 6. The input of the circuit is ground state $|0\rangle$. Two rotation gates and a measurement gate are used. The circuit consists of two quantum gates: an X gate and an Y gate, parameterized with rotations $\Theta[0]$ and $\Theta[1]$ about the x-axis and y-axis, on the Bloch sphere. There is a measurement gate at the very end, converting the output qubit into a classical binary value. This value is an action index. The variational circuit is tuned by training such that it outputs the probably most rewarding choice.

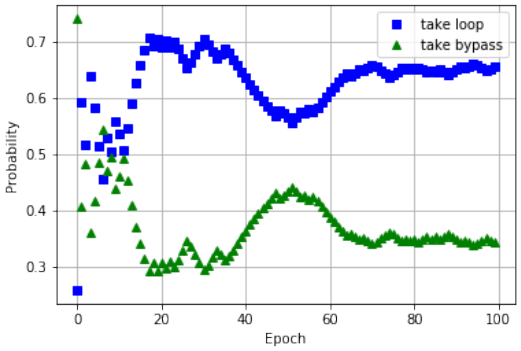


(a) Adversary is behaving arbitrarily.

(b) Adversary mimics agent.



(c) Value of state zero.



(d) Evolution of probabilities.

Figure 7. (a) The adversary randomly alternates between take loop and take bypass, with equal probabilities. (b) The agent choices are leaking, e.g., due to the presence of an insider. With high probability, the adversary is mimicking the agent. (c) Evolution of the value of state zero. (d) Evolution of quantum variational circuit probabilities, with learning rate α is 0.01.

A detailed implementation of the example is available as supplementary material in a companion github repository[19]. Figure 7 provides graphical interpretations of the two-train example. In all the plots, the x-axis represents epoch (time). Part (a) shows the Train 2-Train 1 separation distance (in sections) as a function of the epoch, when the agent is doing the normal behavior, i.e., do action take loop, and the adversary is behaving arbitrarily, p and q are equal to 0.5. The average distance (three sections) indicates that more often the separation distance constraint is not violated. Part (b) also shows the Train 2-Train 1 distance as a function of the epoch, but this time the adversary figured out the behavior of the agent. The average distance (less than two sections) indicates that the separation distance constraint is often violated. Part (c) plots the value of state zero, in Figure 4(b), versus epoch. The adversary very likely learns the choices made by the agent, when at point 0. There is an insider leaking the information. Train 2 is likely to mimic Train 1. The probabilities of p and q are equal to 0.9. In such a case, for Train 1 the most rewarding choice is to take the loop. Part (d) shows the evolution of the probabilities of the actions, as the training of the quantum variational circuit pictured in Figure 6 progresses. They evolve consistently with the value of state zero (learning rate α is 0.01). The y-axis represents probabilities of selecting the actions take loop (square marker) and take bypass (triangle marker). Under this condition, quantum RL infers that the maximum reward is obtained selecting the *take loop* action. It has indeed higher probability than the *take bypass* action.

4 Discussion

Section 3.2 detailed an illustrative example. Of course, it can be enriched. The successors of states 1 and 2 can be expanded. More elaborate railways can be represented. More sophisticated attack models can be studied. For example, let v_i denote the

velocity of Train i, where i is equal to 1 or 2. Hijacking control signals, the adversary may slowly change the velocity of one of the trains until the separation distance is not greater than a threshold τ. Mathematically, the velocity of the victimized train is represented as

$$v_i(t+\Delta) = v_i(t) + \alpha e^{\beta(t+\Delta)}.$$

The launched time of the attack is t. $v_i(t)$ is the train velocity at time t, while $v_i(t+\Delta)$ is the speed after a delay Δ. Symbols α and β are constants. During normal operation, the two trains are moving at equal constant velocities. During an attack on the velocity of a train, the separation distance slowly shrinks down to a value equal to or lower than a threshold. The safe-distance constraint is violated. While an attack is being perpetrated, the state of the system must be recoverable[43], i.e., the velocities and compromised actuators or sensors can be determined using redundant sensing resources.

The approach can easily be generalized to other applications. For instance, let us consider switching control strategies used to mitigate DoS attacks[6] or input and output signal manipulation attacks[44]. States are controller configurations, actions are configuration-to-configuration transitions and rewards are degrees of attack mitigation. The variational circuit is trained such that the agent is steered in an attack mitigation condition. This steering strategy is acquired through RL.

Table 1. Conceptual comparison of classical versus quantum RL.

Reinforcement Learning		
Concept	Q-learning	Quantum
Data structure	Q-value table	Variational circuit
Resources	n times m numbers	$k \log n$ gates

In Section 3.1.2, quantum RL is explained referring to Q-learning. Table 1 compares Q-learning and quantum RL. The first column list the RL concepts. The second column define their implementation in Q-learning[45]. The third column lists their analogous in QML. The core concept is a data structure used to represent the expected future rewards for action at each state. Q-learning uses a table while QML employs a variational circuit. The following line quantifies the amount of resources needed in every case. For Q-learning, n times m expected reward numbers need to be stored, where n is the number of states and m the number of actions of the MDP. For QML, $k \log n$ quantum gates are required, where k is the number of gates used for each variational circuit line. Note that deep learning[45–47] and QML can be used to approximate the Q-value function, with respectively, a neural network or a variational quantum circuit. The second line compares tuneable parameters, which are neural network weight for the classical model and variational circuit rotations for the quantum model. For both models, gradient descent optimization method is used to tune iteratively the model, the neural network or variational circuit. Chen et al.[34] did a comparison of Deep learning and quantum RL. According to their analysis, similar results can be obtained with similar order quantities of resources. While there is no *neural network computer* in the works, apart for hardware accelerators, there are considerable efforts being deployed to develop the quantum computer[48]. The eventually available quantum computer will provide an incomparable advantage to the ones who will have access to the technology, in particular the defender or adversary.

There are a few options for quantum encoding of states, including computational basis encoding, single-qubit unitary encoding and probability encoding. They all have a time complexity cost proportional to the number of states. Computational basis encoding is the simplest to grasp. States are indexed $i = 0, ..., m-1$. In the quantum format, the state is represented as $|i\rangle$.

Amplitude encoding works particularly well for supervised machine learning[31,49]. For example, let $\vec{\psi} = (\psi_0, \ldots, \psi_7)$ be such a unit vector. Amplitude encoding means that the data is encoded in the probability amplitudes of quantum states. Vector $\vec{\psi}$ is mapped to the following three-qubit register

$$|\psi\rangle = \sum_{i=0}^{7} \psi_i |i\rangle .$$

The term $|i\rangle$ is one of the eight computational basis members for a three-qubit register. Every feature-vector component ψ_i becomes the probability amplitude of computational basis member $|i\rangle$. The value ψ^2 corresponds to the probability of measuring the quantum register in state $|i\rangle$. The summation operation is interpreted as the superposition of the quantum states $|i\rangle$, $i = 0, \ldots, 7$. Superposition means that the quantum state $|\psi\rangle$ assumes all the values of i at the same time. In this representation exercise, there is a cost associated with coding the feature vectors in the quantum format, linear in their number. The time complexity of an equivalent classical computing classifier is linear as well. However, in the quantum format the time taken to do classification is data-size independent. The coding overhead, although, makes quantum ML performance comparable classical ML performance. Ideally, data should be directly represented in the quantum format, bypassing the classical to quantum data translation step and enabling gains in performance. Further research in quantum sensing is needed to enable this[50].

There are also other RL training alternatives. Dong et al. have developed a quantum RL approach[51]. In the quantum format, a state $i \in S$ of the MDP is mapped to quantum state $|i\rangle$. Similarly, an action $j \in A$ is mapped to quantum state $|j\rangle$. In state i,

the action space is represented by the quantum state

$$|A_i\rangle = \sum_{j=0}^{m-1} \psi_i |a_j\rangle$$

where the probability amplitudes ψ_i's, initially all equal, are modulated, using Grover iteration by the RL procedure. In state i, selecting an action amounts to observing the quantum state $|A_i\rangle$. According to the non-cloning theorem, it can be done just once, which is somewhat limited.

By far, not all QML issues have been resolved. More research on encoding and training is required. Variational circuit optimization experts[41] highlight the need for more research to determine what works best, among the available variational circuit designs, versus the type of problem considered.

5 Conclusion

We have presented our vision of a next generation cyber-physical defense in the quantum era. In the same way that nobody thinks about system protection making abstraction of the quantum threat, we claim that in the future nobody will think about cyber-physical defense without using quantum resources. When available, adversaries will use quantum resources to support their strategies. Defenders must be equipped as well with the same resources to face quantum adversaries and achieve security beyond breach. ML and quantum computing communities will play very important roles in the design of such resources. This way, the quantum advantage will be granted to defenders rather than solely adversaries. The essence of the war between defenders and adversaries is knowledge. RL can be used by an adversary for the purpose of system identification, an enabler for covert attacks. The paper has clearly demonstrated the plausibility of using quantum technique to search defense strategies and counter adversaries. Furthermore, the design of new defense techniques can leverage quantum ML to speedup decision making and support adaptive control. These benefits of QML will although materialize when the quantum computer will be available. These ideas have been explored in this article, highlighting capabilities and limitations which resolution requires further research.

Acknowledgments — We acknowledge the financial support from the Natural Sciences and Engineering Research Council of Canada (NSERC) and the European Commission (H2020 SPARTA project, under grant agreement 830892).

References

1. Ding, D., Han, Q.-L., Ge, X. & Wang, J. Secure state estimation and control of cyber-physical systems: A survey. *IEEE Transactions on Syst. Man, Cybern. Syst.* **51**, 176–190 (2020).
2. Ge, X., Han, Q.-L., Zhang, X.-M., Ding, D. & Yang, F. Resilient and secure remote monitoring for a class of cyber-physical systems against attacks. *Inf. sciences* **512**, 1592–1605 (2020).
3. Ding, D., Han, Q.-L., Xiang, Y., Ge, X. & Zhang, X.-M. A survey on security control and attack detection for industrial cyber-physical systems. *Neurocomputing* **275**, 1674–1683 (2018).
4. Courtney, S. & Riley, M. Biden rushes to protect power grid as hacking threats grow (2021). Bloomberg, Available on-line at https://j.mp/3fyZcQE, Last Access: June 2021.
5. Teixeira, A., Shames, I., Sandberg, H. & Johansson, K. H. A secure control framework for resource-limited adversaries. *Automatica* **51**, 135–148 (2015).
6. Zhu, Y. & Zheng, W. X. Observer-based control for cyber-physical systems with periodic dos attacks via a cyclic switching strategy. *IEEE Transactions on Autom. Control.* **65**, 3714–3721 (2020).
7. Shor, P. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM J. on Comput.* **26**, 1484–1509 (1997).
8. Shor, P. W. & Preskill, J. Simple proof of security of the bb84 quantum key distribution protocol. *Phys. Rev. Lett.* 441–444 (2000).
9. McEliece, R. J. A public-key cryptosystem based on algebraic. *Coding Thv* **4244**, 114–116 (1978).
10. Merkle, R. *Secrecy, Authentication, and Public Key Systems*. Computer Science Series (UMI Research Press, 1982).
11. Patarin, J. Hidden fields equations (hfe) and isomorphisms of polynomials (ip): Two new families of asymmetric algorithms. In *International Conference on the Theory and Applications of Cryptographic Techniques*, 33–48 (1996).
12. Hoffstein, J., Pipher, J. & Silverman, J. H. Ntru: A ring-based public key cryptosystem. In *International Algorithmic Number Theory Symposium*, 267–288 (Springer, 1998).

13. Regev, O. On lattices, learning with errors, random linear codes, and cryptography. *J. ACM (JACM)* **56**, 34 (2009).

14. Jao, D. & De Feo, L. Towards quantum-resistant cryptosystems from supersingular elliptic curve isogenies. *PQCrypto* **7071**, 19–34 (2011).

15. Nielsen, M. A. & Chuang, I. Quantum computation and quantum information (2002).

16. Satoh, T. *et al.* Attacking the quantum internet. *IEEE Transactions on Quantum Eng.* **2**, 1–17 (2021).

17. Iwakoshi, T. Security evaluation of y00 protocol based on time-translational symmetry under quantum collective known-plaintext attacks. *IEEE Access* **9**, 31608–31617 (2021).

18. Giraldo, J., Sarkar, E., Cardenas, A. A., Maniatakos, M. & Kantarcioglu, M. Security and privacy in cyber-physical systems: A survey of surveys. *IEEE Des. & Test* **34**, 7–17 (2017).

19. Barbeau, M. & Garcia-Alfaro, J. Supplementary material to: Cyber-physical defense in the quantum era. https://github.com/jgalfaro/DL-PoC (2021).

20. Schneier, B. Modelling security threats. *Dr. Dobb's J.* (1999).

21. Lallie, H. S., Debattista, K. & Bal, J. A review of attack graph and attack tree visual syntax in cyber security. *Comput. Sci. Rev.* **35**, 100219 (2020).

22. Heule, M. J. & Kullmann, O. The science of brute force. *Commun. ACM* **60**, 70–79 (2017).

23. Arnold, F., Hermanns, H., Pulungan, R. & Stoelinga, M. Time-dependent analysis of attacks. In *International Conference on Principles of Security and Trust*, 285–305 (Springer, 2014).

24. Hoffman, D. & Karst, O. J. The theory of the rayleigh distribution and some of its applications. *J. Ship Res.* **19**, 172–191 (1975).

25. Gudbjartsson, H. & Patz, S. The rician distribution of noisy mri data. *Magn. resonance medicine* **34**, 910–914 (1995).

26. Arnold, F., Pieters, W. & Stoelinga, M. Quantitative penetration testing with item response theory. In *2013 9th International Conference on Information Assurance and Security (IAS)*, 49–54 (IEEE, 2013).

27. Chio, C. & Freeman, D. *Machine Learning and Security: Protecting Systems with Data and Algorithms* (O'Reilly Media, 2018).

28. Biamonte, J. *et al.* Quantum machine learning. *Nature* **549**, 195–202 (2017).

29. Schuld, M. & Killoran, N. Quantum machine learning in feature Hilbert spaces. *Phys. Rev. Lett.* **122**, 040504 (2019).

30. Havlíček, V. *et al.* Supervised learning with quantum-enhanced feature spaces. *Nature* **567**, 209–212 (2019).

31. Schuld, M. & Petruccione, F. *Supervised Learning with Quantum Computers*. Quantum science and technology (Springer, 2018).

32. Montangero, S. *Introduction to Tensor Network Methods: Numerical simulations of low-dimensional many-body quantum systems* (Springer International Publishing, 2018).

33. Huggins, W., Patil, P., Mitchell, B., Whaley, K. B. & Stoudenmire, E. M. Towards quantum machine learning with tensor networks. *Quantum Sci. technology* **4**, 024001 (2019).

34. Yen-Chi Chen, S. *et al.* Variational quantum circuits for deep reinforcement learning. *IEEE Access* 141007–141024 (2020).

35. Lockwood, O. & Si, M. Reinforcement learning with quantum variational circuit. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, vol. 16, 245–251 (2020).

36. Barbeau, M., Cuppens, F., Cuppens, N., Dagnas, R. & Garcia-Alfaro, J. Resilience estimation of cyber-physical systems via quantitative metrics. *IEEE Access* **9**, 46462–46475 (2021).

37. Bellman, R. A Markovian decision process. *J. mathematics mechanics* **6**, 679–684 (1957).

38. Puterman, M. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Wiley Series in Probability and Statistics (Wiley, 2014).

39. Watkins, C. J. C. H. Learning from delayed rewards. *PhD thesis, King's Coll. Univ. Camb.* (1989).

40. Watkins, C. J. C. H. & Dayan, P. Q-learning. *Mach. Learn.* **8**, 279–292 (1992).

41. Bergholm, V. *et al.* Pennylane: Automatic differentiation of hybrid quantum-classical computations. arXiv preprint arXiv:1811.04968 (2020).

42. Farhi, E. & Neven, H. Classification with quantum neural networks on near term processors (2018). ArXiv:1802.06002.

43. Weerakkody, S., Ozel, O., Mo, Y., Sinopoli, B. *et al.* Resilient control in cyber-physical systems: Countering uncertainty, constraints, and adversarial behavior. *Foundations Trends Syst. Control.* **7**, 1–252 (2019).

44. Segovia-Ferreira, M., Rubio-Hernan, J., Cavalli, R. & Garcia-Alfaro, J. Switched-based resilient control of cyber-physical systems. *IEEE Access* **8**, 212194–212208 (2020).

45. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. *nature* **521**, 436–444 (2015).

46. Mnih, V. *et al.* Human-level control through deep reinforcement learning. *nature* **518**, 529–533 (2015).

47. Mnih, V. *et al.* Asynchronous methods for deep reinforcement learning. In *International conference on machine learning*, 1928–1937 (PMLR, 2016).

48. Barbeau, M. *et al.* The quantum what? advantage, utopia or threat? *Digit. Welt* **4**, 34–39 (2021).

49. Barbeau, M. Recognizing drone swarm activities: Classical versus quantum machine learning. *Digit. Welt* **3**, 45–50 (2019).

50. Degen, C. L., Reinhard, F. & Cappellaro, P. Quantum sensing. *Rev. Mod. Phys.* **89**, 035002 (2017).

51. Dong, D., Chen, C., Li, H. & Tarn, T. Quantum reinforcement learning. *IEEE Transactions on Syst. Man, Cybern. Part B (Cybernetics)* **38**, 1207–1220 (2008).