Deep reinforcement learning using deep-Q-network for Global Maximum Power Point tracking: Design and experiments in real photovoltaic systems

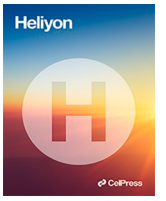
Luis Felipe Giraldo, Jorge Felipe Gaviria, María Isabella Torres, Corinne Alonso, Michael Bressan

Heliyon 10 (2024) e37974

Contents lists available at ScienceDirect

Heliyon

journal homepage: www.cell.com/heliyon



Research article

Deep reinforcement learning using deep-Q-network for Global Maximum Power Point tracking: Design and experiments in real photovoltaic systems

Luis Felipe Giraldo [c], Jorge Felipe Gaviria [a], María Isabella Torres [a,c], Corinne Alonso [b], Michael Bressan [a,d,e,*]

[a] Department of Electrical and Electronic Engineering, Universidad de Los Andes, Bogotá, Colombia
[b] Laboratory for Analysis and Architecture of Systems, LAAS-CNRS, Toulouse, France
[c] Department of Biomedical Engineering, Universidad de Los Andes, Bogotá, Colombia
[d] PROMES-CNRS, Rambla de la thermodynamique, Tecnosud, 66100 Perpignan, France
[e] University of Perpignan Via Domitia, 52 Avenue Paul Alduy, 66860 Perpignan, France

ARTICLE INFO

Keywords:
Photovoltaic systems
Global maximum power point tracking
Neural networks
Reinforcement learning

ABSTRACT

This paper presents a methodology for integrating Deep Reinforcement Learning (DRL) using a Deep-Q-Network (DQN) agent into real-time experiments to achieve the Global Maximum Power Point (GMPP) of Photovoltaic (PV) systems under various environmental conditions. Conventional methods, such as the Perturb and Observe (P&O) algorithm, often become stuck at the Local Maximum Power Point (LMPP) and fail to reach the GMPP under Partial Shading Conditions (PSC). The main contribution of this work is the experimental validation of the DQN agent's implementation in a synchronous DC-DC Buck converter (step-down converter) un-der both uniform and PSC conditions. Additionally, we establish a testing pipeline for DRL models. The DQN agent's performance is evaluated alongside the P&O algorithm. Results consistently indicate the DQN agent's superiority over the P&O algorithm in all simulated scenarios. Although this trend does not entirely replicate in real-world test setups, significant results are observed. Specifically, in PSC sce-narios where the P&O algorithm becomes trapped at an LMPP, the DQN algorithm extracts up to 63.5 % more power than the P&O algorithm. An open repository is available, containing PCB schematic designs and layouts, along with the code used for model training and deployment.

1. Introduction

With the rise in global energy demand, solar energy has emerged as one of the most widely used renewable energy sources. As reported by Ref. [1], solar energy contributed to over 2 % of global electricity in 2019 due to its decreasing cost and significant potential in regions with abundant solar radiation. Furthermore, it is estimated that global solar PV capacity will increase from 593.9 GW in 2019 to 1582.9 GW in 2030, driven by capacity additions in China, India, Germany, the US, and Japan [2]. However, PV generation systems still have low efficiency in electric power generation [3]. The output power of these systems is highly influenced by

* Corresponding author. Department of Electrical and Electronic Engineering, Universidad de Los Andes, Bogotá, Colombia
E-mail addresses: lf.giraldo404@uniandes.edu.co (L.F. Giraldo), jf.gaviria@uniandes.edu.co (J.F. Gaviria), mi.torres@uniandes.edu.co (M.I. Torres), alonsoc@laas.fr (C. Alonso), m.bressan@uniandes.edu.co (M. Bressan).

https://doi.org/10.1016/j.heliyon.2024.e37974
Received 12 October 2023; Received in revised form 31 May 2024; Accepted 14 September 2024
Available online 16 September 2024

Nomenclature

ADC	Analog to Digital Converter
ANN	Artificial Neural Networks
DAC	Digital to Analog Converter
DDPG	Deep Deterministic Policy Gra-dient
DL	Deep Learning
DQL	Deep Q-Learning
DQN	Deep Q-Networks
GMPP	Global Maximum Power Point
GMPPT	Global Maximum Power Point Tracking
IC	Incremental-Conductance
LMPP	Local Maximum Power Point
MDP	Markov Desicion Process
ML	Machine Learning
MPPT	Maximum Power Point Tracking
ONNX	Open Neural Network Exchange
P&O	Perturb and Observe
PMSG	Permanent Magnet Synchronous Generator
PS	Partial Shading
PSC	Partial Shading Conditions
PV	Photovoltaic
PWM	Pulse Width Modulated
RL	Reinforcement Learning
SAC	Soft Actor-Critic
TD3	Twin Delayed Deep Deter-minis- tic Policy Gradient

weather fluctuations and the nature of the connected load due to the non-linear I-V and P-V characteristics of PV modules [4]. To address these inherent issues in PV systems, Machine Learning (ML) methods have been applied to power prediction [5], irradiance prediction [6,7], site adaptation [8], fault detection [9], and control.

A PV array is composed of multiple PV modules connected in series. Each PV module is likely exposed to varying irradiance levels due to factors like soiling, mov-ing clouds, and trees. The output characteristics of these PV modules display nu-merous Local Maximum Power Points (LMPPs). This circumstance leads to energy losses and reduced efficiency. The most widely adopted MPPT control methods are the Perturb and Observe (P&O) and Incremental-Conductance (IC) methods due to their simplicity [10,11]. However, to prevent oscillations around the Maxi-mum Power Point (MPP) during the search process, these methods require specific calibration levels. Under Partial Shading Conditions (PSC), these algorithms tend to remain at an LMPP due to the overheating of shaded PV cells, as observed by Bressan et al. [12], resulting in decreased energy conversion efficiency. Additionally, these algorithms cannot locate the GMPP under PSC conditions, as pointed out by Liu et al. [13]. The motivation behind most research works has been to determine the GMPP at an extremely fast rate, aiming to alleviate specific concerns such as reduced tracking speed, diminished efficiency, and unwarranted sweeping of the PV output power curve [14–16]. Robust GMPPT algorithms need to be developed to enhance the performance of PV systems.

While conventional MPPT methods have received significant attention, there are several challenges that must be addressed to enhance PV system efficiency and relia-bility: (i) High efficiency in extracting GMPP, particularly under complex irradiance conditions, (ii) Practical implementation complexity of existing GMPP methods,(iii) The need for rapid GMPP determination. DRL methods have emerged as an appealing and advantageous choice for addressing the complexities posed by partial shading in PV systems.

1.1. Litterature review

In recent years, many methods have been developed to extract the MPPT of PV systems, including Spider Monkey Optimization-based MPPT [17], Swarm Intelligence-based MPPT [18], Salp Swarm Optimization algorithm [19], and the Archimedes Optimization Algorithm (AOA) [20], as well as approaches using a digital twin [21].

These studies indicate the need for improving convergence speeds and addressing the low mitigation rate of settling time and accuracy. An increasing number of studies have focused on the performance of Reinforcement Learning (RL) agents applied to MPPT. RL is a technique that enables an agent to learn control policies through rewards obtained from interacting with the environment, defined as a Markov Deci-sion Process [22]. One main advantage of these models is their independence from complex mathematical control system models.

One of the initial applications of RL in MPPT was carried out by Kofinas et al. [23]. The authors proposed a tabular Q-Learning approach to track the MPP, achieving high convergence stability within a shorter computational time compared to other meta-heuristic methods. This resulted in higher power extraction using the reinforcement learning method. The obtained results

showed that the designed algorithm outperformed the conventional P&O method in three different scenarios involving changes in irradiance and temperature. This research has led to the ex-ploration of other similar tabular Q-Learning methods applied to MPPT, such as those developed by Aurobinda et al. [24] and Bavarinos et al. [25]. These studies involve comparisons between Q-Learning agents, SARSA agents, and fuzzy-logic- sliding mode control.

While previous authors achieved remarkable results, the algorithms they em-ployed had a significant drawback: the limited state and action spaces used to at-tain their objectives. Furthermore, the studies referenced did not address scenarios involving PSC.

In Singh et al. [26], DRL was used to track MPPT, with a fuzzy reward mechanism introduced to enhance the translation of continuous space into various levels of abstraction. The authors simulated a PV array model connected to a variable load. In Arianborna et al. [27], a DQN approach for controlling the MPPT of a Permanent Magnet Synchronous Generator (PMSG) connected to a wind turbine was proposed. This approach preserved the advantages of the Q-Learning method while addressing its pre-existing disadvantages.

In the study conducted by Ref. [21], a Deep Deterministic Policy Gradient (DDPG) was implemented to attain a stable output at MPP in a DC-DC converter. The results demonstrated notable enhancements over conventional methods, particularly when compared to the widely used P&O method. However, to fully ascertain the effectiveness of the DDPG agent across diverse operating conditions, including sce-narios like PSC, further investigations and testing are imperative.

In Phan et al. [28], DQN and DDPG agents were proposed for conducting MPPT control. These approaches enabled the agents to address continuous state and action spaces in the presence of PSC. The outcomes of these models were compared to the commonly used P&O method. The results demonstrated that the DQN and DDPG models effectively reached the GMPP under PSC, whereas the P&O algorithm got stuck at an LMPP, resulting in lower power production.

Additional actor-critic-based RL agents were also tested for MPPT. Avila et al. [29,30] developed a model-free DL algorithm based on RL, trained and evaluated in a custom OpenAI Gym environment implemented in Python [31]. The TD3 algo-rithm was implemented to solve this issue in Ref. [30]. The experiment results revealed that the algorithm achieved a maximum operating power point with a difference of less than 1 % from the theoretical MPP. Furthermore, Naseem et al. [32] provided a detailed assessment of different GMPPT methods, emphasizing that significant enhancements can be made to GMPPT methods in terms of efficiency and fast op-eration, warranting further investigation for improved results. For instance, process and hardware implementation, as well as circuit intricacy, play significant roles in selecting the appropriate GMPPT method.

Comprehensive reviews on the application of machine learning techniques to MPPT were conducted by Gaviria et al. [33] and Glavic [34].

The main limitation in all previous studies is the lack of experiments in real scenarios to validate the results presented in simulations. Moreover, an experimental framework is necessary to guide future research applying RL, especially DRL, to GMPPT.

1.2. Contributions

With this motivation, the authors aim to conduct simulations and practical im-plementations utilizing a DC-DC buck converter. The primary objective is to employ a DQN agent and analyze its performance in comparison to the P&O method for GMPPT. This comparative analysis will encompass scenarios under both uniform and PSC, and will span across simulated as well as real environmental conditions. The proposed system is capable of rapidly extracting GMPPT under PSC, using 125,000 samples for model training. Moreover, providing an experimental guide us-ing deep RL agents on GMPPT is essential, facilitated by the open-source resources and pipeline provided. This paper continues the work from Ref. [33] and is highly inspired by the research in Ref. [28]. The main contributions of this research are as follows.

- A comparison between a DQN agent and the P&O method for MPPT under uniform and PSC conditions in both simulated and real scenarios.
- The establishment of a pipeline for real-time testing of DRL models trained using MATLAB/Simulink, a Raspberry Pi, and TensorFlow Lite.
- An open repository that includes the schematic of the used converter, the training environment for the RL model, the code for converting the MATLAB model to the TensorFlow Lite framework, and the code for deploying and testing the algorithms on a Raspberry Pi 4b. The open-source repository can be accessed through the following GitHub repositories[1]: and.[2]

Although this paper focuses on DQN experimentation, the established pipeline would be useful for testing and conducting experiments with any of the following DRL algorithms for GMPPT.

- Deep Q-Network (DQN): A reinforcement learning algorithm that combines traditional Q-learning with deep neural networks to handle complex, high-dimensional environments, such as those encountered in the GMPPT problem. It uses experience replay and fixed Q-targets to stabilize and improve the learning process.

[1] Schematic and Layout of the Buck Converter, along with the training procedure of the DQN model and conversion to TFLite file: https://github.com/SmartSystems-UniAndes/Train_and_Convert_RL_MPPT.

[2] Deployment of the DQN model to Raspberry Pi 4b: https://github.com/SmartSystems-UniAndes/Reinforcement_Learning_MPPT_RaspberryPi_Deploy.

- Deep Deterministic Policy Gradient (DDPG): An algorithm in reinforcement learning that combines policy gradient methods with Q-learning, optimized for continuous action spaces. It uses an actor-critic approach, where the actor learns the policy and the critic evaluates it, and employs experience replay and target networks for stability.
- Twin Delayed Deep Deterministic Policy Gradient (TD3): An extension of DDPG, introducing three key improvements: twin Q-networks to reduce over-estimation of Q-values, delayed policy updates, and target policy smoothing, significantly enhancing performance and stability in continuous action spaces.
- Soft Actor-Critic (SAC): Focuses on learning policies that maximize not just reward but also entropy, promoting exploration. SAC is actor-critic-based, uses twin Q-networks similar to TD3, and is known for its sample efficiency and robustness across a wide range of environments.

This paper is organized as follows. Firstly, a brief introduction to the main ML techniques applied in PV systems is presented in Section 2. Within this section, the converter and PV system used for conducting the experiments are described, along with the definition of the Markov Decision Process, detailing the simulation setup, and explaining the experimental setup. Afterward, the obtained results are presented in 3. Initially, the training process of the DQN agent is shown, followed by a comparison of the P&O algorithm and the DQN agent using Simulink simulations. Additionally, the collected experimental results are presented. Finally, conclusions and future works are outlined in Section 4.

2. Approach to deep reinforcement learning for GMPPT

In this section, we explore the overall methodology applied for utilizing DRL in the context of GMPPT. Fig. 1 illustrates the process of GMPPT extraction using our DRL methodology. A PV panel with specifications outlined in Table 1 was utilized. This panel is equipped with three bypass diodes, resulting in the appearance of three maximum power points. Fig. 2 shows the power-versus-voltage curve in the presence of PSC across different sections of the PV module, which includes the bypass diodes. It is important to note that there exist multiple LMPPs for different values of voltage, within which a GMPPT algorithm could become stuck.

To elaborate on the content presented in Fig. 1, this study references subsequent

sections outlining the Buck converter configuration, the Markov Decision Process utilized for GMPPT extraction (including detailed explanations of all equations), simulation procedures, and experimental setups. It is important to note that the study focuses on a single PV array. However, this PV array contains three inner bypass diodes, as illustrated in Fig. 3, resulting in multiple LMPPs and a single GMPP, as depicted in Fig. 2.

2.1. PV system and buck converter configuration

To conduct the current research, it was necessary to characterize a DC-DC con-verter along with a set of PV arrays. The PV module TE2200 was used for boththe simulation and experimental implementation of the system. Its specifications are outlined in Table 1. The simulation of the selected PV module accounted for the three bypass diodes, which divide the PV module into three sections. Therefore, during the system simulation, three sections of the PV module were simulated, each with the characteristics presented in Table 2.

Clarification of Key Concepts: In this study, it is essential to distinguish between converter efficiency and the tracking efficiency of the MPPT algorithm: **Converter Efficiency** refers to the ratio of the output power to the input power of the DC-DC converter, indicating how effectively the converter transforms power from the PV modules to the load. **MPPT Algorithm Tracking Efficiency** refers to the effectiveness of the MPPT algorithm in maximizing power extraction from the PV modules under varying environmental

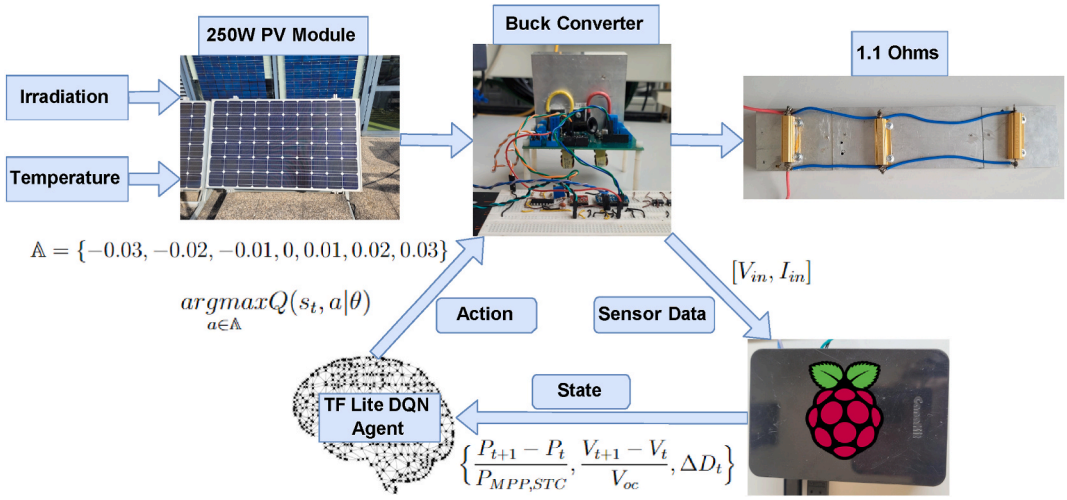


Fig. 1. Experimental set-up diagram.

Table 1
Specifications of TE2200 photovoltaic module.

Specifications	Value
Maximum Power (Wp)	250
Number of Cells	60
Open Circuit Voltage, V_{oc} (V)	37.3
Short Circuit Current, I_{sc} (A)	8.6 A
Voltage at MPP at standard test conditions, V_{mp} (V)	30.3
Current at MPP at standard test conditions, I_{mp} (A)	8.3
Temperature Coefficient of I_{sc} (%/°C)	0.065
Temperature Coefficient of V_{oc} (%/°C)	−0.32

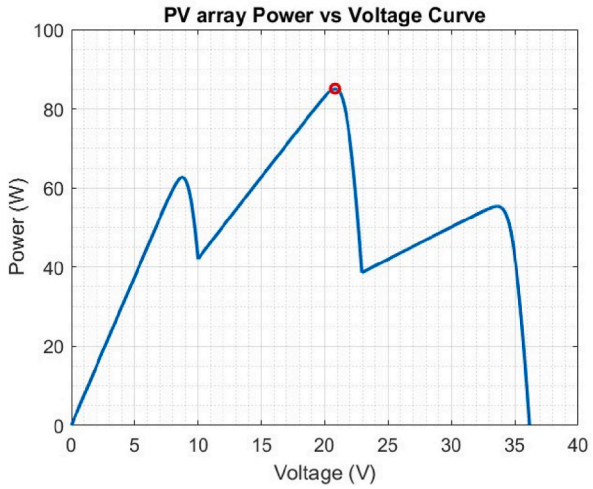


Fig. 2. PV array power curve under partial shading conditions. The red dot depicts the GMPP of the system. (For interpretation of the references to colour in this figure legend, the reader is referred to the Web version of this article.)

Fig. 3. Bypass Diodes of the PV array.

Table 2
Specifications of the simulated PV module TE2200 for each bypass diode connection.

Specifications	Value
Number of bypass diodes of PV module	3
Maximum Power per section (1 diode) (Wp)	83.3333
Number of Cells per section	20
Open Circuit Voltage per section (V)	10.1
Short Circuit Current per section (A)	8.6
Voltage at MPP per section at Standard Test Conditions (V)	12.83333
Current at MPP per section at Standard Test Conditions (A)	8.3
Temperature Coefficient of I_{sc} (%/°C)	0.065
Temperature Coefficient of V_{oc} (%/°C)	−0.32

conditions. It is calculated as the ratio of the actual power extracted by the MPPT algorithm to the theoretical maximum power available from the PV modules.

Using the chosen PV module as a basis, a synchronous buck converter was devel-oped specifically for this research. The specifications of this converter are detailed

in Table 3. The synchronous buck converter plays a crucial role in the system by efficiently stepping down the voltage to match the requirements of the PV module, thereby optimizing the power extraction process.

The equation used to calculate the input capacitance is defined as follows:

$$C_{in} \geq \frac{(1-D)*I_{mp}}{F_s*\Delta V_{in}} \tag{1}$$

where D represents the duty cycle, I_{mp} is the current at the maximum power point, F_s is the frequency at which the Pulse With Modulation is configured and ΔV_{in} is the desired variation input voltage.

The equation used to calculate the inductance of the converter is defined as follows:

$$L = \frac{D*(1-D)*V_{mp}}{F_s*\Delta I_L} \tag{2}$$

where $V\ mp$ represents the voltage at the maximum power point, and ΔI_L is the desired variation output current. The equation employed to calculate the value of the output capacitance is as follows:

$$C_{out} = \frac{\Delta V_{in}}{8*F_s*\Delta V_{out}} \tag{3}$$

where ΔV_{out} is the desired variation of the output voltage. Based on the designed converter, the DC-DC Tester at the LAAS Laboratory, Toulouse, France was used to establish the output voltage in relation to the input voltage, as follows:

$$V_{out} = V_{in}*D*\frac{R}{R_{on}+R_L+R} \tag{4}$$

where R is the load resistance, R_L is the inductance resistance and R_{on} is the MOSFET channel resistance. The efficiency of the converter was also determined, which is defined as follows:

$$\eta = \frac{R}{R_{on}+R_L+R}*\frac{1}{1+F_s\left[\frac{t_r}{D}+\frac{Q_r*R}{D^2*V_{in}}\right]} \tag{5}$$

Table 3
Specifications of the synchronous buck converter.

Specifications	Value
Frequency, F_s (kHz)	90
Duty Cycle, D (%)	60
ΔV_{in}, (V)	V_{mp} * 0.015
ΔI_L, (A)	I_{mp}/D*0.3
ΔV_{out}, (V)	V_{mp} * D * 0.0033
Inductance, L (μH)	20
Input Capacitance, C_{in} (μF)	100
Output Capacitance, C_{out} (μF)	110
Inductance Resistance, R_L (Ω)	0.01
Large-signal MOSFET channel Resistance, R_{on} (Ω)	0.036
Reverse Recovery Charge, Q_r (μC)	1.8
Reverse Recovery Time, t_r (ns)	220

where t_r and Q_r are the reverse recovery time and reverse recovery charge of the MOSFET, respectively. A series of tests were conducted to assess the efficiency of the converter. Referring to Fig. 4, it is apparent that a stable duty cycle range between 0.4 and 0.8 maintains an efficiency of up to 85 %. To facilitate these findings, the DC-DC tester from the LAAS-CNRS laboratory was employed.

2.2. Markov Decision Process for GMPPT

To implement a control system based on DRL, the first step involves defining a Markov Decision Process (MDP) model that characterizes the behavior of the PV system. MDPs serve as a formal framework for depicting sequential decision-making by an agent, where actions influence immediate rewards and subsequent states [22]. Essentially, MDPs consist of an agent and an environment. The agent interacts with the environment during each sequence of discrete time steps. At a given time step $t \in \mathrm{N}$, the agent receives a representation of the environment's state, denoted as $S_t \in \mathrm{S}$, and based on this state, it selects an action, represented as $a \in \mathrm{A}$.

At the next time step, the agent receives a numerical reward, designated as $R_t \in \mathrm{R}$, and finds itself in a new state S_{t+1}. For the implementation of DRL for GMPPT, it is essential to define the pertinent variables. Firstly, observations within a given state are defined. These observations include the difference between the current and previous time steps of power, divided by the maximum power obtained under standard test conditions ($P_{MPP,STC}$). Furthermore, the discrepancy between the current and preceding time steps of voltage, divided by the open circuit voltage (V_{oc}), is taken into consideration. Lastly, the previous perturbation of the duty cycle is factored in. In the context of time step t, the state space is delineated as follows:

$$S_t = \left\{ \frac{P_{t+1} - P_t}{P_{MPP,STC}}, \frac{V_{t+1} - V_t}{V_{OC}}, \Delta D_t \right\} \tag{6}$$

In [28], the state-space was defined as the combination of voltage, current, duty cycle, and the previous duty cycle perturbation. This definition has the disadvantage as it would require training again a model to adapt new configuration set of PV arrays. Given that our state-space features are standardized by $P_{MPP,STC}$ and V_{oc}, an agent trained for a specific PV array configuration could potentially be used to control the MPP extraction for other configurations of PV arrays. Although this concept was not tested within the scope of this paper, it holds potential for exploration in future research. A represents different perturbations of the duty cycle ΔD. The Q network is represented as $Q(s, a|\theta)$, with θ referring to the parameters of the neural network. The action at each step is calculated using an epsilon-greedy strategy as shown in Equation (8), where $c \in [0, 1]$ denotes a random number:

$$A = \{-0.03, -0.02, -0.01, 0, 0.01, 0.02, 0.03\} \tag{7}$$

$$a_{t+1} = \begin{cases} argmax Q(s_t, a|\theta)_{a \in \mathbb{A}} & if\ c \leq \epsilon \\ random(a \in A) & if\ c > \epsilon \end{cases} \tag{8}$$

The reward function is defined as follows:

$$r = r_1 + r_2 \tag{9}$$

$$r_1 = \frac{P_{t+1}}{P_{MPP,STC}} \tag{10}$$

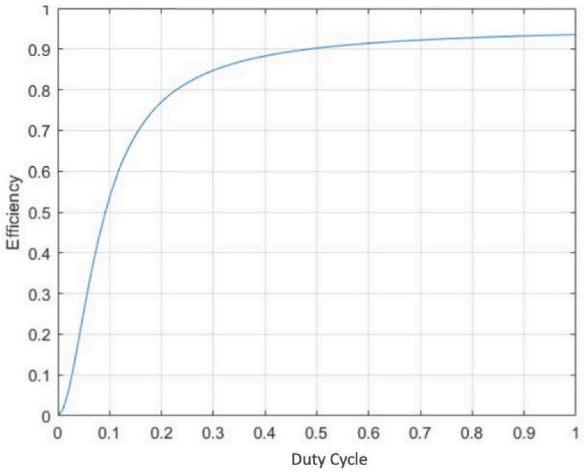


Fig. 4. Variation of efficiency in the buck converter with changing duty cycle.

$$r_2 = \begin{cases} \left(\frac{P_{t+1}}{P_{MPP,STC}}\right)^2 & if\ P_{t+1} - P_t \geq \delta_1 \\ -\left(\frac{P_{t+1}}{P_{MPP,STC}}\right)^2 & otherwise \end{cases} \tag{11}$$

In Equation (11), δ_1 stands for a small constant that permits some oscillation around the MPP achieved by the agent. The agent receives the reward r at each time step. Firstly, r_1 allows the agent to distinguish between global and local MPP by giving the higher rewards when the agent remains at higher power points. On the other hand, r_2 allows the agent gives the agent a positive rewards if there are positive increments and a negative reward otherwise.

2.3. Simulation set-up

A simulated scenario was implemented in Matlab/Simulink using the Reinforce-ment Learning Toolbox. The system operated with a time step of 0.02 s, which was defined based on the operational capacities of the Raspberry Pi 4b for both the DQN algorithm and the P&O algorithm. Each episode lasted 0.5 s and established the initial conditions of irradiance, temperature and the initial duty cycle of the system. The irradiance was determined according to the following rules; there was a 60 % chance of the episode having no PSC; a 20 % chance of one of the PV arrays being partially shaded; and another 20 % chance of two of the PV arrays experiencing partial shading. The deep neural network architecture is employed to approximate the critic agent. Each fully connected layer consists of 100 neurons, each connected to a ReLU activation function, except the last layer, which comprises one neuron with a linear activation function. To predict the action to be taken in a given time-step, each action is forwarded through out the network based on the previous state. The action that generates the highest predicted reward is selected for the next time-step, as previously showed in Equation (8). For training the neu-ral network, the Adam optimization method is used. The parameters employed for training the model are outlined in Table 4.

At each time step in the simulations, the network was trained using the stochastic gradient descent algorithm to minimize the loss function, which is defined as:

$$L(\theta) = E_{st,a_t}\left[(y_{t+1} - Q(st, a_t|\theta))^2\right] \tag{12}$$

where $Q(s_t, a_t|\theta)$ is the predicted Q value at time step t and y_{t+1} is the target Q value, defined as

$$y_{t+1} = E_{st+1}\left[r + \lambda^* maxQ(s_{t+1}, a|\theta')_{a\in\mathbb{A}}\right] \tag{13}$$

where λ is the discount factor, r is the received reward, and $max\ Q(s_{t+1}, a|\theta')_{a\in\mathbb{A}}$ is the maximum Q value at time step $t+1$ across all possible actions. To prevent the risk of catastrophic forgetting, the training algorithm employs the double DQN

approach [35], using a target network θ' to compute the target Q. The equation to update the target network parameters θ' is as follows:

$$\theta' = \tau^*\theta + (1-\tau)^*\theta' \tag{14}$$

where τ is the target smoothing factor. Further details about the DQN training algorithm in MATLAB can be found in Ref. [36].

2.4. Experimental set-up

Table 5 presents the hardware specifications of the experiment. The chosen development board was the Raspberry Pi 4 Model B, which required an Analog to Digital Converter (ADC) for reading analog data. To control the power MOSFET IRFP150N on both the low and high sides of the converter, the Half-Bridge Driver IR2104 was used. A 90 kHz Pulse Width Modulated (PWM) signal, necessary

Table 4
DQN parameters.

Parameters	Value
Discount Factor, λ	0.9
Target Smoothing Factor, τ	1e-03
Mini-Batch Size	512
Experience Buffer Length	1e6
Exploration Rate	1
Learning Rate	0.001
Optimizer	Adam
Minimum Exploration Rate	0.001
Decay of exploration rate	0.000022
Number of Episodes	5000
Sample Time, (s)	0.02
Simulation Length, (s)	0.5

for the design of the synchronous converter, was generated using the TL494 PWM control circuit. The TL494 module was controlled by a Digital to Analog Converter (DAC). The system was characterized to ensure that the TL494 produced the required duty cycle for the development board at a frequency of 90 kHz. A load resistance of 1.1 Ω was connected to the output of the converter. The schematic of the system is shown in Fig. 5.

In order to deploy the trained model on the Raspberry Pi, it was converted to the TensorFlow Lite framework through a multi-step process. Initially, the MATLAB model was converted into the Open Neural Network Exchange (ONNX) format. Following that, the model was adapted for TensorFlow and finally converted into TensorFlow Lite.

To compare the performance of the two algorithms, they were tested under differ-ent scenarios as shown in Fig. 6. The tests were conducted at LAAS-CNRS, with the PV modules inclined at 45°. Firstly, the algorithms were evaluated in an unshaded scenario. Then, they were tested in partial shading scenario 1, which covered 80 % of the surface of one of the strings connected to the first diode. This configuration was designed to create both local and global MPP that the algorithms needed to ex-ceed. Furthermore, the algorithms were tested in partial shading scenario 2. In this scenario, two sections of the module, separated by a diode, were uniformly shaded, covering 40 % of each section. Lastly, the algorithms were assessed in partial shad-ing scenario 3, where non-uniform shading covered 60 % and 40 % of two different sections of the module, leading to the presence of two LMPPs and one GMPP.

3. Simulation and experimental results

In this section, the obtained results will be presented. The section is divided as follows: First, the training results are detailed, where the DL model was trained solely using simulations. Next, the performance of the model in simulations, based on the previously trained model, is demonstrated. Finally, the performance of the model under the experimental setup is showcased.

3.1. Training results

The training results of the DQN algorithm in the simulated scenario are illus-trated in Fig. 7. The amount of data used for training the algorithm can be calculated by multiplying the number of episodes employed (5000) and the duration of each episode (0.5 s). This result is then divided by the sample time (0.02 s), yielding a total of 125,000 samples used for training the model. The acquired reward in each episode is highly dependent on the initial conditions of the episode, including irradiation and temperature received by each PV module, the presence of partially shaded modules, and the initially defined duty cycle. It is important to note that the DQN algorithm converges at around 3000 episodes and is highly influenced by the decay of the defined exploration rate. Approximately at episode 4186, the ex-ploration rate was around 0.1, allowing the algorithm to continue exploring within the maximum power point it had reached. The equation to calculate the episode at which the exploration rate arrives at a certain value, given an exploration decay d, is defined as:

$$N = \frac{1}{n} log_{1-d} \frac{E_{pn}}{E_{po}} \tag{15}$$

where n defines the steps used in each episode, E_{pn} is the desired exploration rate, and E_{p0} is the initial exploration rate as mentioned in Ref. [37].

3.2. Simulation results

The results from comparing the P&O algorithm with the DQN algorithm in sim-ulation environments are presented. Initially, the performances under standard test conditions were compared, as illustrated in Fig. 8. In this scenario, both algorithms reached the Maximum Power Point (MPP) easily, with each achieving an average power output of 231W. This indicates that under optimal conditions, both the P&O and DQN algorithms perform similarly well in extracting maximum power.

The two algorithms were subsequently tested under different partial shading (PS) scenarios. In the scenario involving shading on one PV module, as illustrated in Fig. 9, two sections of the PV module received an irradiation of 1000W/m^2 while the last section received an irradiation of 200W/m^2. Based on the presented figure, it is evident that the DQN algorithm was capable of reaching the Global Maximum Power Point (GMPP), whereas the P&O algorithm got stuck at a Local Maximum Power Point (LMPP). This is confirmed by the average power extracted by both algorithms.

Table 5
Hardware specifications.

Hardware	Specifications
Development Board	Raspberry Pi 4b
ADC	ADS1115
DAC	MCP4725
Current Sensor	LTS-25NP
PWM Generator	TL494
MOSFET	IRFP150N
MOSFET Driver	IR2104

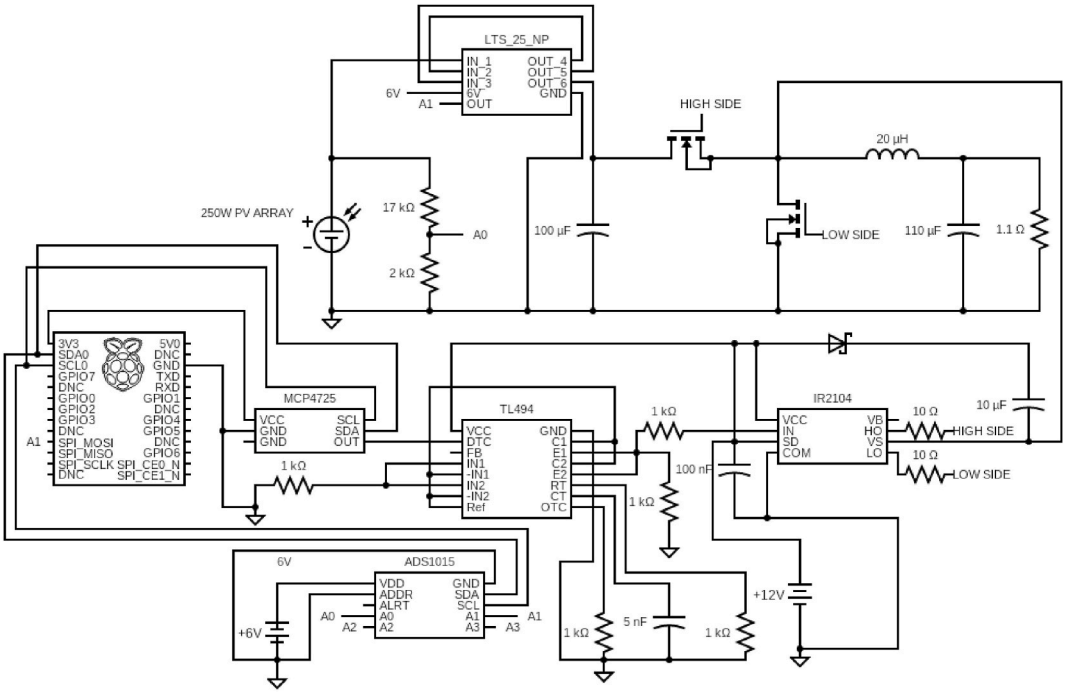


Fig. 5. Schematic circuit of buck converter and control system.

In Fig. 10, two sections of the PV module were partially shaded with different shading patterns. The capability of the DQN algorithm to reach the GMPP is also evident here, in contrast to the P&O algorithm, which converged to a LMPP.

This discrepancy allowed the DQN algorithm to achieve a higher average power extraction.

3.3. Experimental results

Firstly, a brief description of how the experiments were conducted will be pro-vided in this part. Each algorithm (DQN and P&O) was tested alternately in 1- second intervals within two different tests: one involved varying the initial duty cycle by randomly selecting it, while the other kept the initial duty cycle fixed. Power and duty cycle values were recorded at each time step (with a sample time of 0.02s) until the completion of the 1-s interval. Following the testing of both algorithms over a span of 2 s, the same experiment was repeated. This process was repeated 50 times. Irradiation was measured throughout the 50 repetitions of the experiment for each algorithm. Once all the data had been collected, the average and standard deviation of power and duty cycle were calculated for each time step. Each of the displayed graphs plots the acquired average power and duty cycle, as well as the corresponding standard deviation, represented by a shaded plot, across the 50 runs carried out for each algorithm.

3.3.1. No partial shading

Figs. 11 and 12 depict the performance of the DQN and P&O algorithms under no partial shading conditions. In both scenarios, whether a fixed duty cycle or a varying duty cycle was set, the P&O algorithm was able to achieve a higher average power throughout the experiments. Moreover, the standard deviation reached by the P&O algorithm was lower than that of the DQN algorithm, indicating better stability for the P&O algorithm throughout the experiments.

3.3.2. Partial shading scenario 1

Figs. 13 and 14 show the performance of the DQN and P&O algorithms under partial shading scenario 1, while varying and fixing the initial duty cycle for each ex-periment, respectively. Both algorithms achieved similar performance when varying the initial duty cycle of the system, as depicted in Fig. 13. However, when the initial duty cycle was set to 0.25, as shown in Fig. 14, the P&O algorithm got stuck at a Local Maximum Power Point (LMPP) over the course of the 50 runs. Meanwhile, the DQN agent managed to reach the Global Maximum Power Point (GMPP) on average throughout these 50 runs. In this scenario, the DQN algorithm was able to extract approximately 60.

3.3.3. Partial shading scenario 2

Figs. 15 and 16 show the performance of the DQN and P&O algorithms under partial shading scenario 2. It is evident that the DQN algorithm had more trou-ble reaching the GMPP when the duty cycle was varying compared to when the duty cycle was fixed. On the other hand, the P&O algorithm consistently reached the GMPP with minimal variance at each time step. However, the DQN algorithm

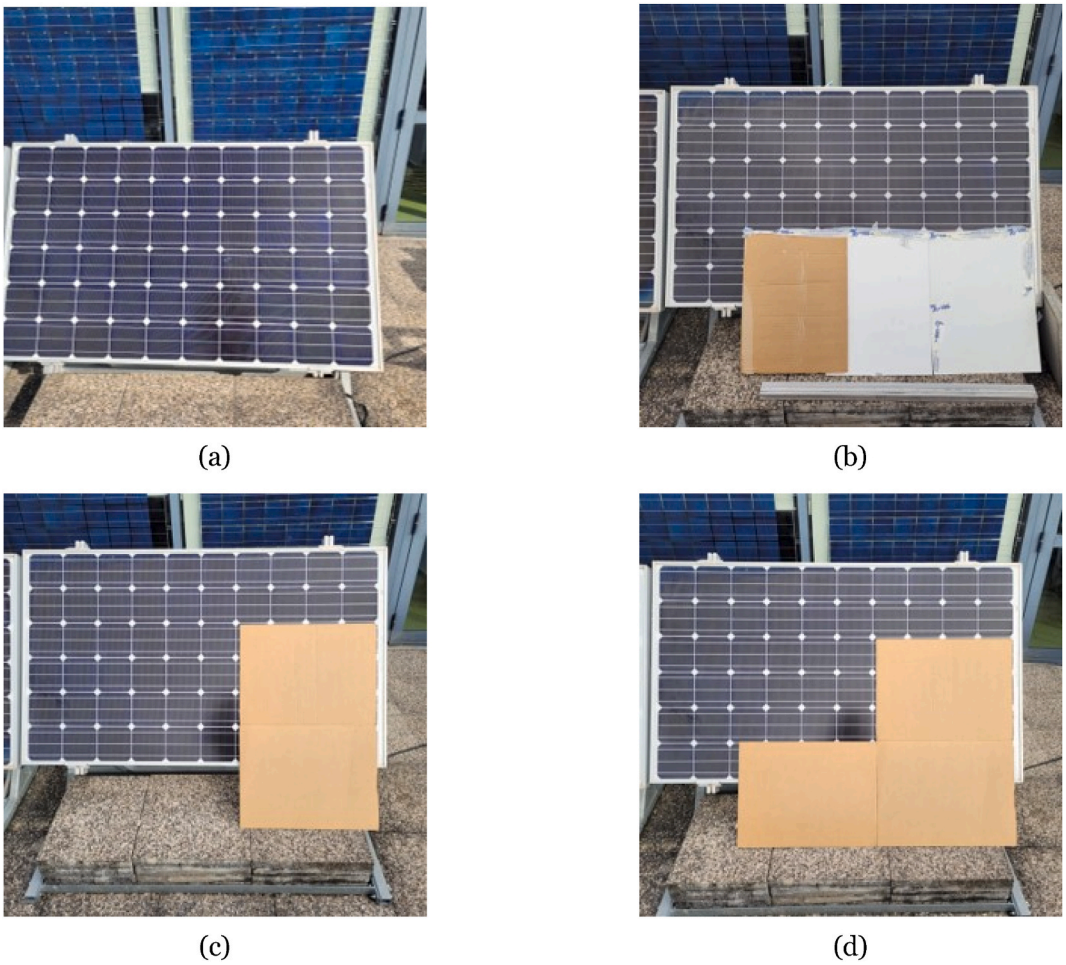


Fig. 6. Images of the experimental scenarios: (a) no partial shading; (b) partial shading scenario 1; (c) partial shading scenario 2; (d) partial shading scenario 3.

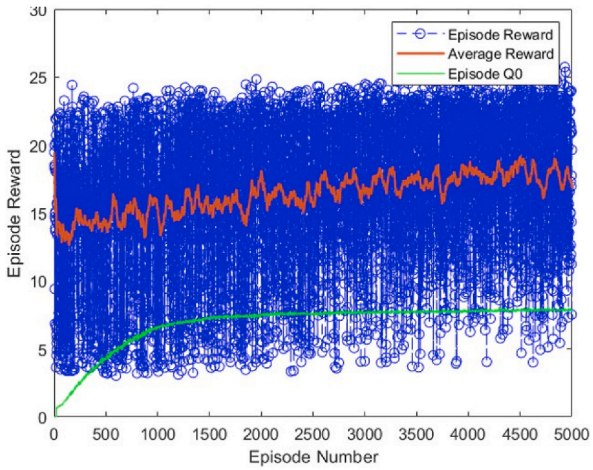


Fig. 7. Training process of DQN algorithm.

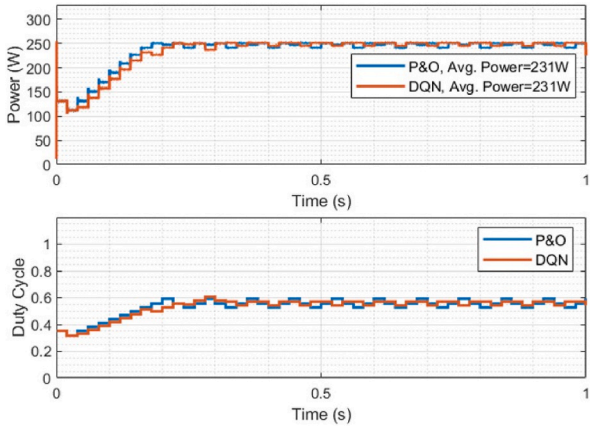


Fig. 8. Power and duty cycle of the DQN and P&O algorithms under standard test conditions. ($G1 = 1000\text{W/m}^2$, $G2 = 1000\text{W/m}^2$, $G3 = 1000\text{W/m}^2$ and $T = 25°C$).

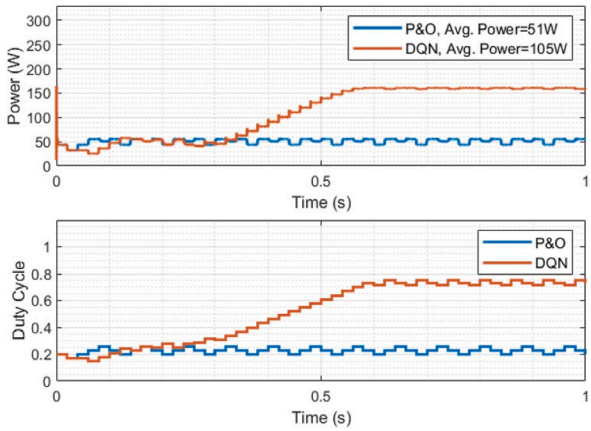


Fig. 9. Dqn algorithm training process ($G1 = 1000\text{W/m}^2$, $G2 = 1000\text{W/m}^2$, $G3 = 200\text{W/m}^2$ and $T = 35°C$).

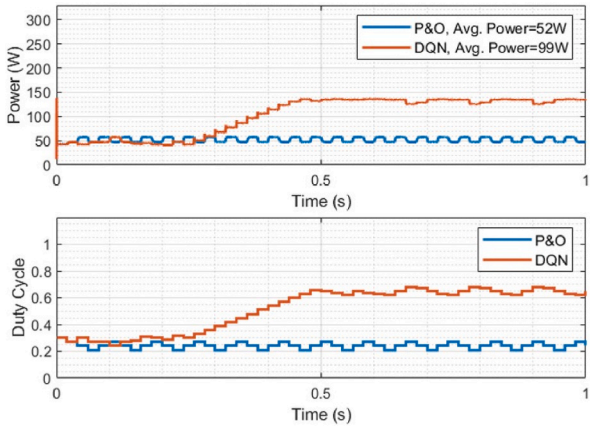


Fig. 10. Dqn algorithm training process ($G1 = 1000\text{W/m}^2$, $G2 = 800\text{W/m}^2$, $G3 = 200\text{W/m}^2$ and $T = 35°C$).

exhibited high variance at each time step, indicating that it was not able to consis-tently reach the GMPP across various scenarios. In Fig. 16, it can be observed that the stability of the DQN algorithm improved when the duty cycle was initialized at a constant value of 0.2.

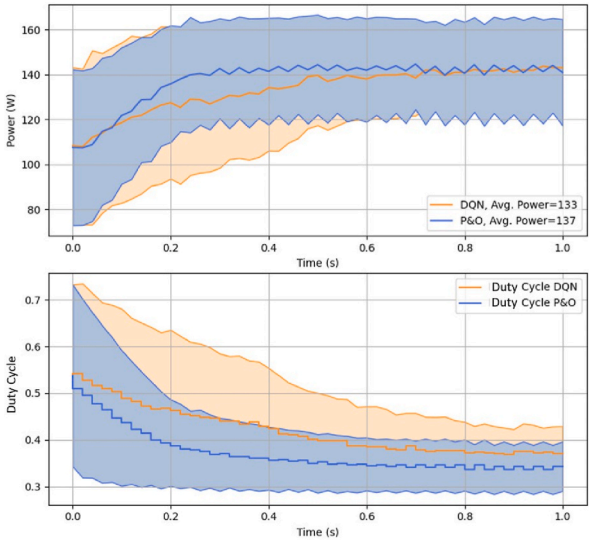


Fig. 11. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under no partial shading conditions ($G = 700 - 800\text{W/m}^2$), with random initial duty cycle averaged over 50 runs.

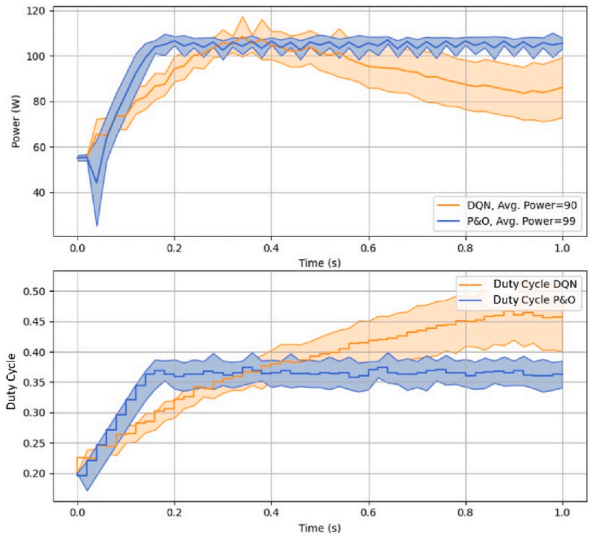


Fig. 12. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under no partial shading conditions ($G = 510 - 610\text{W/m}^2$), with fixed initial duty cycle averaged over 50 runs.

3.3.4. Partial shading scenario 3

Based on the acquired data, Table 6 presents a comparison of the power extraction percentages between the DQN agent and the P&O algorithm. A negative value indicates that the P&O algorithm extracted more power compared to the DQN agent. In the context of partial shading scenario 1, the DQN agent managed to extract over double the amount of power compared to the power extracted by the P&O algorithm. These results were observed during experiments conducted with a fixed initial duty cycle of 0.25 (see Fig. 17).

3.3.5. Analysis of experimental results

Based on the acquired data, Table 6 presents a comparison of the power ex-traction percentages between the DQN agent and the P&O algorithm. A negative value indicates that the P&O algorithm extracted more power compared to the DQN agent. In partial shading scenario 1, the DQN agent managed to extract more than twice the amount of power compared to the P&O algorithm. These results were observed during experiments conducted with a fixed initial duty cycle of 0.25.

However, the experimental results showed that the DQN algorithm was unable to surpass the P&O algorithm in multiple scenarios, even though it did so in the simulations. There could be several reasons for this discrepancy. Firstly, the sensors used in the experiments may not have been precise enough for the DQN algorithm to achieve the same results as it did in the simulations. Secondly, it is possible

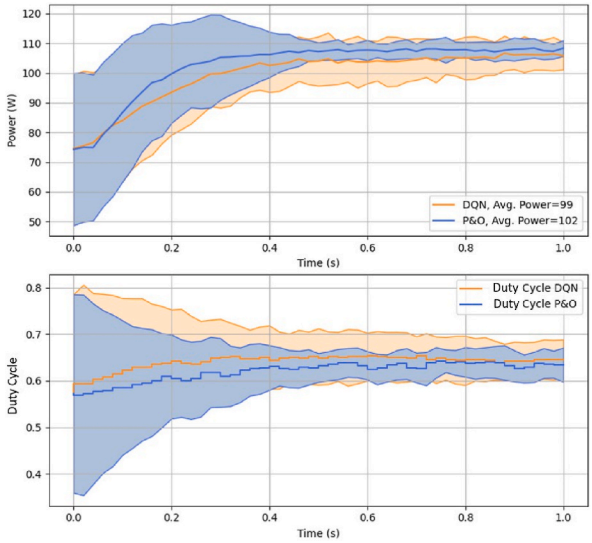


Fig. 13. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under partial shading scenario 1 ($G = 650 - 750\text{W/m}^2$), with random initial duty cycle averaged over 50 runs.

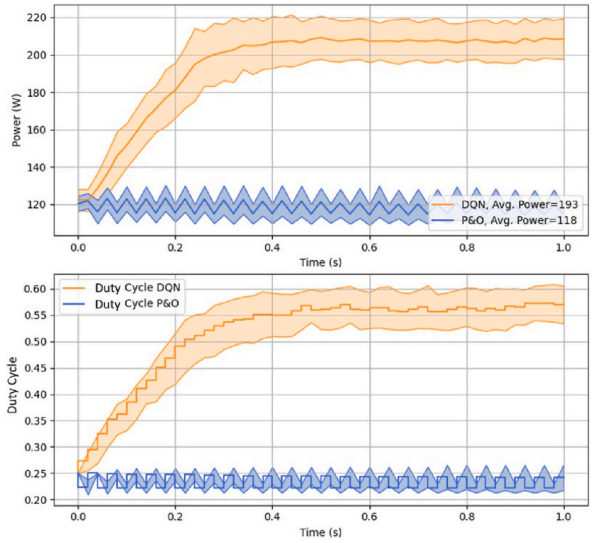


Fig. 14. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under partial shading scenario 1 ($G = 810 - 910\text{W/m}^2$), with fixed initial duty cycle averaged over 50 runs.

that the simulations in Matlab need to more accurately represent the conditions of the actual experiments (see Fig. 18).

Lastly, based on the results, it was observed that both the P&O algorithm and the DQN algorithm were able to reach the GMPP under partially shaded conditions. This implies that while the P&O algorithm got stuck at a LMPP in simulations, in practice, it managed to overcome this LMPP and achieve the GMPP, as shown in Fig. 19. The figure illustrates that around 0.3 s, the P&O algorithm almost got stuck at a LMPP but successfully continued its search, eventually reaching the GMPP.

This result might be due to the simplicity of the implemented test environment.

The experiments only used a single 250W PV module. This, combined with the noise from current sensors, could have allowed the P&O algorithm to surpass the LMPP in partially shaded scenarios and match the performance of the DQN algorithm in such situations. This might not have been the case if the environment comprised three or more PV modules, as minor fluctuations in sensor data would likely not have as significant an impact on the overall output power. Consequently, the P&O method could easily struggle to transition from a LMPP to the GMPP in more complex environments.

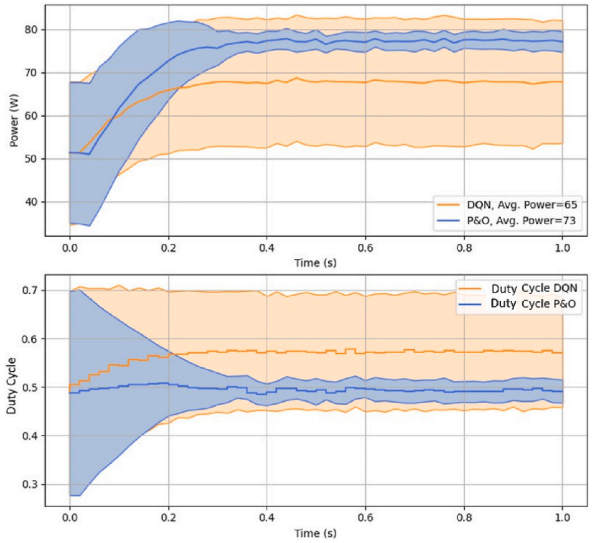


Fig. 15. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under partial shading scenario 2 ($G = 710 - 810\text{W/m}^2$), with random initial duty cycle averaged over 50 runs.

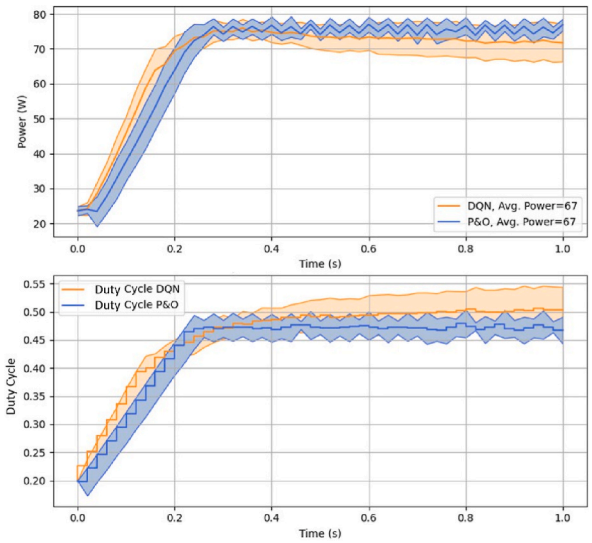


Fig. 16. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under partial shading scenario 2 ($G = 710 - 810\text{W/m}^2$), with fixed initial duty cycle averaged over 50 runs.

4. Conclusions and future research directions

In this study, a GMPPT controller based on DQN was proposed and tested against the P&O method in both simulations and real test conditions. The DQN agent outperformed the P&O algorithm in simulations. However, in real test sce-narios, the DQN did not always outperform the P&O algorithm. When the P&O algorithm got stuck at a LMPP during partial shading scenarios, the DQN algorithm was able to generate up to 63.5 % more power than the P&O algorithm. Therefore, this study not only demonstrates the potential of the application but also high-lights the need for more experimental studies involving more complex environments. These environments could better show the limitations of the P&O algorithm in ex-perimental setups compared to the DQN agent. Such environments might include the utilization of a larger number of PV modules producing at least 1000W.

Moreover, this study provides a guideline for conducting more experiments using deep RL agents for GMPPT, thanks to the availability of open-source data and the provided pipeline. The main drawbacks of the DRL methodologies are related to the amount of resources needed to train the models to obtain good results, and the need for very precise simulations that accurately depict the behavior of the entire system as it would in real-life scenarios. Additionally, these models need to be re-trained each time the configuration of the PV system changes.

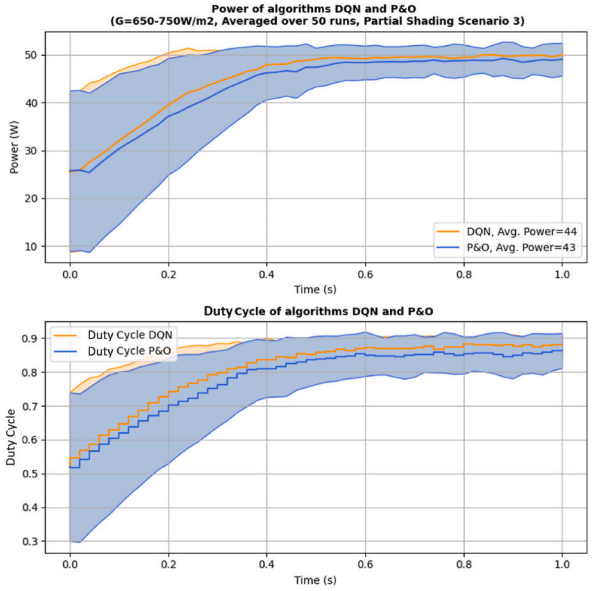


Fig. 17. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under partial shading scenario 3 ($G = 650 - 750\text{W/m}^2$), with random initial duty cycle averaged over 50 runs.

Table 6
Comparison of power extraction percentage between DQN agent when compared to P&O algorithm for each scenario.

Scenario	Random Initial Duty	Fixed Initial Duty
No Partial Shading	−2.92 %	−9.09 %
Partial Shading Scenario 1	−2.94 %	63.5 %
Partial Shading Scenario 2	−10.96 %	0 %
Partial Shading Scenario 3	2.3 %	7.4 %

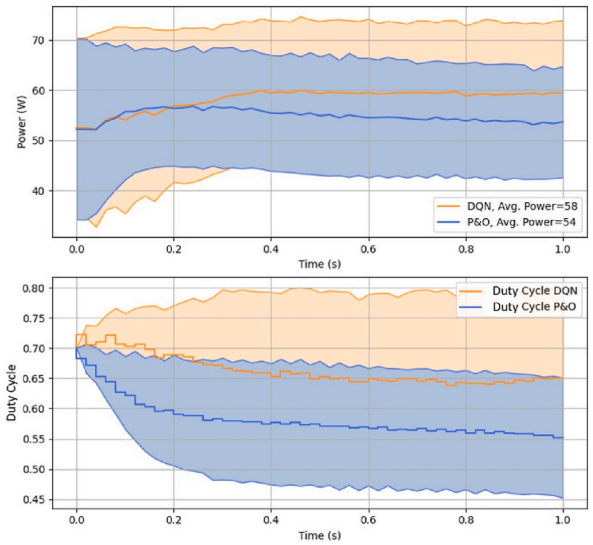


Fig. 18. Mean (solid line) and standard deviation (shaded region) of power and duty cycle, obtained using DQN and P&O under partial shading scenario 3 ($G = 700 - 800\text{W/m}^2$), with fixed initial duty cycle averaged over 50 runs.

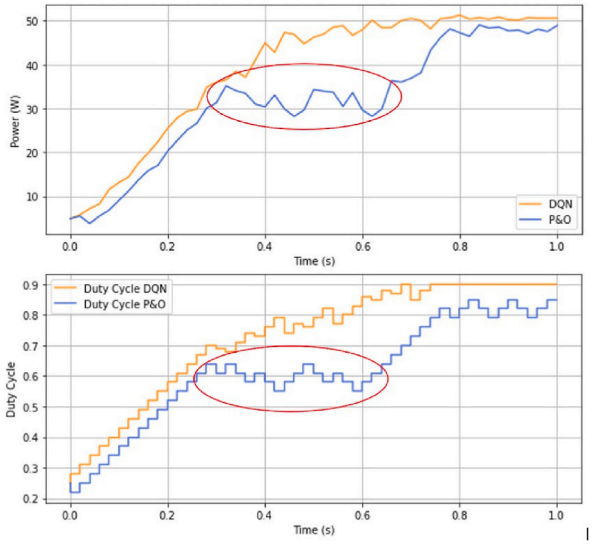


Fig. 19. Power and duty cycle of DQN and P&O under partial shading scenario 3 ($G = 650 - 750\text{W/m}^2$) with a fixed initial duty cycle.

Future research could focus on testing different deep RL agents, including the Soft-Actor-Critic, Deep Deterministic Policy Gradient, and Twin-Delayed Deep De-terministic Policy Gradient agents, within actual testing scenarios. The use of an emulator for partially shaded PV systems, similar to the one presented in Ref. [38], would be useful for algorithm testing in more controlled PS scenarios. Furthermore, using the proposed pipeline enables the evaluation of alternative embedded systems to deploy trained RL agents for real-time decision-making.

Data availability statement

Datasets:. The data that support the findings of this study are openly available in https://github.com/SmartSystems-UniAndes. This link includes.

- Schematic and Layout of the Buck Converter, along with the training pro-cedure of DQN model and conversion to TFLite file: https://github.com/SmartSystems-UniAndes/Train_and_Convert_RL_MPPT
- Deployment of the DQN model to Raspberry PI 4b: https://github.com/SmartSystems-UniAndes/Reinforcement_Learning_MPPT_RaspberryPi_Deploy

Funding statement

All authors acknowledge financial support provided by the Vice Presidency of Research & Creation publication fund at the Universidad de los Andes.

CRediT authorship contribution statement

Luis Felipe Giraldo: Writing – review & editing, Supervision. **Jorge Felipe Gaviria:** Writing – original draft, Validation, Resources, Methodology, Conceptualization. **María Isabella Torres:** Writing – review & editing, Investigation, Formal analysis. **Corinne Alonso:** Writing – review & editing, Supervision. **Michael Bressan:** Writing – review & editing, Validation, Methodology.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] H. Ritchie, M. Roser, P. Rosado, Renew. Energy (2022) [Online]. Available: https://ourworldindata.org/renewable-energy.
[2] G. Data. Global solar photovoltaic (PV) market update, 2019 with historic (2006-2018) and forecast (2019-2030). [Online]. Available: https://www.businesswire.com.
[3] M.E. El Telbany, A. Youssef, A.A. Zekry, Intelligent techniques for MPPT control in photovoltaic systems: a comprehensive review, in: 2014 4th International Conference on Artificial Intelligence with Applications in Engineering and Technology, IEEE, 2014, pp. 17–22, ieeexplore.ieee.org/document/7351807/.

[4] F. Liu, S. Duan, F. Liu, B. Liu, and Y. Kang, "A variable step size INC MPPT method for PV systems," IEEE, vol. 55, no. 7, pp. 2622–2628, 2008, conference Name: IEEE Transactions on Industrial Electronics.

[5] M.S. Hossain, H. Mahmood, Short-term photovoltaic power forecasting using an LSTM neural network and synthetic weather forecast, IEEE 8 (2020) 172 524–172 533, ieeexplore.ieee.org/document/9200614/.

[6] E. Miranda, J.F.G. Fierro, G. Narváez, L.F. Giraldo, M. Bressan, Prediction of site-specific solar diffuse horizontal irradiance from two input variables in Colombia, Heliyon 7 (12) (2021) e08602 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2405844021027055.

[7] A.F. Zambrano, L.F. Giraldo, Solar irradiance forecasting models without on-site training measurements, Renew. Energy 152 (2020) 557–566 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148116307297.

[8] G. Narvaez, L.F. Giraldo, M. Bressan, A. Pantoja, Machine learning for site-adaptation and solar radiation forecasting, Renew. Energy 167 (2021) 333–342 [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0960148120318395.

[9] A. Rico Espinosa, M. Bressan, L.F. Giraldo, Failure signature classifica- tion in solar photovoltaic plants using RGB images and convolutional neural networks, Renew. Energy 162 (2020) 249–256 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148120312301.

[10] M. Danandeh, S. Mousavi G, Comparative and comprehensive review of maximum power point tracking methods for PV cells, Renew. Sustain. Energy Rev. 82 (2018) 2743–2767 [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1364032117313813.

[11] O.F. Tozlu, H. Calik, A review and classification of most used MPPT algorithms for photovoltaic systems, Hittite Journal of Science and Engineering 8 (3) (2021) 207–220, number: 3. [Online]. Available: https://dergipark.org.tr/en/pub/hjse/issue/65166/888919.

[12] M. Bressan, Y. El Basri, A.G. Galeano, C. Alonso, A shadow fault detection method based on the standard error analysis of i-v curves, Renew. Energy 99 (2016) 1181–1190 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148116307297.

[13] H.-D. Liu, C.-H. Lin, K.-J. Pai, C.-M. Wang, A gmppt algorithm for preventing the lmpp problems based on trend line transformation technique, Sol. Energy 198 (2020) 53–67 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0038092X20300566.

[14] M.A. Husain, A. Jain, A. Tariq, A. Iqbal, Fast and precise global maximum power point tracking techniques for photovoltaic system, IET Renew. Power Gener. 13 (14) (2019) 2569–2579, ietresearch.onlinelibrary.wiley.com/doi/abs/10. 1049/iet-rpg.2019.0244.

[15] M. Naseem, M.A. Husain, J.D. Kumar, A.F. Minai, A. Ahmad, S.M. Ali, A.S. Ansari, A spider monkey optimization based global maximum power point tracking technique for photovoltaic systems, in: 2022 2nd International Conference on Emerging Frontiers in Electrical and Electronic Technologies (ICEFEET), 2022, pp. 1–6.

[16] S.D. Al-Majidi, M.F. Abbod, H.S. Al-Raweshidy, A particle swarm optimisation-trained feedforward neural network for predicting the maximum power point of a photovoltaic array, Eng. Appl. Artif. Intell. 92 (2020) 103688 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197620301238.

[17] M.A. Husain, S.B. Pingale, A. Bakar Khan, A. Faiz Minai, Y. Pandey, R. Shyam Dwivedi, Performance analysis of the global maximum power point tracking based on spider monkey optimization for pv system, Renewable Energy Focus 47 (2023) 100503 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1755008423000996.

[18] D.K. Kishore, M. Mohamed, K. Sudhakar, K. Peddakapu, Swarm intelligence-based mppt design for pv systems under diverse partial shading conditions, Energy 265 (2023) 126366 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544222032522.

[19] B. Yang, S. Wu, J. Huang, Z. Guo, J. Wang, Z. Zhang, R. Xie, H. Shu, L. Jiang, Salp swarm optimization algorithm based mppt design for pv-teg hybrid system under partial shading conditions, Energy Convers. Manag. 292 (2023) 117410 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0196890423007562.

[20] I. Sajid, A. Sarwar, M. Tariq, F.I. Bakhsh, S. Ahmad, A. Shah Noor Mohamed, Archimedes optimization algorithm (aoa)-based global maximum power point tracking for a photovoltaic system under partial and complex shading conditions, Energy 283 (2023) 129169 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S036054422302563X.

[21] E. Artetxe, J. Uralde, O. Barambones, I. Calvo, I. Martin, Maximum power point tracker controller for solar photovoltaic based on reinforcement learning agent with a digital twin, Mathematics 11 (9) (2023) [Online]. Available: https://www.mdpi.com/2227-7390/11/9/2166.

[22] R.S. Sutton, A.G. Barto, Reinforcement learning: an introduction. Ser. Adaptive Computation and Machine Learning Series, second ed., The MIT Press, 2021.

[23] P. Kofinas, S. Doltsinis, A. Dounis, G. Vouros, A reinforcement learning approach for MPPT control method of photovoltaic sources, Renew. Energy 108 (2017) 461–473 [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0960148117301891.

[24] B. Aurobinda, B. Subudhi, P. Kumar, A combined reinforcement learning and sliding mode control scheme for grid integration of a PV system, IEEE (2019) [Online]. Available: https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8928283.

[25] K. Bavarinos, A. Dounis, P. Kofinas, Maximum power point tracking based on reinforcement learning using evolutionary optimization algorithms, Energies 14 (2) (2021) 335, number: 2, Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1996-1073/14/2/335.

[26] Y. Singh, N. Pal, Reinforcement learning with fuzzified reward approach for mppt control of pv systems, Sustain. Energy Technol. Assessments 48 (2021) 101665 [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2213138821006792.

[27] M. Arianborna, J. Faiz, A. Erfani-Nik, Mppt control of a pmsg connected to the wind turbine based on deep q-network, in: 2023 10th Iranian Conference on Renewable Energy Distributed Generation (ICREDG), 2023, pp. 1–5.

[28] B.C. Phan, Y.-C. Lai, C.E. Lin, A deep reinforcement learning-based MPPT control for PV systems under partial shading condition, Sensors 20 (11) (2020) 3039, number: 11 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/1424-8220/20/11/3039.

[29] L. Avila, M. D. Paula, I. Carlucho, and C. S. Reinoso, "MPPT for PV systems using deep reinforcement learning algorithms," IEEE, vol. 17, no. 12, pp. 2020–2027 2019, conference Name: IEEE Latin America Trans- actions. [Online]. Available: https://www.research.ed.ac.uk/en/publications/mppt-for-pv-systems-using-deep-reinforcement-learning-algorithms.

[30] L. Avila, M. De Paula, M. Trimboli, I. Carlucho, Deep reinforcement learning approach for MPPT control of partially shaded PV systems in smart grids, Appl. Soft Comput. 97 (2020) 106711 [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1568494620306499.

[31] OpenAI. Gym: A toolkit for developing and comparing reinforcement learning algorithms.,[Online]. Available: https://gym.openai.com.

[32] H. Naseem M, Assessment of meta-heuristic and classical methods for gmppt of pv system, Transactions on Electrical and Electronic Materials 22 (2021) 217–234 [Online]. Available: https://link.springer.com/article/10.1007/s42341-021-00306-3#citeas.

[33] J.F. Gaviria, G. Narváez, C. Guillen, L.F. Giraldo, M. Bressan, Machine learning in photovoltaic systems: a review, Renew. Energy (2022) [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0960148122009454.

[34] M. Glavic, (deep) reinforcement learning for electric power system control and related problems: a short review and perspectives, Annu. Rev. Control 48 (2019) 22–35 [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S1367578819301014.

[35] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell, "Overcoming catastrophic forgetting in neural networks," Proc. Natl. Acad. Sci. USA, vol. 114, no. 13, pp. 3521–3526 2017, publisher: Proceedings of the National Academy of Sciences. [Online]. Available: https://www.pnas.org/doi/10.1073/pnas.1611835114.

[36] MathWorks. Deep q-network (DQN) agents - MATLAB & simulink.,[On- line]. Available: https://www.mathworks.com/help/reinforcement-learning/ug/dqn-agents.html.

[37] MATLAB. Options for q-learning agent - MATLAB - MathWorks, [Online]. Available: https://www.mathworks.com/help/reinforcement-learning/ref/rlqagentoptions.html.

[38] M. Bressan, A. Gutierrez, L. Garcia-Gutierrez, C. Alonso, Development of a real-time hot-spot prevention using an emulator of partially shaded PV systems, Renew. Energy 127 (2018) [Online]. Available: https://hal.archives-ouvertes.fr/hal-01962921.