Electrical Engineering (2022) 104:4213–4225
https://doi.org/10.1007/s00202-022-01615-3

ORIGINAL PAPER

Reinforcement-learning-based damping control scheme of a PV plant in wide-area measurement system

Ismael Abdulrahman[1]

Received: 1 January 2022 / Accepted: 16 July 2022 / Published online: 7 August 2022

Abstract
Modern power systems are witnessing a noticeable increase in the integration of low-inertia renewable sources which require robust control schemes to damp out low-frequency oscillations emerged by this expansion. This paper proposes a reinforcement learning (RL)-based controller using a deep deterministic policy gradient (DDPG) algorithm to damp inter-area oscillations. The learning process of the controller is enhanced using a discrete reward-function which is selected to be a reciprocal function of the input error. To allow the agent to drive the total error lower and lower, both the absolute error and integral of error are included in the observation state vector. A two-area system with a solar plant integrated is used as the test system. The controller obtains its global input signal from PMU devices in wide-area measurement systems using frequency information. A comprehensive analysis is presented using several analytical control tools including time-domain simulation, pole-zero plot, mode shape, frequency response, and participation factor map. Furthermore, a package of programs has been developed for this study using MATLAB and Simulink. The communication latency is also included in the design of the controller considering constant and variable practical values of latency. The proposed controller demonstrates its effectiveness in damping inter-area oscillation and improving the system stability.

Keywords Reinforcement learning · Wide-area measurement systems · DDPG · Solar plant · Inter-area oscillations

1 Introduction

With the rapid expansion in the size of interconnected power systems, high penetration of renewable energy, and integration of low-inertia sources and flexible loads, inter-area oscillations problem becomes one of the major concerns in the stability of power systems. It reduces the maximum transfer capability of transmission lines and involves many parts of the power system contributing to this instability [1, 2]. Such an interconnected system requires effective control strategies to damp out inter-area oscillations and enhance power system security. Wide-area measurement system (WAMS) provides controllers with global information obtained from phasor measurement units (PMUs) with time-stamped and high sampling rates.

The control techniques used in power systems can be divided into model-based and model-free approaches. In the model-based systems, the controller is represented by the mathematical equations of the model. Design of a model-based controllers for complex systems with uncertainties requires physical model and structure detail which is a challenging task for interconnected systems with large state and action spaces [1]. On the other hand, a model-free controller is an intelligent system with parameters optimized by mapping and learning from its input–output data. Model-free controllers do not require internal knowledge or mathematical equations of the model because it is based on measurements and information collected online. For this reason, model-free controllers can be employed to solve control problems in large-scale complex power systems. Advanced sensors in a smart grid produce big data and useful information that can be utilized to build intelligent agents to control shortcomings in the system [3]. In the design of these controllers, it is possible to include many scenarios and operation conditions to gain experience from the data and make decisions. Machine learning (ML)-based controller is the state-of-the-art technique for model-free controllers. ML can

✉ Ismael Abdulrahman
ismael.abdulrahman@epu.edu.iq

[1] Department of Technical Information Systems Engineering, Erbil Technical Engineering College, Erbil Polytechnic University, Erbil 44001, Kurdistan region, Iraq

be classified into three branches: supervised, unsupervised, and reinforcement learning (RL). The first two techniques are mainly used for clustering and labeling static data, respectively, whereas the latter is used in dynamic environments. The aim of RL is to generate actions based on state observations and received rewards through continuous interaction with the environment [4, 5].

On the applications of RL in power systems, there are several excellent recent works presented in [1, 6]–[12]. These researches proposed intelligent controllers based on the RL approach to solve low-frequency oscillations, in particular inter-area oscillations. Reference [1] proposes a robust WAMS controller based on policy gradient learning to adjust the field voltages of multiple synchronous generators. Several remote and local measurements are taken for the reward function including speed deviation, sustaining relative speed changes, and voltage phase angles difference of remote buses. The controller's design is complex; it requires multiple coordinated controllers with several local and remote signals per each to solve a single problem which is inter-area oscillations. In addition, the study focuses on the side of synchronous generators while there are two solar plants integrated to the two-area system with no damping control-scheme shown. Reference [6] proposes a multi-band power system stabilizer using deep reinforcement learning, and the controller parameters are tuned using the proximal policy optimization (PPO) technique. However, the system is considered to involve only conventional generators, and the controllers input signals are local measurements including rotor angle, active power, and voltage magnitude. Some other recent studies proposed an RL approach to solve load frequency control [13]–[16], whereas [3, 17]–[21] listed the most recent studies in power system control using RL. Among all these studies, a few studies worked on inter-area oscillations in systems that are integrated with renewable energy sources (RESs). Low-inertia resources decrease damping in the system and hence increase instability, complexity, and uncertainty.

This paper presents a reinforcement learning control to damp out inter-area oscillations using DDPG approach. The controller is installed at the no-inertia side where a solar plant is installed with remote-signal input obtained from WAMS. A two-area system is used as the test system with all PSSs removed to show the contribution of the proposed controller in damping the oscillations. The system is comprehensively analyzed and simulated considering time latency by using a set of programs developed for this purpose. The paper tackles the low-frequency oscillation using the combination of the state-of-the-art technologies of WAMS, machine learning, and green energy represented by a photovoltaic plant without the need of additional damping controllers such as stabilizers. In addition, the proposed controller uses two error signals to quickly correct the given action. Since the controller uses global measurement that might witness communication delay, the controller includes a wide practical range of latency in its design. All the programs, codes, algorithms, and figures were created by the author for this study.

The rest of this paper is organized as follows. Section II describes the control scheme of the RL agent. The problem statement is presented in Section III whereas the RL setup is introduced in Section IV. Section V and VI describe the test system and the programs developed for this study, respectively. The results and discussion are given in Section VII, and finally, the conclusion is presented in Section VIII.

1.1 Reinforcement-leaning control scheme

In conventional feedback control, designers usually use adaptive or optimal control. The main difference between these two techniques is the way the controller's parameters are tuned: online or offline. Adaptive control tunes the unknown parameters using online real-time measurements but they are not optimized [22]. Optimal control, on the other hand, is an offline control with parameters optimized using a mathematical equation but it requires modeling the dynamic system. Reinforcement learning (RL) is a machine learning approach to learn the behavior of a system from the plant's input–output data to design an adaptive-optimized controller. The method does not require modeling the plant, and the controller's parameters are tuned through a data learning process.

RL has two main components: agent and environment. Agent is the controller to be designed. Environment is the whole system excluding the controller, in other words, it is the system plant. There are two directed signals connecting the agent and environment forming a closed loop: (1) an outgoing signal from the agent to the environment which represents the controller action (2) another two outgoing signals from the environment to the agent which takes the output of the plant as observation states (feedback signal) adding to it a reward function signal. The reward signal is used to reinforce the goodness of the agent's action over time. Inside the agent block, we observe two components: actor and critic. These are two neural networks we want to optimize their parameters (weights and biases) to gain maximum learned information about the behavior of the plant. Figure 1 visualizes this description, while Fig. 2 shows the same description implemented in Simulink.

The agent uses an algorithm or technique to make the final policy for the controller. One of the popular algorithms in RL agent is deep deterministic policy gradient (DDPG). It is a model-free, actor-critic, off-policy-based approach that is known for its working space in both continuous and discrete action domain. In the DDPG algorithm, two models are used: actor and critic. In the actor model, the state is taken as the input whereas the action is the output of the model. Given a state, the actor proposes an action for the agent. Sometimes,

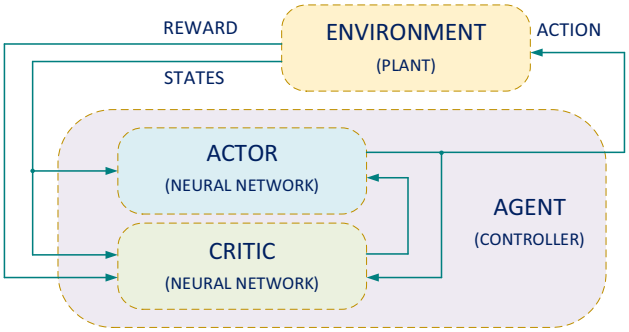

Fig. 1 The overall RL agent-environment model

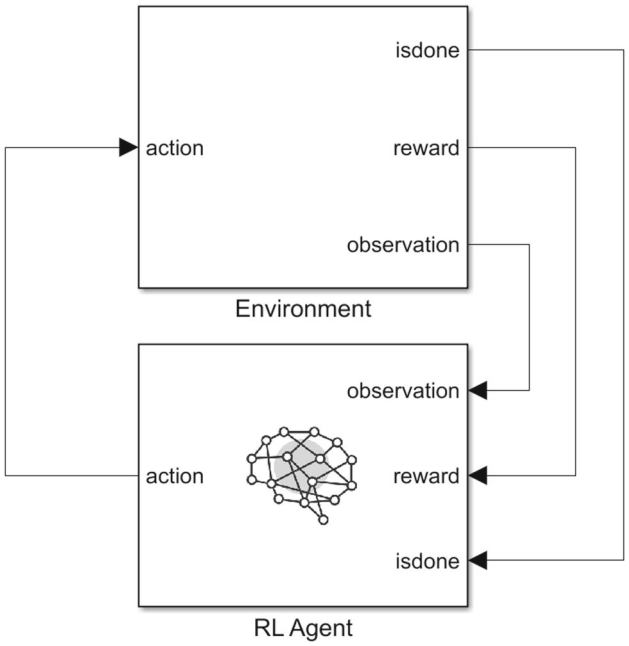

Fig. 2 Simulink diagram of agent-environment model

we call the agent a learner, a decision maker, or simply a controller. In the critic model, both the state and action are used as the input signal, whereas Q-value is the output of the network. This Q-value is used to predicts whether the action proposed by the actor is good or bad depending on the sign and value of Q-value. Policy maker or behavior-function are two different names used for the actor-function. Policy evaluator or Q-function are also two different terms used for the critic function.

1.2 Problem statement

The problem in this research can be stated as follows:

- Damp inter-area oscillations by adding a stabilizing signal to the active control loop of the solar plant. Use information from WAMS that alarms us about this oscillation when it occurs.
- We are given the environment X that represents the entire system excluding the controller. X includes the state s_t and communication delay T_d.
- Our target is to find an agent with a policy π and action a_t that adjusts the control loop to damp down these oscillations.

1.3 RL setup

1.3.1 RL agent setup

To effectively damp inter-area oscillations, an RL-based controller is proposed in this study with a remote signal obtained from wide-area measurements system. Since the agent's action is a function of its input, certain measurements are selected as observation states in order to maximize the RL's knowledge on these oscillations once occurred. Based on the definition of inter-area oscillation where a group of machines in one area swings against another groups of machines in another area, a sum of measurements from each area is chosen to form a center-of-inertia difference between areas. The signal used for this concept is the machine frequency obtained from PMU owing to its key indicator and sensitivity to such changes in the system. Figure 3 shows the Simulink setup of the entire system including the RL agent with two inputs (observation and reward) in addition to the termination action for the episodes shown as "isdone". In the following sections, we will discuss the agent's input signal, actor-critic networks, and the plant.

1.3.2 Observation signal

In control system, we observe a feedback signal from the output of plant to the input reference. Similarly, to design an RL controller, we use a vector of states as an observation signal from the output of the plant (here, the center of inertia difference of frequency deviations) to the input forming an error input-signal as shown in Fig. 3. In addition to this error, we use the integral of error to add memory to the error over

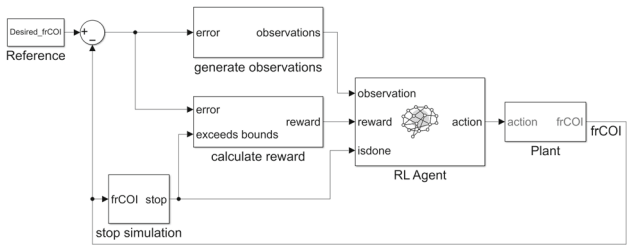

Fig. 3 Control scheme of agent-environment using Simulink

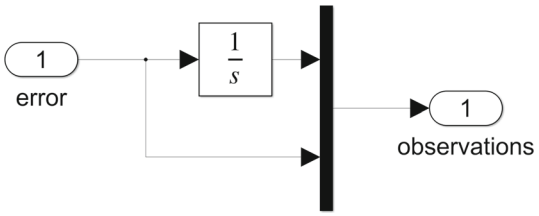

Fig. 4 Observation signal

time and drive the total error under the curve lower and lower as shown in Fig. 4.

1.3.3 Reward function

One of the main challenges in training an RL agent is its learning speed which usually lasts for several hours to gain experience from the data and make a good policy for the agent. To reduce the training time, one way is to focus is on the reward function. The agent needs a well-defined reward function so that it accelerates the convergence speed of learning. From Fig. 3 which shows a tracking control scheme in the form of agent-environment coupling, we can give the agent a reward value reciprocal to the absolute error. The lower the error, the higher the reward value, and vice-versa. By doing so, we try to minimize the error and let the agent learn to track the reference and drive this error to the minimum. In this study, a discrete reciprocal reward-function with 0.1 step is used for the absolute error between 0 and 1. If the error is beyond this value, we use a penalty with a negative value (-10). If the error is zero or very close to zero, we add a small value to the error to avoid division by zero. The function is plotted and shown in Fig. 5 which shows how the reward value reduces as the error increases.

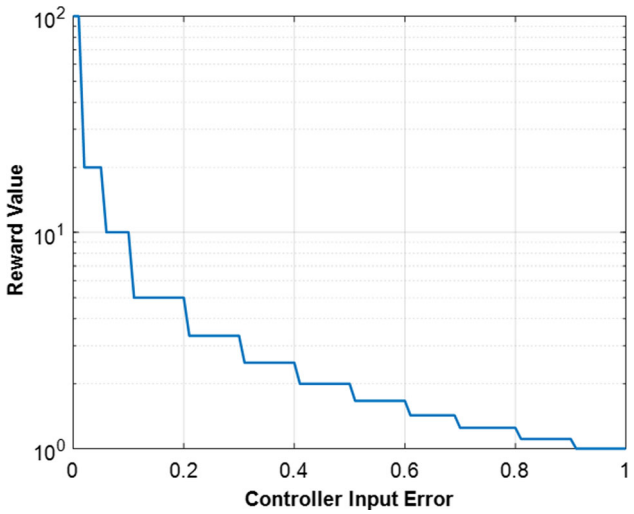

Fig. 5 Reward function

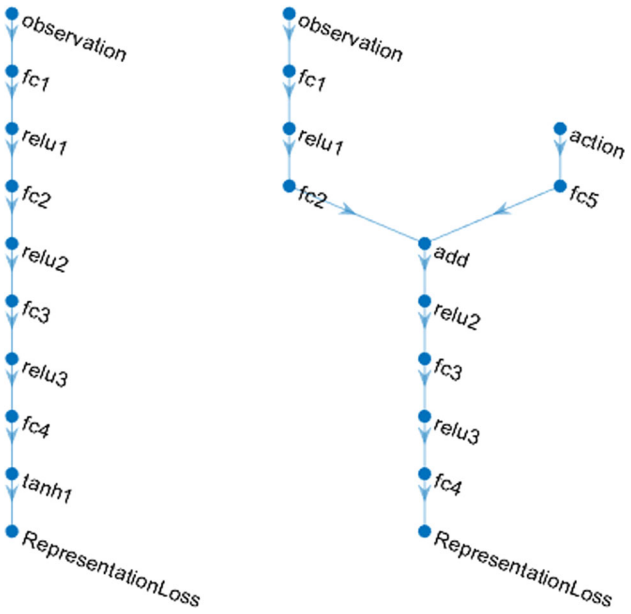

Fig. 6 Actor network (left) and critic network (right)

1.3.4 Actor-critic network

To create a DDPG agent and approximate the average long-term reward, two deep neural networks are designed for the actor and critic models. Number of neurons and hidden layers are chosen empirically and experimentally with all having 50 neurons and fully-connected layers as shown in Fig. 6. The actor has state vectors composed of the error and integral of error with three hidden layers. The critic network takes the input and output of the actor as its two inputs and it consists of five hidden layers. To increase the learning process, all layers, except the actor output, are normalized to the range of [0 1] using the rectified linear activation function "$relu$". The actor output is connected to both the environment block and critic network and is normalized to the range of $[-1\ 1]$ with a hyperbolic tangent activation function "$tanh$". The two networks are weighted initially with a nonzero small matrix.

1.3.5 Training settings

The agent is trained using the following settings. The action signal is saturated by the generator capacity measured in pu. The sampling time for this training is chosen to be 0.5 s over 10 s simulation time. To get the same training results when the program is simulated in the next time, a reset function is used namely "rng" with zero value in MATLAB. Maximum episode value is chosen based on the ratio of simulation time to sampling time rounded to the nearest integer value which is 20 episodes in this study. This is also the maximum steps per episode. The training is stopped after 1475 s when the episode

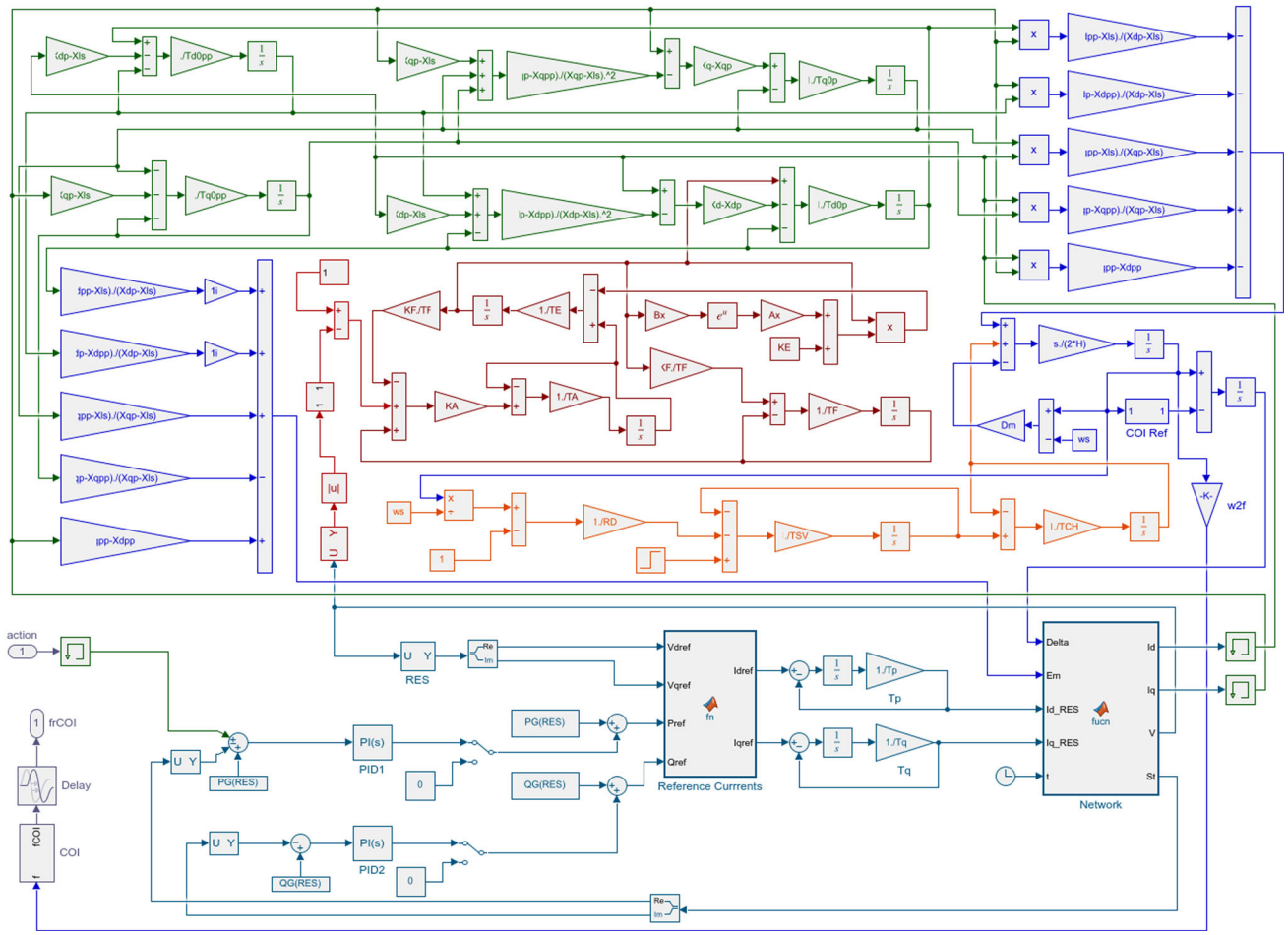

Fig. 7 The environment (plant) diagram implemented in Simulink (created by the author for this study)

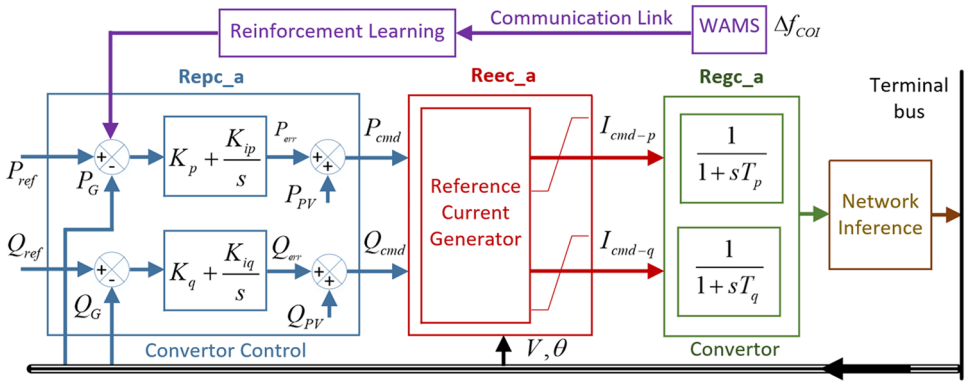

Fig. 8 PV control loops

reward reaches a value with a satisfied damping which is evaluated to be 20 for the reward function defined in this study. Other training setting is the discount factor which is a value in the range of [0 1] used to describe the importance of future rewards to the current value. The discount factor for this work is set to 0.995. Another important factor in training an RL is the noise variance and its decay rate which are selected to be 0.3 and 0.001, respectively. The noise is added to the input data to reduce overfitting– that is, to reduce memorizing the given dataset by the network especially when the data sample is small. As a result, the error generalization and mapping process are improved.

1.3.6 Environment setup

The agent is designed to control a problem in the environment (plant) which is inter-area oscillations. In this study, our environment is the power system excluding the agent. Since our agent is designed in Simulink, we use the same platform to model the power system based on the programs developed in [2]. In this paper, the detailed sixth-order model of the generator is employed and the machines are equipped with excitor and turbine systems as shown in Fig. 7. The mathematical model of the system represented by differential algebraic equations (DAEs) are presented in [23] with calculation of the initial conditions of all variables and states. The PV plant is modeled based on the two-control-loops model introduced by WECC Renewable Energy Modeling Task Force [24]–[26] as shown in Fig. 8. This model is widely-used by researchers and is added to the libraries of most professional software for dynamic analysis of large-scale power systems. Loads are modeled using a combination of constant power, constant voltage, and constant impedance. A reference is necessary to be assigned for all phase angles in the system which can be achieved by subtracting speeds from either one of the machine's speed or the center of inertia of all machines [27]. For the linearization, it is required to highlight the input and output signals of the system which are chosen to be the voltage reference and actual bus voltage of machine 1, respectively.

Figure 9 shows the overall RL agent-environment layout with internal connections and detail structure of neural networks representing the actor (red) and critic (green). The nodes\edges of the two networks are displayed in circles\lines, respectively, with the activation functions ($tanh$ and $relu$).

1.4 Test system

A two-area test system with a PV plant installed at one side is used to train and verify the proposed agent. The controller is installed at the solar plant [2] to add additional damping amount to the active power loop and damp out the oscillations. The original data for this test system is obtained from [28]. The PSSs are removed from the system to observe the impact of the controller on damping the oscillations and filter out other effects. A total of 400/900 pu active power is planned by the original study to flow through the tie-line. The total generation from the synchronous machines in Area 1 is reduced by 200/900 pu and is compensated by the solar plant in the same area. The power flow remained unchanged with this change. The single line diagram of the system with the PV integrated to the grid is shown in Fig. 10.

1.5 Development of MATLAB \ Simulink programs

To investigate the problem, integrate the PV plant, design the controller, simulate the system in time-domain and small-signal analysis, a set of programs and tools were developed by the author using MATLAB\Simulink. The block diagram for the base program (plant) is shown in Fig. 7 where each part of the system including the synchronous machines, excitors, turbines, and solar plant, is distinguished and given a different color. The other parts of the system including the controller, reward function, state vector and the tracking reference are visualized in Fig. 3. The programs can be used for further studies to be fully available as open-source programs for educational and research purposes.

2 Results and discussion

To get insight into the problem, the system with and without controller is analyzed using time-domain simulation, modal analysis, participation factor, and frequency response analysis. The mathematical equations and theoretical background of these subjects can be found in [27]–[29] and therefore, are not discussed here in this paper. Time-domain simulation is

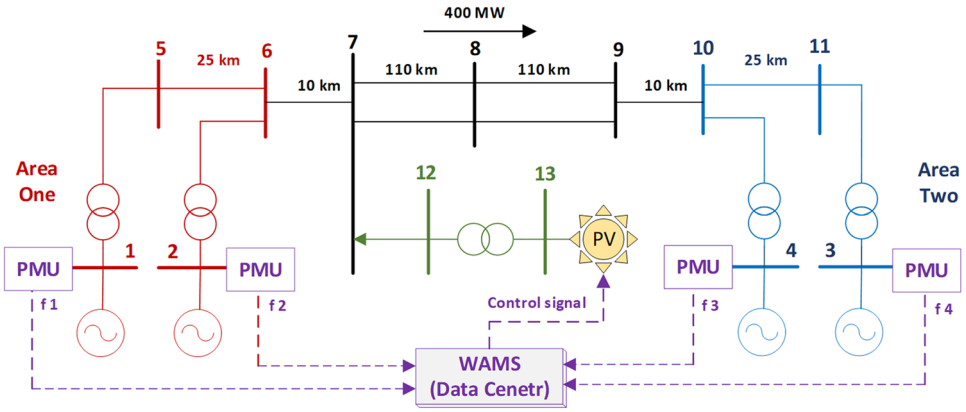

Fig. 9 A two-area test system

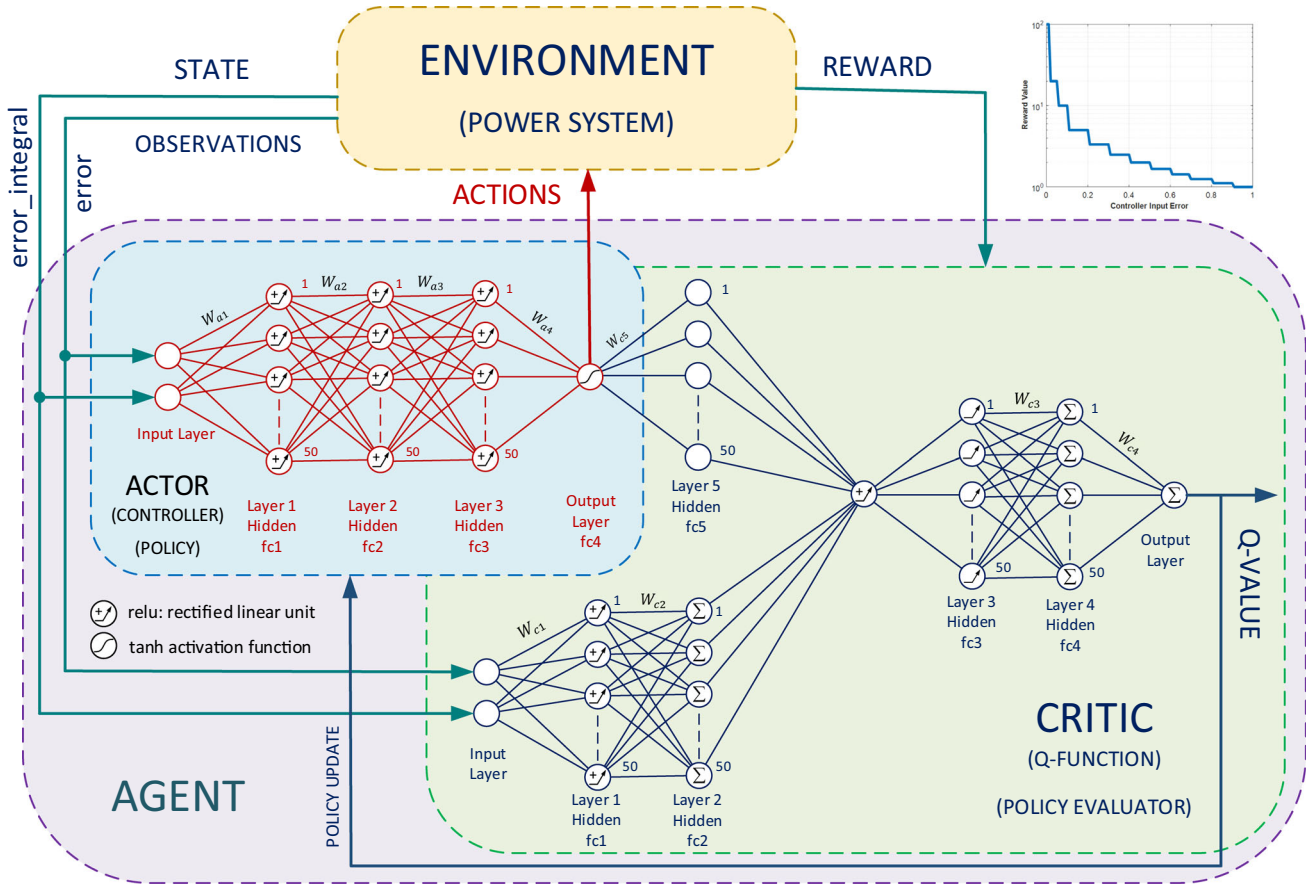

Fig. 10 The complete diagram of the system including the environment and agent

used to analyze the dynamic behavior of the system represented by a set of differential algebraic equations at each time step. In addition, the system is linearized around an operating point to compute the damping ratios, frequencies, eigenvalues, right and left eigenvectors of all modes, and the system's A, B, C, and D matrices.

The results obtained from modal analysis reveal the presence of undamped inter-area oscillations in the system. Figure 11 shows the simulation plots for the machine frequencies, where generators 1–2 in Area 1 swing as one oscillatory group against generators 3–4 in Area 2 evidenced from the $\sim 180^{o}$ degree out-of-phase (opposite direction) between the two areas. The oscillations are also confirmed from the modal analysis study. Table 1 lists the modes with the lowest damping ratios and their corresponding frequencies and eigenvalues. As it can be noted, there is a mode with a negative damping ratio (an indicator of unstable system) and this mode has a frequency of 0.60935 Hz which is in the range of inter-area frequencies lying between 0.1 and 0.8 [27]–[29]. This mode is the main cause of instability and inter-area oscillations in the system owing to involving all rotating parts of the system across the tie-line.

The mode-shape graphs for the modes listed in Table 1 are plotted and shown in Fig. 12. From these figures, we observe

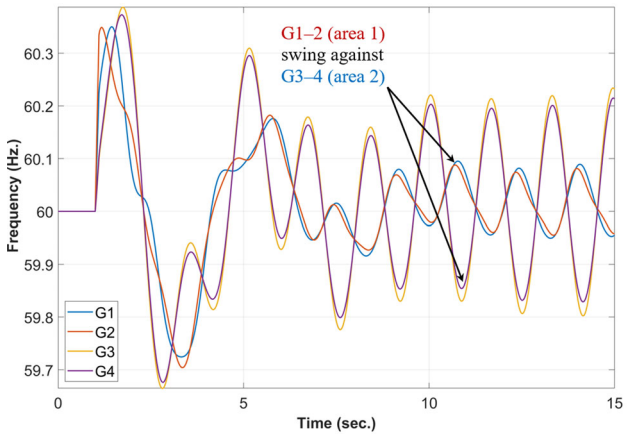

Fig. 11 Time-domain simulation of the system without controller

two types of oscillations in the system: inter-area oscillations shown in Fig. 12a and inter-plant local oscillations displayed in Fig. 12 b–c. The first mode-shape shows how the generators in one area swing against the other two generators in the other area (see the directions of the arrows). The other two mode-shapes reveal another type of oscillations known among generators in one area which has less influence on the

Table 1 Modal analysis results–without controller

Damping ratio (%)	Frequency (Hz)	Mode
− 0.0091393	0.60935	0.034993 + 3.8287i
0.066865	1.1327	− 0.47695 + 7.1171i
0.085341	1.0879	− 0.58546 + 6.8352i

overall insatiability and can be damped out by local PSSs. Mode-shapes reveal another observation– which is also confirmed by the time-domain simulation in Fig. 11: generators in Area 2 have more impact on the inter-area mode compared to generators in Area 1. This can be concluded from the lengths of the arrows in Fig. 1 and the heights of oscillations in Fig. 11.

The pole-zero map of the system is displayed in Fig. 13 detecting an unstable mode with positive real part of the eigenvalue (0.034993 + 3.8287i). Figure 14a–c show the results obtained from the frequency response analysis including Bode, Nyquist, and Nichols plots. All these plots show a sudden sharp change in the gain and phase diagrams which is an indicator of instability in the system. Stable systems show smooth plots over a wide-range of frequencies. Nyquist plot shows this instability state in another form through observing the critical point (-1, 0) which is encircled by the diagram. Furthermore, Nichols plot confirms this instability condition through looking at the critical point which passes over the point (180^o, 0).

Another useful analysis-tool is the participation factor map obtained from the right and left eigenvectors of the linearized system shown in Fig. 15. This map visualizes the role of state variables in the system affecting the modes, and we are more interested in the oscillatory modes. The most influential state-variables that contribute to the oscillations are the electromechanical state variables–that is, rotor angle and machine speed. As expected from other analyses, generators

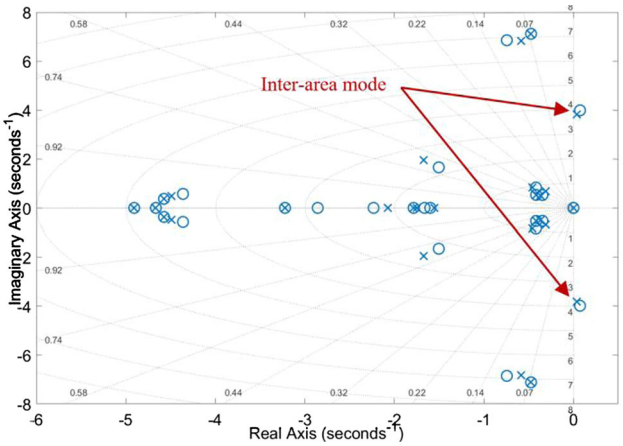

Fig. 13 Pole-zero map of the system without controller

3–4 in Area 2 participate more into the inter-area modes (see the yellow and green colors in the southwest side of Fig. 15). Note that each oscillatory mode comes with a complex conjugate number representing the eigenvalues. For this reason, each two consecutive rows in Fig. 15 are identical except for non-oscillatory modes that have 100% damping ratios. The color bar on the right of the figure shows how these participation factors change gradually over the normalized range [0 1]. The map is developed in a way that by moving over any position on the map, all necessary information is displayed including the eigenvalue, damping ratio, frequency, and the relationship between each state on $x-$ axis and each mode on $y-$ axis.

The above discussion was for the system without controller. Now, we want to discuss training the controller and the results obtained from the design of controller. For this training, a three-phase fault-disturbance is applied to the system for the period t = 1–2 s to generate some extreme oscillations in the system. Since the overshoot is higher at the beginning of the fault, only a 10 s simulation-window is selected

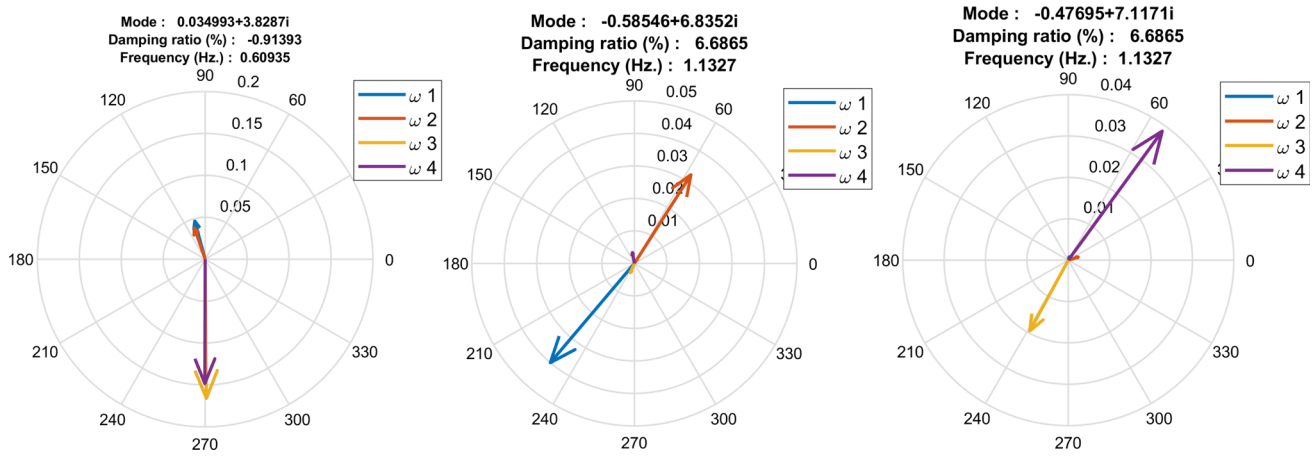

Fig. 12 Mode-shape plots of the inter-area mode (left), local mode 1 (middle), and local mode 2 (right)

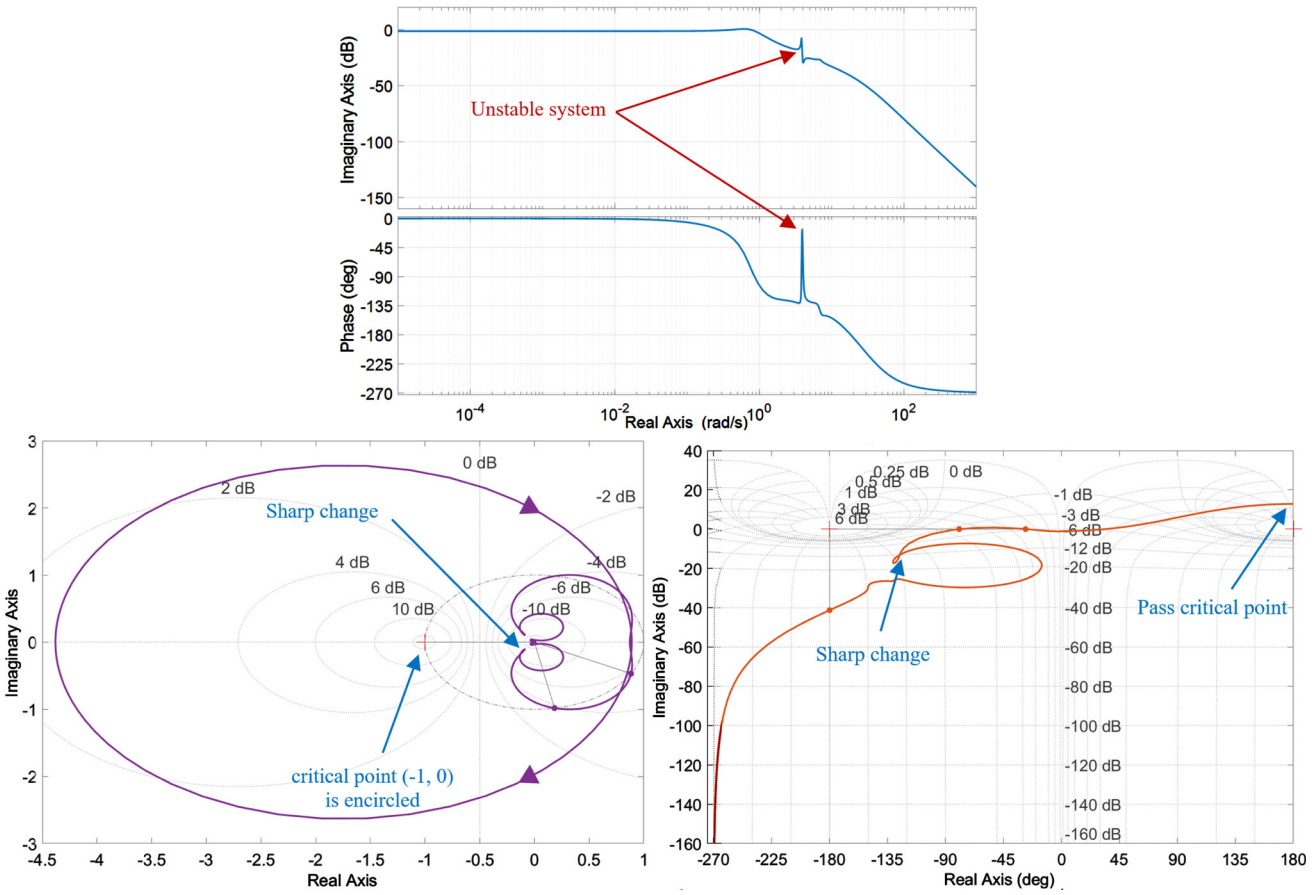

Fig. 14 Frequency response analysis: Bode plot (top), Nyquist plot (bottom left), and Nichols plot (bottom right)

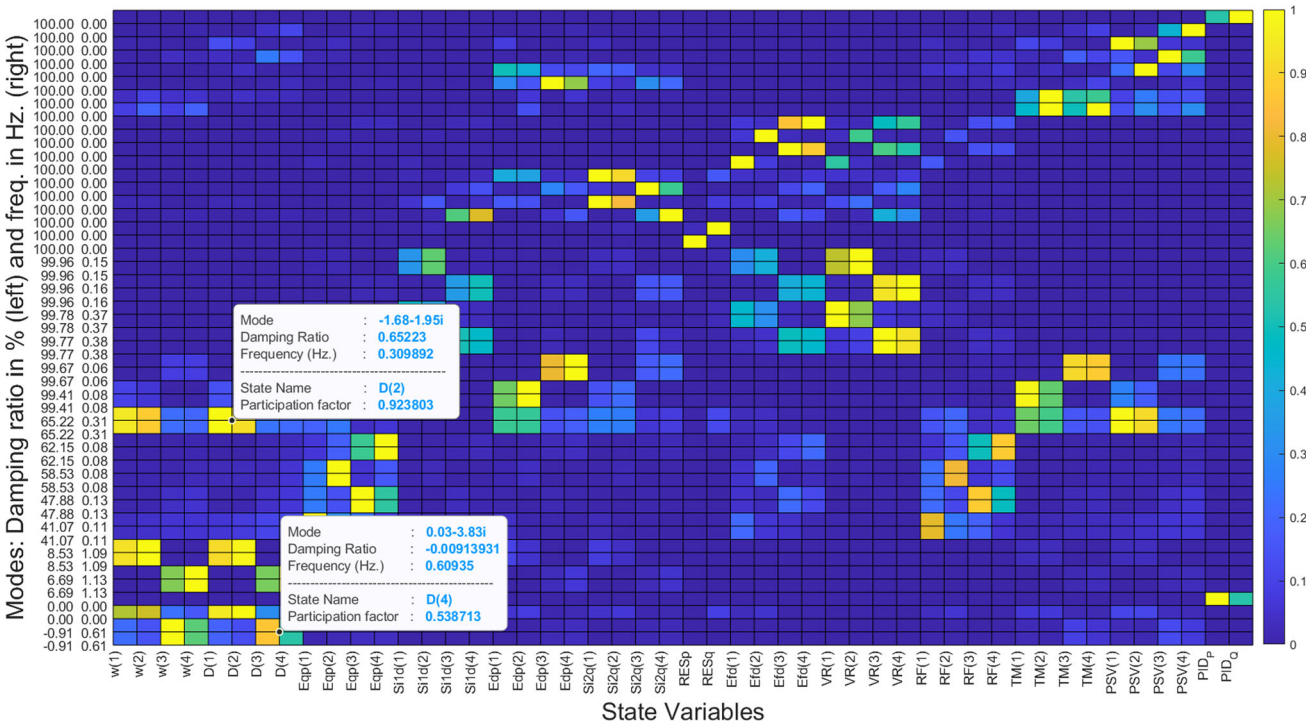

Fig. 15 Participation-factor map of the linearized system

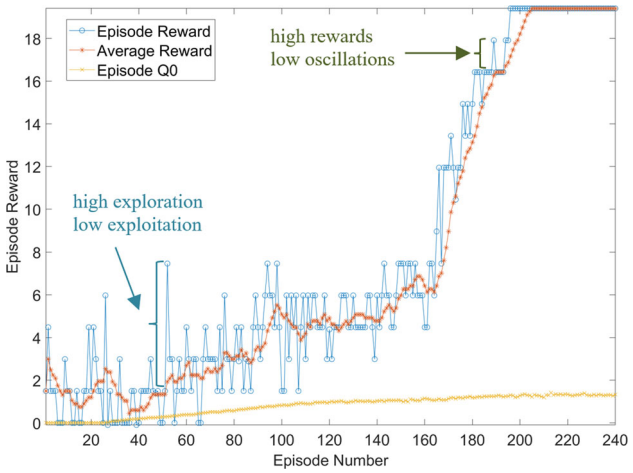

Fig. 16 Training episode rewards as a function of episode number

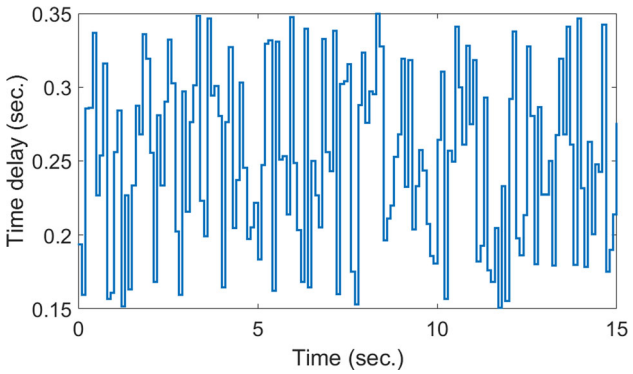

Fig. 17 Communication-delay variation applied to the controller input signal

for training the agent. Figure 16 shows the training results for this design with the episode reward labeled on y–axis against the episode number on x–axis. In this figure, the episode reward is plotted in blue, the average reward in red, and the episode Q-value in yellow. The training starts with a low episode-reward around a value of 3, then by exploring for more possible rewards, the average reward drives higher especially around the episode number 160 when a noticeable increase in rewards emerges. The training stops when the reward reaches a value where no improvement is obtained and this occurs around a value of 20.

Note that the training starts with a high exploration space to choose actions for the agent that explore more unknown parts of the environment. As the training continues and episode reward increases, the agent tries to exploit the environment for the rewards that are already known to the agent. This can be observed from Fig. 16 where the variation space for the episode reward (the blue vertical lines) is high at the beginning of training but it decreases as time goes on. Connecting to the problem in this study, a high reward value provides a high damping of oscillation and vice versa.

Table 2 Modal analysis results–with controller and delay

Delay	Damping ratio (%)	Frequency (Hz)	Mode
0.25 s	11.3	0.5730	−0.406 + 3.58i
0.15–0.35 s	9.63	0.6000	−0.36 + 3.75i

When the episode reward stops improving, it refers to reaching the maximum possible amount of damping for the controller. Comparing to some recent studies [1], the proposed reward function is effective to achieve its maximum learning from online data within a relatively short time. For instance, the later study required over 4000-episodes for the agent to increase its reward to the maximum for the same test system. While training time is not pointed out in [1], the proposed agent requires only ∼ 200 episodes with a training time of 1315.7 s to maximize the information gained from interaction with the environment.

After the training stopped, the system with the designed controller is simulated using the same analytical tools we used for the case without controller. To incorporate the communication latency, two realistic scenarios are considered: (1) constant time delay of 0.25 s (2) variable time delay in the range of 0.15–0.35 s as shown in Fig. 17. This delay domain covers practical values for the majority of communication links including fiber-optic cables, microwave links, power line carriers, and telephone lines. The results obtained from modal analysis show that the inter-area mode is well-damped and moved to the stable region with the characteristics listed in Table 2. The controller with constant latency shows slightly better damping (11.3%) than the variable delay (9.63%) which is expected since the variable delay goes up to 350 ms, whereas the lower limit is set to 150 ms and not zero. Figure 18a–d show some results obtained from the time-domain simulation, pole-zero map, and frequency response analysis. Figure 19a–b exhibit the simulation results when variable delay is used.

When the episode reward stops improving, it refers to reaching the maximum possible amount of damping for the controller. Comparing to some recent studies [1], the proposed reward function is effective to achieve its maximum learning from online data within a relatively short time. For instance, the later study required over 4000-episodes for the agent to increase its reward to the maximum for the same test system. While training time is not pointed out in [1], the proposed agent requires only ∼ 200 episodes with a training

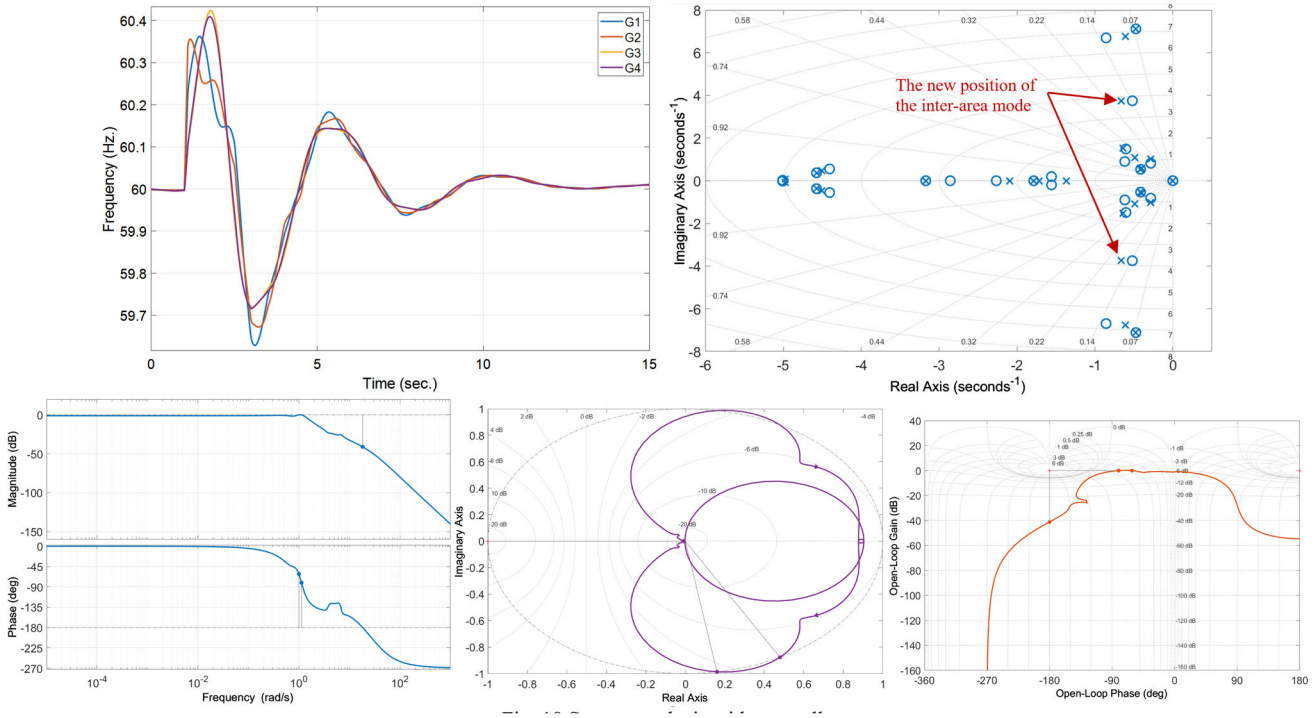

Fig. 18 System analysis with controller Top left: time-domain simulation, Top right: pole-zero map, bottom left: Bode plot, bottom middle: Nyquist plot, and bottom right: Nichols plot

time of 1315.7 s to maximize the information gained from interaction with the environment.

After the training stopped, the system with the designed controller is simulated using the same analytical tools we used for the case without controller. To incorporate the communication latency, two realistic scenarios are considered: (1) constant time delay of 0.25 s (2) variable time delay in the range of 0.15–0.35 s as shown in Fig. 17. This delay domain covers practical values for the majority of communication links including fiber-optic cables, microwave links, power line carriers, and telephone lines. The results obtained from modal analysis show that the inter-area mode is well-damped and moved to the stable region with the characteristics listed in Table 2. The controller with constant latency shows slightly better damping (11.3%) than the variable delay (9.63%) which is expected since the variable delay goes up to 350 ms whereas the lower limit is set to 150 ms and not zero. Figure 18a–d show some results obtained from the time-domain simulation, pole-zero map, and frequency response analysis. Figure 19a–b exhibit the simulation results when variable delay is used.

It is worth mentioning that the frequency of inter-area mode stays almost around the same value for the cases: (1) without controller, 0.60935 Hz (2) with controller–constant delay, 0.573 Hz, and (3) with controller–and variable delay, 0.6 Hz. In the design of controller, the objective is to damp the oscillations by shifting the real parts of the eigenvalues to the far-left side apart from the imaginary axis of the complex plane. In other words, the mode frequencies remained unchanged.

There are several applications of the proposed control strategy. Improving power system stability through damping inter-area oscillations is one of these applications demonstrated by this study. Inter-area oscillations problem causes instability if not damped well. The control system used to solve this problem requires global information because the oscillation occurs due to interactions and swinging among generators in one area against generators in another area connected through weak tie-lines. In addition to damping inter-area oscillations, the proposed agent can be used to reduce local interaction among generators in one area. This can be achieved by using auxiliary control component in a solar plant without even the need of a power system stabilizer.

3 Conclusion

This paper proposed a reinforcement-learning-based controller for a solar plant connected to a weak tie-line in a two-area system to damp inter-area oscillations. Deep deterministic policy gradient (DDPG) technique was used as the algorithm for training the agent. The reward function used as an input to the controller was a discrete reciprocal function of the error signal. Deep neural networks are used for the actor

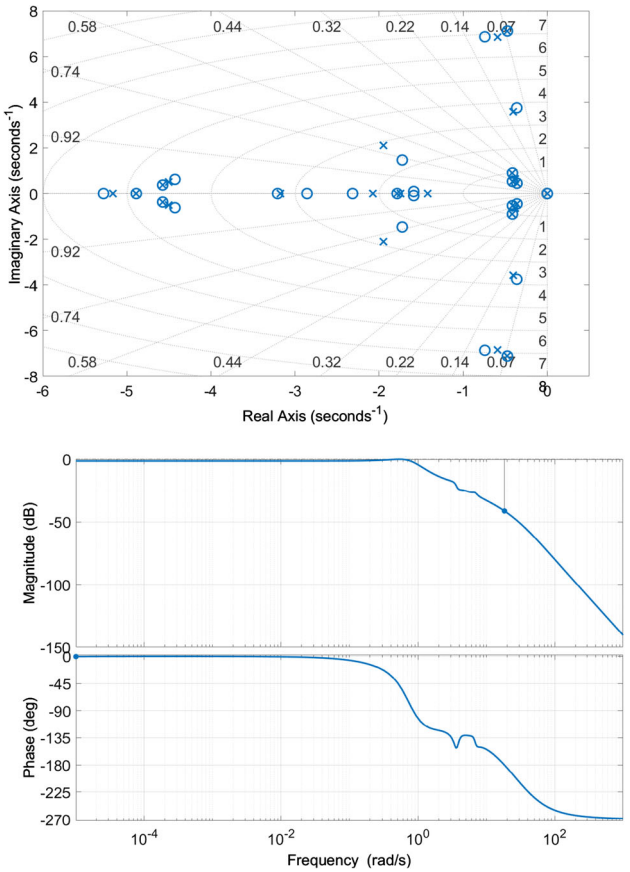

Fig. 19 Simulation results for the system with controller and variable time delay. Top: pole-zero map, bottom: bode plot

and critic which were exhibited in detail in this paper. The controller input is a remote signal obtained from the wide-area measurement system through PMUs installed at the generator sides. The system with and without the controller was comprehensively analyzed using several engineering and control tools including time-domain simulation, frequency response analysis, pole-zero map, and participation factor map. In the design of the agent, a range of practical communication time delays was included to show its robustness against oscillations generated by the latency. For this simulation, several programs were developed using MATLAB and Simulink which can be employed for further studies. The proposed controller showed its effectiveness in damping inter-area oscillations which was the main problem to be solved in this paper. The training process was relatively fast and successful in accelerating the convergence. It should be noted that the proposed control approach is a model-free reinforcement-learning technique which requires numerous amounts of data and computations for training the controller. All possible scenarios should be carefully considered when the controller is designed including system dynamics, noise, and disturbances. If a policy is found not quite right, then it will be computationally expensive and time consuming to redesign, train, and test the agent.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s00202-022-01615-3.

Funding Not applicable.

Declarations

Conflict of interest There are no competing interests in this manuscript.

References

1. Hashmy Y, Yu Z, Shi D, Weng Y (2020) Wide-area measurement system-based low frequency oscillation damping control through reinforcement learning. IEEE Trans Smart Grid 11(6):5072–5083. https://doi.org/10.1109/TSG.2020.3008364
2. Abdulrahman I, Belkacemi R, Radman G (2021) Power oscillations damping using wide-area-based solar plant considering adaptive time-delay compensation. Springer J Energy Syst 12:459–489
3. Cao D et al (2020) Reinforcement learning and its applications in modern power and energy systems: a review. J Modern Power Syst Clean Energy 8(6):1029–1042. https://doi.org/10.35833/MPCE.2020.000552
4. Sewak M (2019) Deep reinforcement learning frontiers of artificial intelligence. Springer Nature Pte Ltd, Singapore
5. Dong H et al (2020) Deep reinforcement learning fundamentals research and applications. Springer Nature Pte Ltd, Springer
6. Zhang G, Hu W, Zhao J, Cao D, Chen Z, Blaabjerg F (2021) A novel deep reinforcement learning enabled multi-band PSS for multi-mode oscillation control. IEEE Trans Power Syst 36(4):3794–3797. https://doi.org/10.1109/TPWRS.2021.3067208
7. Gupta P, Pal A, Vittal V (2022) Coordinated wide-area damping control using deep neural networks and reinforcement learning. IEEE Trans Power Syst. https://doi.org/10.1109/TPWRS.2021.3091940
8. Lee H et al (2019) Artificial neural network control of battery energy storage system to damp-out inter-area oscillations in power systems. Energies 12:3372. https://doi.org/10.3390/en12173372
9. Younesi A (2017) Application of reinforcement learning for generating optimal control signal to the IPFC for damping of low-frequency oscillations. Wiley Trans Electr Energ Syst. https://doi.org/10.1002/etep.2488
10. Liu F et al (2016) Robust wide-area damping controller design for inter-area oscillations with signals' delay. Wiley Trans Electr Energ Syst 11:206–215
11. Chen C, Cui M, Li F, Yin S, Wang X (2021) Model-free emergency frequency control based on reinforcement learning. IEEE Trans Industr Inf 17(4):2336–2346
12. Zhang G et al (2020) Deep reinforcement learning-based approach for proportional resonance power system stabilizer to prevent ultra-low-frequency oscillations. IEEE Trans Smart Grid 11(6):5260–5272. https://doi.org/10.1109/TSG.2020.2997790
13. Yan Z et al (2020) Deep reinforcement learning-based optimal data-driven control of battery energy storage for power system frequency support. In IET Gen Trans Dis. https://doi.org/10.1049/iet-gtd.2020.0884
14. Abouheaf M et al (2019) Load frequency regulation for multi-area power system using integral reinforcement learning. IET Gen Transm Distrib. https://doi.org/10.1049/iet-gtd.2019.0218

15. Zheng Y et al (2021) Power system load frequency active disturbance rejection control via reinforcement learning-based memetic particle swarm optimization. IEEE Access 9:116194–116206. https://doi.org/10.1109/ACCESS.2021.3099904
16. Yan Z, Xu Y (2019) Data-driven load frequency control for stochastic power systems: a deep reinforcement learning method with continuous action search. IEEE Trans Power Syst 34(2):1653–1656. https://doi.org/10.1109/TPWRS.2018.2881359
17. Chen X et al (2021) Reinforcement learning for decision-making and control in power systems: Tutorial, Review, and Vision, in arXiv: 2102.01168
18. Yang T etal (2020) Reinforcement learning in sustainable energy and electric systems: a survey, in Elsevier Annual Reviews in Control
19. Zhang Z, Zhang D, Qiu RC (2020) Deep reinforcement learning for power system applications: an overview. CSEE J Power and Energy Syst 6(1):213–225. https://doi.org/10.17775/CSEEJPES.2019.00920
20. Glavic M (2019) (Deep) Reinforcement learning for electric power system control and related problems: a short review and perspectives, in Elsevier Annual Reviews in Control
21. Zhang D, Han X, Deng C (2018) Review on the research and practice of deep learning and reinforcement learning in smart grids. CSEE J Power and Energy Syst 4(3):362–370. https://doi.org/10.17775/CSEEJPES.2018.00520
22. Siraskar R (2021) Reinforcement learning for control of valves. Elsevier Mach Learn Appl. https://doi.org/10.1016/j.mlwa.2021.100030
23. Abdulrahman I, Radman G (2018) Wide-area-based adaptive neuro-fuzzy SVC controller for damping interarea oscillations. Can J Electric Comput Eng 41(3):133–144
24. WECC Renewable Energy Modeling Task Force (2014) WECC solar plant dynamic modeling guidelines
25. Sandia National Laboratories (2013) Generic solar photovoltaic system dynamic simulation model specification", California
26. Islam MR, Rahman F, Xu W (2016) Green energy and technology: advances in solar photovoltaic power plants. Springer, Berlin
27. Pauer P, Pai MA, Chow JH (2017) Power system dynamics and stability, 2nd edn. Wiley, Amsterdam
28. Kundur P et al (2004) Definition and classification of power system stability IEEE/CIGRE joint task force on stability terms and definitions. IEEE Trans Power Syst 19(3):1387–1401
29. Pal B, Chaudhuri B (2005) Robust control in power system. Springer, New York