İSTANBUL UNIVERSITY
CERRAHPAŞA

ORIGINAL ARTICLE
Electrica 2023; 23(2): 202-211

Design of Graphical User Interface for Artificial Intelligence-Based Energy Management System for Microgrids

Necati Aksoy, **İstemihan Genç**

Department of Electrical Engineering, İstanbul Technical University, İstanbul, Turkey

Cite this article as: N. Aksoy and İ. Genç, "Design of graphical user interface for artificial intelligence-based energy management system for microgrids," *Electrica*, 23(2), 202-211, 2023.

ABSTRACT

Microgrids emerge as a structure that gains more importance day by day with the reduction of energy loss rate, the efficient use of renewable energy sources, the possibility of autonomous operation with energy storage systems, and the profitability it offers. Furthermore, this structure, which helps to reduce the carbon footprint, will become undeniably critical to use in the near future with the nanogrid and smart grid. An innovative dynamic energy management system will make these advantages offered by the microgrid more accessible while facilitating the integration and effective contribution of electric vehicles. On the other hand, thanks to promising and useful developments and algorithms in machine learning and deep learning, artificial intelligence (AI)-based control methods and applications are constantly increasing. Accordingly, the concept of reinforcement learning (RL) offers an unconventional perspective on the control of systems. This study, which is the last step of creating an AI-based energy management system, presents a graphical interface design in light of all these requirements and developments. In this study, the deep RL agent used in determining management actions, together with the prediction models created to make the necessary predictions, are gathered under a single roof.

***Index Terms*—**Reinforcement Learning, GUI design, microgrid, deep learning, energy management, artificial intelligence

I. INTRODUCTION

Whereas studies on energy are increasing day by day, solutions are sought with smart grids, microgrids and nanogrids for the use of energy with as little loss as possible, the correct management of distributed generation resources, and the reduction of carbon emissions in energy production. In addition, the difficulties encountered in the installation, correct sizing, the correct selection of the energy storage unit, and its integration into the system, as well as a strong energy management, appear as scientific issues that are studied in the microgrid specifically [1]. The scientific infrastructure built on the energy management system must contain the innovations of today's world and be open to future improvements. In this direction, the fact that the computer and calculation tools are advanced in terms of technology and the algorithms used are efficient and up-to-date leads us to the idea that the energy management system should be based on artificial intelligence (AI) [2]. If the concept of AI is to be detailed, the main title of AI covers machine learning algorithms, which is the subtitle, and machine learning includes deep learning algorithms and structures [3]. One of the most important elements for efficient energy management of microgrids is to use and control the energy storage unit optimally [4]. Although this control problem can be solved at one level with an expert system-based control structure, if it is aimed to establish an AI-based energy management system, solving this control problem with an AI-based control structure will make the system more innovative and efficient [2]. The idea of using reinforcement learning (RL) algorithms is worth working on in this context [5, 6]. The efficient use and control of the energy storage unit will reduce the energy cost of the microgrid and provide improvements in the installation payback time [7]. In addition, while operating in island mode, it will make the system more stable and will reduce the need for energy to be taken from the grid when working in connection with the grid.

This autonomous capability of microgrids requires a strong energy management system infrastructure. Today, microgrids are sized using classical installation methods and controlled with expert system-based management systems [8]. This type of management both limits the

Corresponding author:
Necati Aksoy

Email: aksoyn18@itu.edu.tr

Received: June 1, 2022

Revised: July 8, 2022

Accepted: July 19, 2022

Publication Date: November 12, 2022

DOI: 10.5152/electrica.2022.22102

efficiency that can be obtained at the time of operation of the microgrid and has difficulty in following new trends in energy storage technologies [9, 10]. Detailed analysis of wind data alone will also increase the efficiency of renewable energy sources both in the microgrid and as a separate power plant [11]. The idea that current energy storage technologies are used effectively in the system and that today's AI technology forms the basis of the energy management system of microgrids constitutes the importance of the study. This study presents the graphical interface design (GUI) design part, which is the last section of the larger AI-based energy management system design work. Designing an innovative dynamic energy management system with novel and eminent AI methods and trends in the relatively new field of smart microgrid systems makes the study significant and valuable. The grid-connected microgrid, which is aimed to be simulated in the GUI, uses both historical and current data in the energy management, as opposed to using only optimization methods. It is planned to provide continuous and uninterrupted energy to the user while reducing the cost of energy by processing this data with AI methods. In addition, it is targeted to contribute to the reduction of carbon emissions by aiming to minimize the energy to be demanded from the grid connection and to maximize the use of renewable energy sources and energy storage systems (ESS) in the microgrid.

II. ARTIFICIAL INTELLIGENCE IN THE INTERFACE

The infrastructure of the designed GUI consists of current trends and methods of AI. While the prediction and classification models created using various machine learning algorithms provide very powerful results for energy management, the deep RL agent using these prediction models offers the user a control action that should be run. This section is reserved to briefly explaining these AI methods used in the designed GUI.

A. Reinforcement Learning

The concept of RL, which is one of the three basic building blocks of machine learning, offers us a different framework for the control of systems from the classical control methods. Thanks to this concept, suitable control solutions are sought for problems that could have discrete or continuous states. In today's world, both RL algorithms and the problems and solutions in which these algorithms are applied are in a continuous development. This concept has come to the fore as a structure that has been studied academically intensively in the last couple of years, and up-to-date academic results are obtained day by day.

As can be seen in Fig. 1, the RL framework consists of an agent and environment as well as action A_t, current state of state space S_t, and rewards R_t [12].

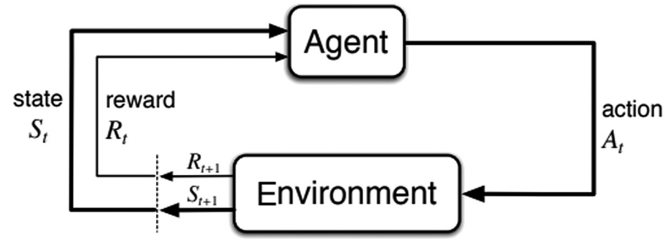

Fig. 1. Markov decision process framework for RL.

The basic concepts used in all RL algorithms are based on Markov decision process (MDP) framework. Markov decision process is a discrete-time stochastic control process. It provides a mathematical framework for modeling decision-making in situations where outcomes are partly random and partly under the control of a decision-maker. Markov decision processes are useful for studying optimization problems solved via dynamic programming and Monte Carlo. Accordingly, MDPs, more specifically finite MDPs, form the basis of RL. Markov decision processes are based on a framework of determining an action according to the outcome they receive (as a reward, can be positive or negative) by interacting with an environmental model. In RL, created agent choose an action a considering on state-value *V(s)* or action-state value *Q(s,a)*. Therefore, as the agent interacts with the environment model, it updates these state values or action-state values using a temporal difference (TD). For problems with a limited number of states, RL algorithms create action-value tables (Q-Tables), while in cases where the number of states is too large or unlimited, these algorithms use approximate action-state value results. They do these approaches using deep learning networks trained by a certain number of episodes. In RL, the goal is to develop a proper policy function *π(s)* that will find an appropriate action for every possible situation. Policy function uses action-state values to select an action. Hence, some RL algorithms use deep Q networks (DQN) to approximate *Q(s,a)* value. Deep Q Learning, Deep SARSA, Rainbow DQN, Double DQN (DDQN), and Normalized Advantage Function (NAF) algorithms are based on approximating *Q(s,a)*. On the other hand, some RL algorithms directly focus on approximating policy function, and these are called as policy gradient algorithms such as Actor-Critic (A2C, A3C), REINFORCE, PPO, and TRPO. Furthermore, some algorithms like Deep Determined Policy Gradient, Soft Actor Critic, and Twin Delayed Deep Deterministic Policy Gradients (TD3) use policy gradient and DQN together.

A deep RL agent is created and trained for action recommendation to the users in designed GUI. Created agent uses DDQN algorithm, which is trained using price, load, and power generation forecasts that come from prediction models.

B. Deep Learning

Deep learning is a type of machine learning that uses deep artificial neural networks to enable digital systems to learn and make decisions based on unstructured, unlabeled data. Generally, machine learning trains AI systems to learn by examining experiences with data, recognizing patterns, making recommendations, and adapting. Especially when it comes to deep learning, digital systems learn from examples rather than just responding to rule sets and then use that information to react, behave, and perform like humans [3]. What differentiates deep learning from artificial neural networks (ANNs) is the high number of hidden layers in the network structures established in deep learning and the diversity of neurons used in deep learning. In this study, deep learning structures are preferred to create forecasting models from dynamic price and load demand data acquired as time series due to the fact that deep learning is very effective in drawing conclusions from such time-dependent data. In particular, neuron structures with memory offered by deep learning learn more easily how time series data will change with the help of their memories. Recurrent neural networks (RNNs) and long short-term memory (LSTM), which are memory cell structures, stand out in this regard. Since LSTM is an advanced RNN structure, let us first look at RNN. Unlike ANN, in RNN, besides the input data at time t, the hidden layer results from time $t-1$ are also the input of the hidden layer

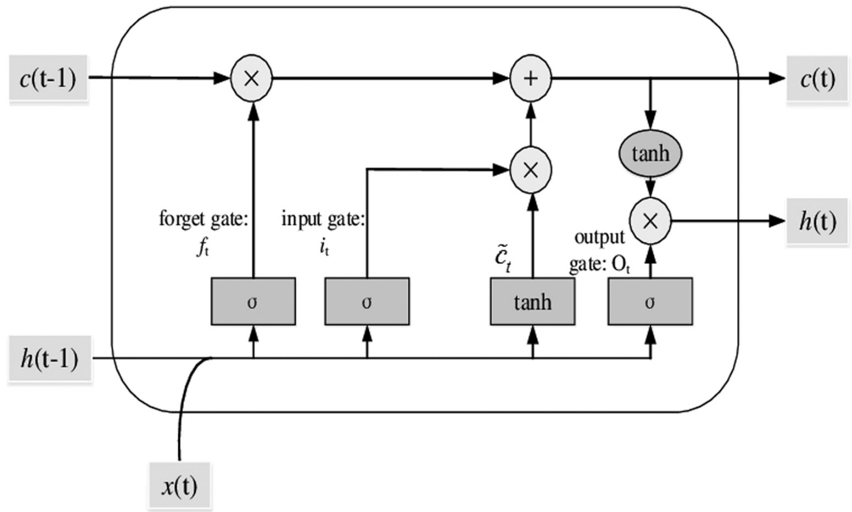

Fig. 2. LSTM cell structure.

at time t. The decision made for the input at time $t-1$ also affects the decision to be made at time t. In other words, in these networks, inputs produce output by combining current and previous information. Recurrent structures are distinguished from feedforward structures because they use their output as input in the next process. It can be said that recurrent networks have a memory. The reason for adding memory to a network is because a set of inputs in a certain order has a meaning for the output. Feedforward networks are not sufficient for such datasets. This is where RNNs come into play. Recurrent networks are used to understand the structure of incoming data in a certain order, such as text, speech, and various sensory or statistical data depending on time. As for LSTM, Sepp Hochreiter and Juergen Schmidhuber developed LSTM in 1997 [13] to solve the vanishing gradient problem. Long short-term memory, which was later edited and popularized with the contribution of many people, now has a wide usage area. Long short-term memory preserves the error value from different times and layers in backpropagation. By providing a more constant error value, it ensures that the learning steps of recurrent networks can continue. It does this by opening a new channel between cause and effect. The difference of the repeating module in the LSTM structure is that instead of a single neural network layer, there are four layers that are connected in a special way. These layers are also called gates. It is a structure that receives information outside of the normal flow. This information can be stored, written to the cell, and read.

As can be seen on Fig. 2, the LSTM cell structure is seen. In this figure, $c(t-1)$ represents the long-term memory line from the past, and $h(t-1)$ is the output value of the previous cell. Here, the current input is shown as $x(t)$, and it is decided whether this input will be forgotten or stored in long-term memory with a forget gate and a sigmoid function. When we look at the cell outputs, the $c(t)$ value sends the current long-term memory output and $h(t)$ sends the current output value to the next cell. In this context, it should be determined how many LSTM cells will be used, depending on the complexity and type of the problem. Because of all these useful features, the network structures established with LSTM are used to create prediction models.

C. Ensemble Learning

It might be concluded that a group of people who are different from each other in social life are more likely to make better decisions than individuals. What is similar in machine learning is getting better and healthier results with different models compared to single models.

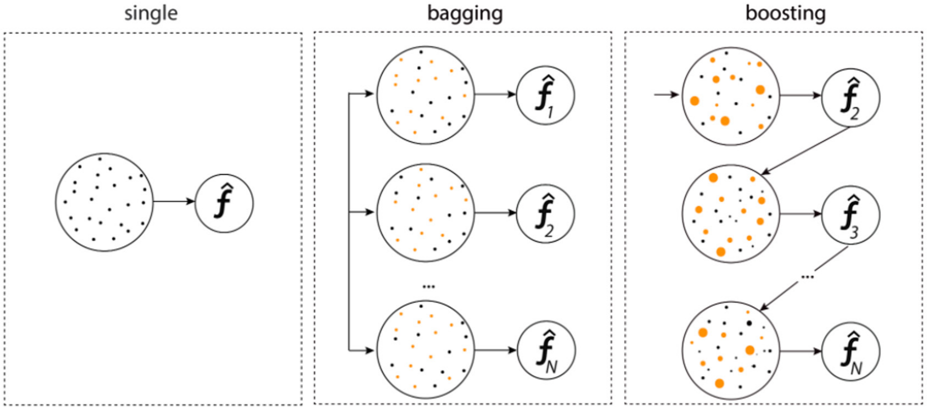

Fig. 3. Ensemble learning: bagging and boosting vs others.

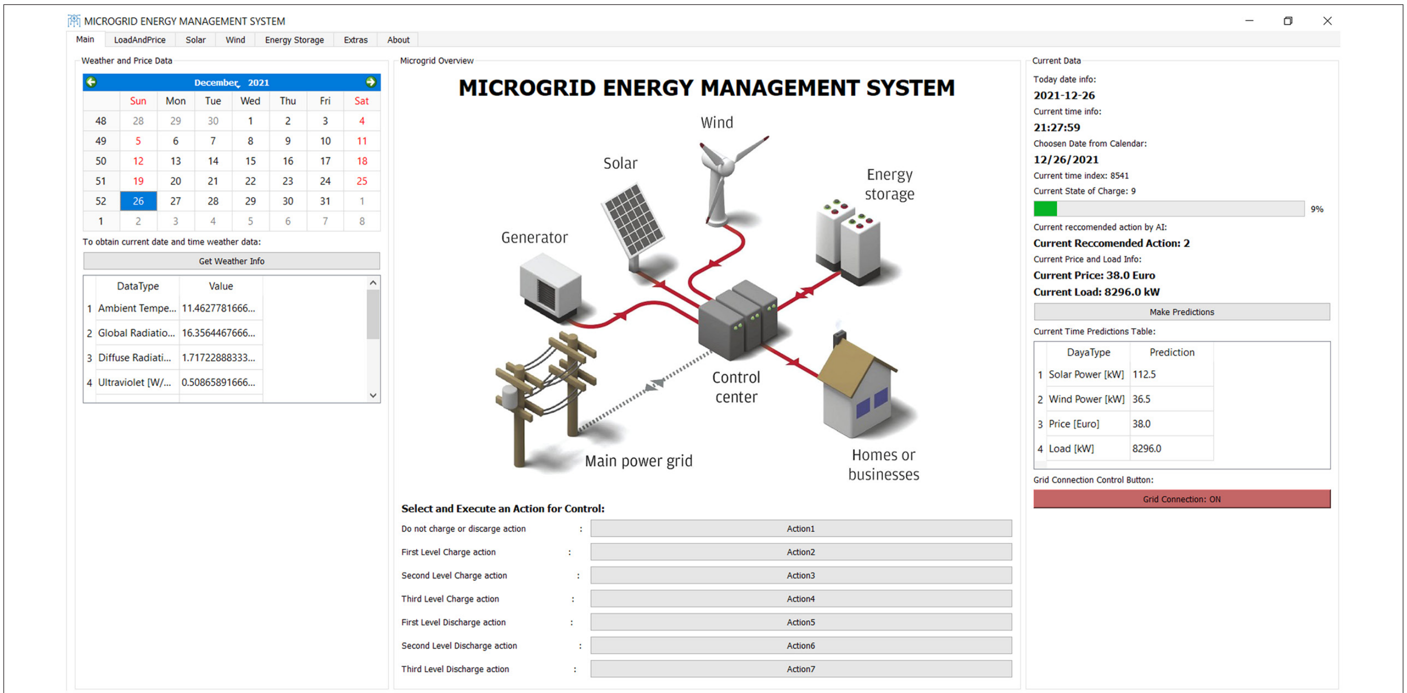

Fig. 4. Home page where the user can see current information about the microgrid and perform control actions.

This diversity in machine learning is accomplished through a technique called ensemble learning (EL). Ensemble learning can basically be summarized as the results of different models deciding on a result together.

Ensemble learning can be divided into three main topics: bagging, boosting, and stacked generalization (or stacking). In Fig. 3, single and both methods based on bagging and boosting of EL are illustrated. Ensemble learning methods are gaining popularity due to the fact that bagging or boosting algorithms have superiority to other machine learning regression or classification methods. Specially, boosting-based algorithms are improving, and novel algorithms come to the fore. To explain boosting algorithms briefly, boosting is to create a strong learner by bringing together many weak learners. The basic approach of many boosting methods is to train the estimators cumulatively. Decision trees are generally used as predictive models. Each tree is trained with a modified version of an original dataset, and a powerful classifier is eventually created. The most commonly used boosting models are: AdaBoost (1995, Y. Freund), Gradient Boosting (2001, Friedman), XGBoost (2016, Washington University), LightGBM (2017, Microsoft), and CatBoost (2018, Yandex).

Since EL-based algorithms give good results in creating prediction models, the designed energy management system uses these models together with other machine learning algorithms such as Support Vector Machine (SVM), linear regression, or logistic regression for solar power and wind power prediction.

III. CONSTITUENTS OF THE DESIGN

The GUI for energy management system is designed as seven pages together with the main (home) page. Each page is the section where the data of the systems in a microgrid or the data that the microgrid would need are processed and displayed. In this section, these pages are explained one by one consecutively.

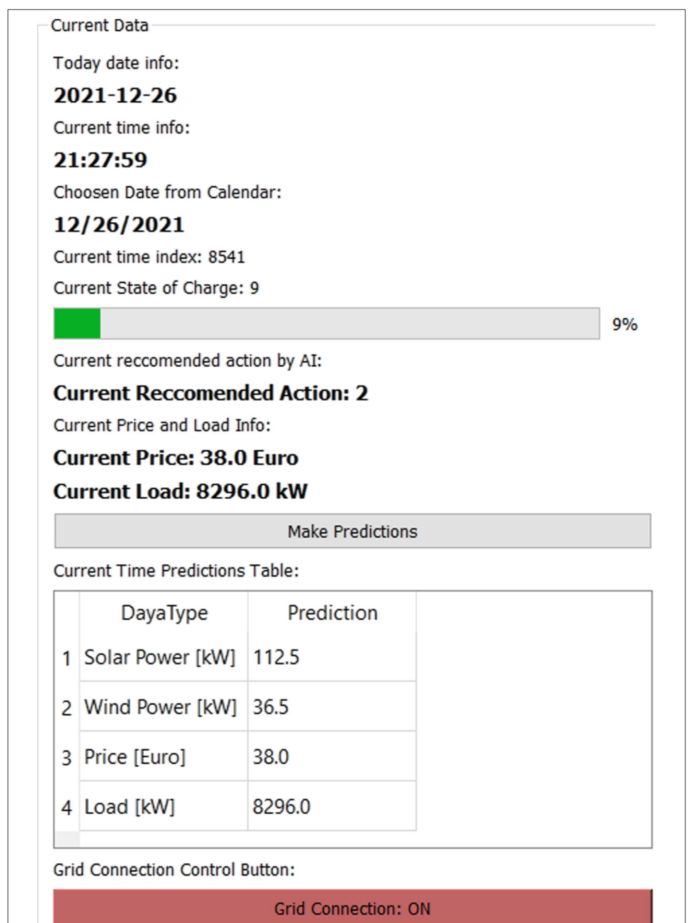

Fig. 5. Right part of the main page layout.

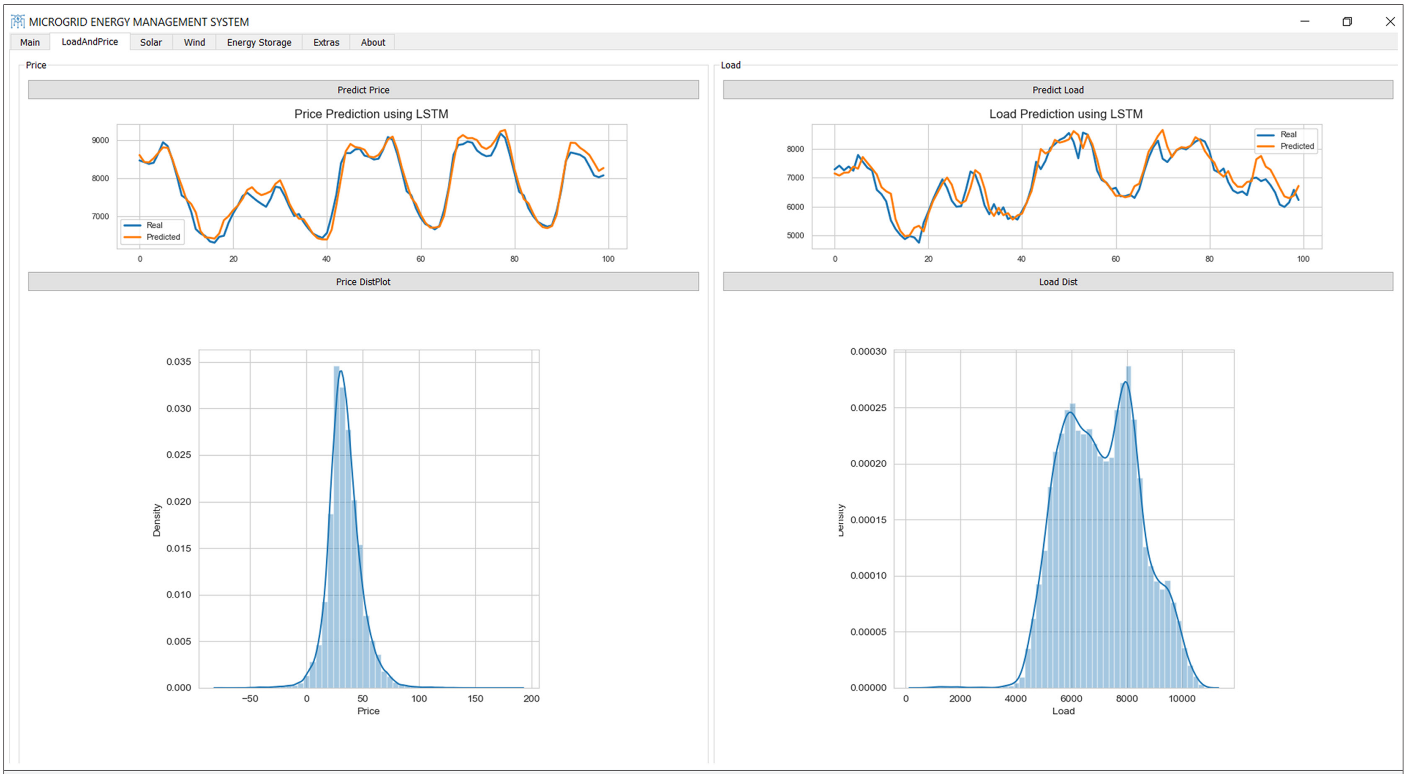

Fig. 6. Page where price and load data can be analyzed and prediction results can be analyzed.

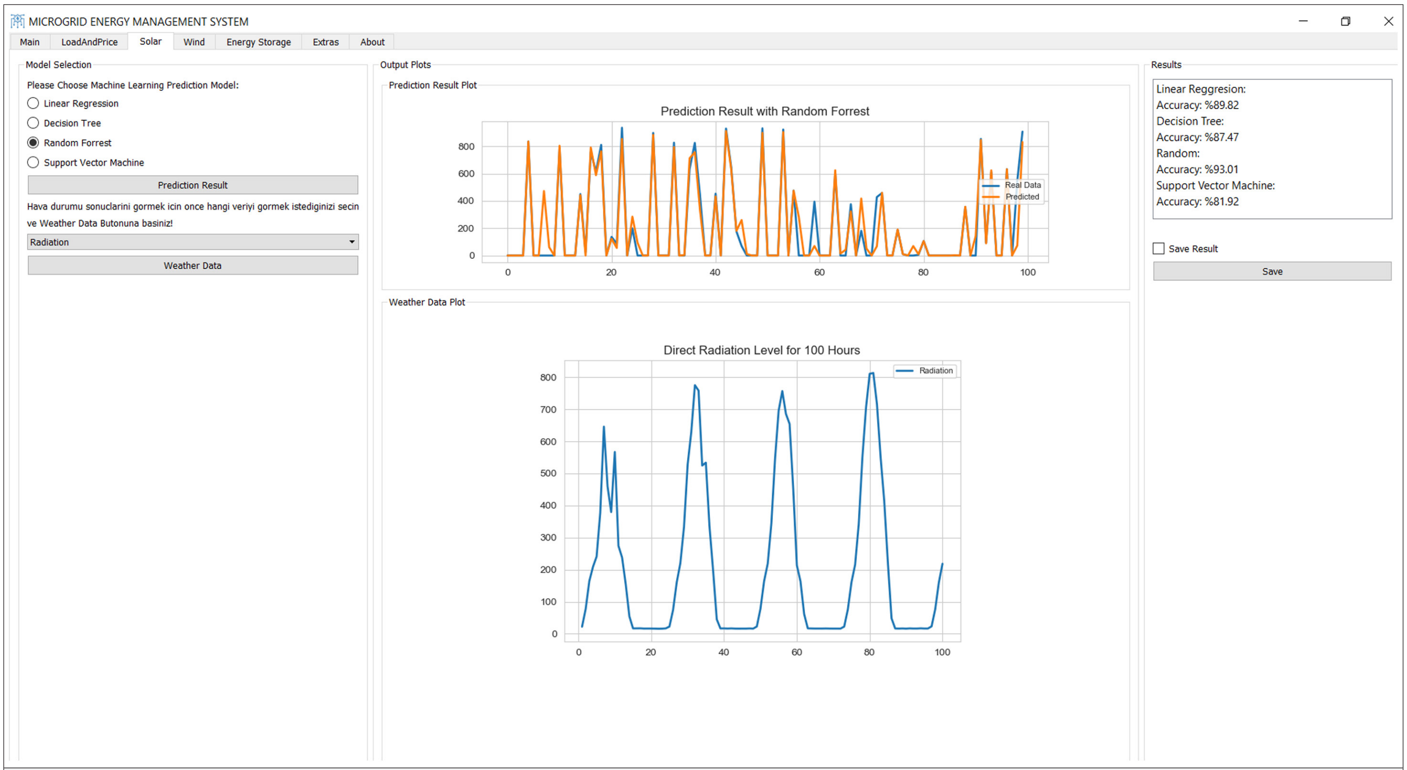

Fig. 7. Solar page: management and analysis page of the solar power plant in the microgrid.

A. Home Page

The home page in the GUI of the energy management system is designed as a place where the user can view the current data needed by the user, make predictions about that moment, and view the results, as well as select and apply a series of control actions. The layout plan is designed in three parts in Fig. 4. In the first of the three parts on the right, the calendar can be seen and the forecast values for the past or future days to be selected from this calendar can be obtained.

Under the calendar, weather data for that day and time can be pulled from the system and displayed. In the middle part of layout, control actions could be seen. Users can select and execute an action to control ESS. On the right part of home page, the user can see date and time information, current state of charge (SoC) value, and recommended action that comes from DDQN agent as well as current price and load demand values shown in Fig. 5.

Besides, using the “Make Predictions” button, solar and wind power generation forecasts and price data and load demand forecasts can be done by the user. Last, the “Grid Connection” button offers the user to turn on or off the grid connection.

B. Load and Price Page

Knowing pricing information is vital to the profitability of microgrids. Also, the amount of load to be demanded from the microgrid is very important for energy management. For these reasons, this page is reserved for the user to delve into pricing and load demand data. On this page, the user can view the price data of the last year together with the forecast results of the LSTM-based prediction model for that year. In this way, it can be analyzed whether the prediction model is working well and examined how it behaves in sharp points. This page consists of two parts. The left side is reserved for price data analysis, while the right side is allocated to analysis of load demand data. The page layout is presented in Fig. 6.

The “Predict Price” button is placed on the left top of the page which makes predictions and plot together with real price data for the same hundred days. The “Price DistPlot” button plots a distribution plot of the prices for a year that offers the users to examine price changes. Similarly, on the right part of the page, the user can plot load demand forecasts as well as real load demand data for same hundred days of the last year. How the load demand has changed in a

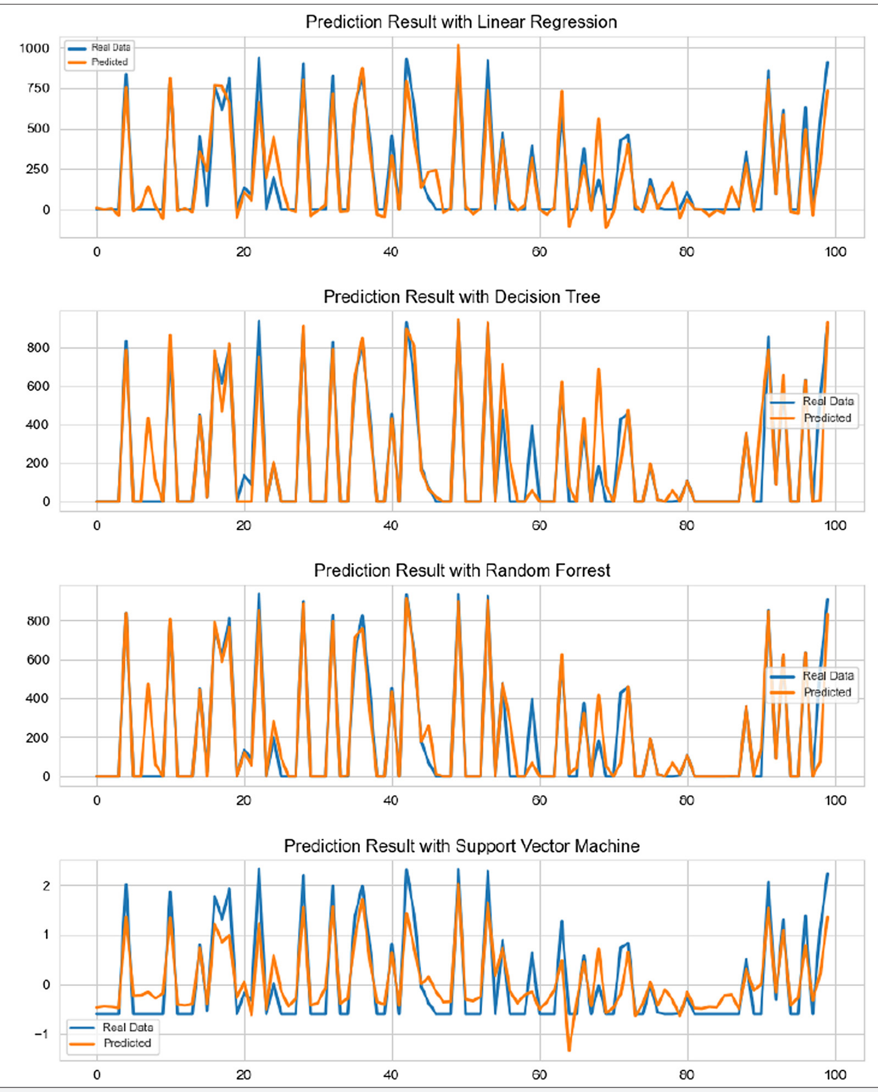

Fig. 8. Solar power prediction results with different models.

year is shown to the user via the "Load DistPlot" button. Accordingly, the user can take measures for the microgrid.

The important point here is that when the forecast buttons are pressed, the GUI does not bring up and display ready-made graphics but instead makes the forecasts using forecast models again each time it is pressed and shows it to the user. Thus, the user can evaluate the current performance of the prediction models.

C. Solar Page

The effective use of renewable energy sources is very important for the self-sustaining purpose of the microgrid. The performance of renewable energy sources is highly dependent on weather conditions. Of these, the study of solar energy generation is an important part of the energy management system of the microgrid. This page, prepared with all this information in mind, allows the user to analyze meteorological data as well as to create solar energy generation prediction models and compare their prediction results with real power generation data. Solar page layout comprised four parts. These are as follows: the "Model Selection" section consists of selecting the machine learning algorithm that we will use to create forecast models, the model creation button called "Prediction Result," as well as a combobox where we can select one of the meteorological data, and "Weather Data" button used to plot the data type we selected. The page is presented in Fig. 7.

The middle section is divided into two and is reserved for the display of graphics. Predicted power generation data and real power generation data are plotted and are shown on the middle top part. Each time the user presses the "Prediction Result" button, the GUI regenerates the prediction model using the selected machine learning algorithm and visualizes the results together with the real data. Also on this page, the user can select one of seven different meteorological data (ambient temperature, global radiation, diffuse radiation, ultraviolet, wind velocity, precipitation, and atmospheric pressure) and visualize it in the lower middle part. Thus, it can be examined how meteorological conditions affect renewable energy sources.

The far right side of the page is apportioned for displaying the accuracy percentages of the predicted models to the user. In Fig. 8, it is seen that EL-based algorithms give better results than other machine learning algorithms. In this section, the user can also save these results. When the "Save" button is pressed, the results are saved in a desired location as .txt extension.

D. Wind Page

Another renewable energy source likely to be in microgrids is wind turbines. The energy management system GUI assumes that several wind turbines are used in this microgrid. The correlation map, known as the heat map, which shows the relationship between the data obtained from the structural parts of the turbine and the power data produced by the turbine, is positioned in the middle of the page. The left side of this four-part page is devoted to machine learning algorithm selection and model creation for the prediction model. In addition, with the heat map button, a heat map graph of the data of a turbine can be drawn in the middle upper section. On the right side of the page, the GUI presents the accuracy values of the predicted models and the error values to the user using the MSE, RMSE, MAPE, SMAPE, and MAE error functions. Thus, the user will have access to valuable information about the reliability of the prediction models. The wind page is shown in Fig. 9.

The bottom part of the page is allocated to the visualization of the forecast results together with the actual values. Thanks to the

Fig. 9. Wind page where the power generation prediction of the wind turbines in the microgrid can be performed and examined.

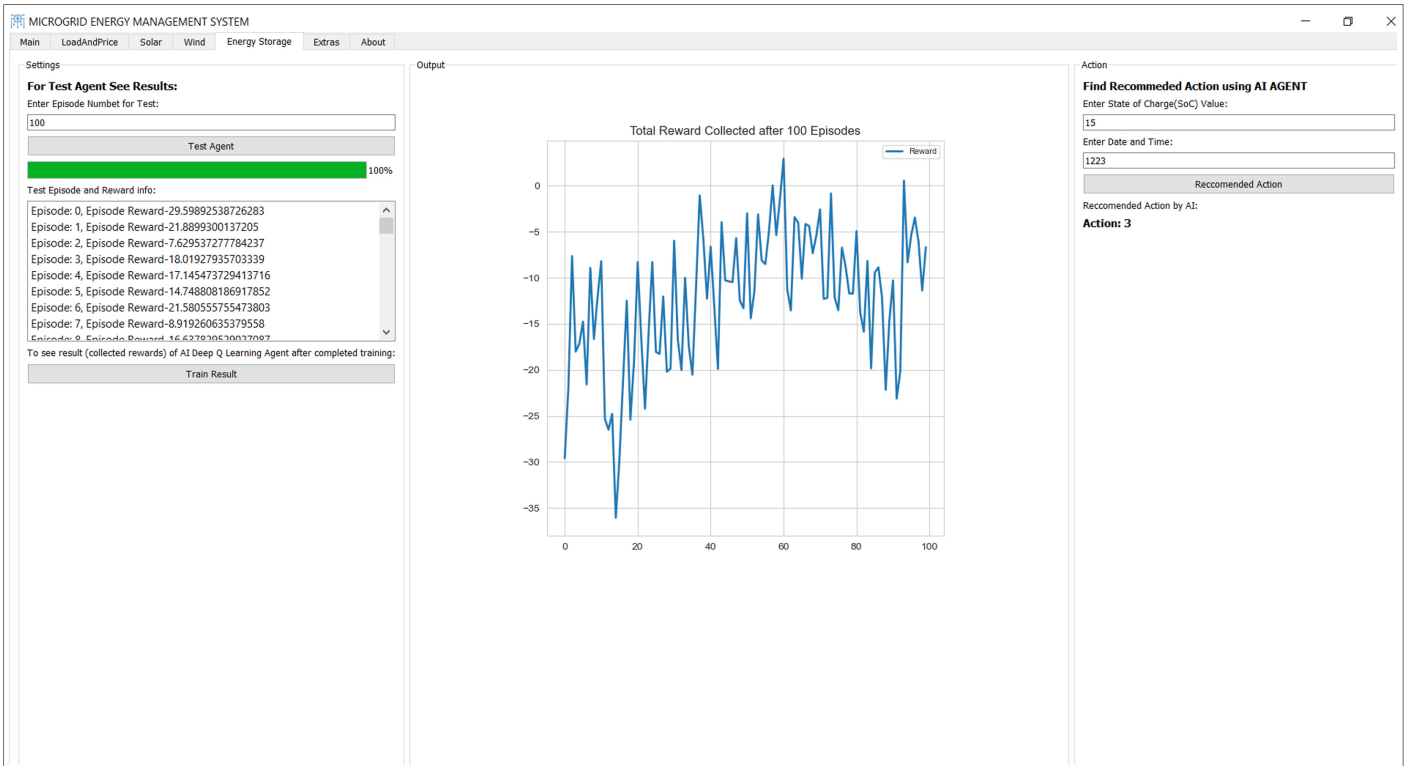

Fig. 10. Energy storage control and RL agent performance analysis page.

100-hour visualization in this wide area, the user can observe how the forecast model behaves in sharp transitions.

E. Energy Storage Page

The use of the energy storage unit is decisive for both the profitability and the autonomous operating behavior of a microgrid. The designed GUI uses deep RL agent in addition to the user as a decision-maker. This agent is focused on learning how to use the ESS effectively. For this purpose, the agent trains itself using prediction models created on other pages. This page has been prepared to examine the results of this RL agent's training. This page consists of three parts: training the agent (on the left), visualizing the performance output (middle part), and getting the trained agent's action suggestion (on the right). It is shown in Fig. 10.

After the user enters the number of episodes he wants to train the agent for, he starts the training by pressing the test agent button. Created using DDQN in the GUI, the RL agent trains and develops itself on the basis of trying and collecting rewards throughout the number of episodes entered. At the end of the training, it shows the reward values collected in each episode to the users via a listbox for them to examine. Additionally, the return values collected during the number of episodes entered are visualized in the middle. An action suggestion can be made on the far right according to the time and SoC information entered from the reliable RL agent, which was observed to collect meaningful rewards as a result of the training.

F. Extras and About Page

The last two pages available in the GUI are called "Extras" and "About". The "Extras" page has a layout structure that will form the basis for possible improvements. The GUI structure designed is open to development and for features that the designer might want to add, for example, it may be desirable to visualize information from various sensors in the microgrid or features for the control of various switches and servos. For this reason, the "Extras" page provides ease of use to the developer. The "About" page provides information about the AI-based energy management system, which is the main work of this designed interface.

IV. CONCLUSION

An AI-based energy management system for microgrids can provide many advantages besides improving profitability and autonomous working conditions. With this basic impulse, an AI-based energy management system using deep learning and machine learning algorithms together with deep RL agent is designed as the superstructure of this study. In this study, a GUI design suitable for the dynamic energy management system, which is the last step and whose infrastructure has been completed, is explained. The designed GUI presents and visualizes many real or predicted data of the microgrid to the user. Furthermore, it offers a control action suggestion by using deep RL agent for productive energy management. The user makes a decision in line with these recommendations and can take the action. It can predict current solar power, wind power generation, dynamic price, and load demand with high accuracy while forecasting meteorological data for the coming days and hours. It can examine the performance of the deep RL agent, which is the most important unit of the GUI. This GUI design, which is open to development, provides a strong foundation for dynamic energy management of the microgrid. This work can be improved by using deep RL agents to be created with different algorithms. In addition, more up-to-date deep learning and machine learning models can be adapted to the system.

Peer-review: Externally peer-reviewed.

Author Contributions: Concept – N.A., İ.G.; Design – N.A., i.G.; Supervision – İ.G.; Data Collection and/or Processing – N.A., İ.G.; Analysis and/or Interpretation – N.A., İ.G.; Literature Review – N.A., N.G.; Writing – N.A., İ.G.; Critical Review – N.A., İ.G.

Declaration of Interests: The authors have no conflicts of interest to declare.

Funding: The authors declared that this study has received no financial support.

REFERENCES

1. A. Cagnano, E. De Tuglie, and P. Mancarella, "Microgrids: Overview and guidelines for practical implementations and operation," *Appl. Energy*, vol. 258114039, 2020. [CrossRef]
2. S. Aslam, H. Herodotou, S. M. Mohsin, N. Javaid, N. Ashraf, and S. Aslam, "A survey on deep learning methods for power load and renewable energy forecasting in smart microgrids,"' *Renew. Sustain. Energy Rev.*, vol. 144, p. 110992, 2021. [CrossRef]
3. Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444 ,2015, 2015. [CrossRef]
4. S. Choudhury, "Review of energy storage system technologies integration to microgrid: Types, control strategies, issues, and future prospects," *J. Energy Storage*, vol. 48, p. 103966, 2022. [CrossRef]
5. Y. E. García Vera, R. Dufo-López, and J. L. Bernal-Agustín, "Energy management in microgrids with renewable energy sources: A literature review," *Appl. Sci.*, vol. 9, no. 18, p. 3854, 2019. [CrossRef]
6. T. Gao, and W. Lu, "Machine learning toward advanced energy storage devices and systems," *iScience*, vol. 24, no. 1, 2021, p. 101936.
7. C. A. Marino, and M. Marufuzzaman, "A microgrid energy management system based on chance-constrained stochastic optimization and big data analytics," *Comput. Ind. Eng.*, vol. 143, p. 106392, 2020. [CrossRef]
8. J. J. Yang, M. Yang, M. X. Wang, P. J. Du, and Y. X. Yu, "A deep reinforcement learning method for managing wind farm uncertainties through energy storage system control and external reserve purchasing," *Int. J. Electr. Power*, vol. 119, p. 105928, 2020. [CrossRef]
9. M. F. Roslan, M. A. Hannan, P. J. Ker, M. Mannan, K. M. Muttaqi, and T. I. Mahlia, "Microgrid control methods toward achieving sustainable energy management: A bibliometric analysis for future directions," *J. Cleaner Prod.*, vol. 348, p. 131340, 2022. [CrossRef]
10. Y. Liu, D. Zhang, and H. B. Gooi, "Optimization strategy based on deep reinforcement learning for home energy management," *CSEE J. Power Energy Syst.*, vol. 6, no. 3, pp. 572–582, 2020.
11. E. Dokur, S. Ceyhan, and M. Kurban, "Analysis of wind speed data using Finsler, Weibull, and Rayleigh distribution functions," *ELECTRICA* , vol. 22, no. 1, pp. 52–60, 2022. [CrossRef]
12. R. S. Sutton, and A. G. Barto, *Reinforcement Learning: An Introduction*, 2nd ed. Cambridge, MA, USA: MIT Press, 2020.
13. S. Hochreiter, and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997. [CrossRef]

Mr. Aksoy earned his B.Sc.in Electrical and Electronics Engineering. Then, he received his Master's degree from the Temple University, USA. He is a Ph.D. candidate at Istanbul Technical University and a member of the Smart Grid lab at ITU. His research interests include reinforcement learning, microgrids, smart grids, and machine learning applications in power engineering and related fields.

Prof. Genc received the B.Sc. degree in electrical engineering from Istanbul Technical University, the M.Sc. degrees in electrical engineering, systems and control engineering, and systems science and mathematics from Istanbul Technical University, Boğaziçi University, and Washington University, respectively. After receiving the doctor of science (D.Sc.) degree in 2001 from Washington University in St. Louis, he joined Istanbul Technical University where he is currently a Professor in the Department of Electrical Engineering. His research interests include power system dynamics and stability, smart grids, optimization and control applications in power engineering.