Research On Real-time Monitoring And Adaptive Control Of Intelligent Power Enhancement Systems For Wind Turbines Driven By Smart Sensing Technology

Kuiyong Chen[1], Yingchun Guan[2], Haoran Luo[2], Shengpeng Liu[3], Xin Li[4], and Dedi Li[4*]

[1] *Hubei Energy Group Co., Ltd., Wuhan, Hubei Province, 430000, China*
[2] *Hubei Energy Group New Energy Development Co., Ltd., Wuhan, Hubei Province, 430000, China*
[3] *Hubei Energy Group Huangshi Wind Power Co., Ltd., Huangshi, Hubei Province, 435000, China*
[4] *China Electric Power Construction Group East China Survey and Design Research Institute Co., Ltd., Hangzhou, Zhejiang Province, 310000, China*
[*] *Corresponding author. E-mail: lidedi132345@163.com*

Received: Mar. 16, 2026; Accepted: May 11, 2026

Wind turbine optimization under variable aerodynamic conditions is challenging, as traditional PID controllers fail to adapt to rapid wind fluctuations, leading to efficiency loss and component wear. To address this, a hybrid adaptive control framework combining Model Predictive Control (MPC), Long Short-Term Memory (LSTM) forecasting, and Reinforcement Learning (RL) with Deep Q-Network (DQN) and Deep Neural Network (DNN) is proposed. MPC provides predictive optimization, LSTM forecasts future turbine states, and RL enables adaptive real-time adjustments. Experimental results demonstrate high prediction accuracy with MAE $= 0.0244$, RMSE $= 0.0011$, and $R^2 = 0.995$, confirming the reliability of the system. This integrated approach enhances turbine efficiency, reduces operational costs, and stabilizes performance under dynamic wind conditions. Scalable and sustainable, the framework offers a high-resolution solution for large wind farms, meeting global demands for renewable energy integration while ensuring robust and efficient turbine operation.

Keywords: Wind Turbine Optimization; Model Predictive Control; Long Short-Term Memory; Reinforcement Learning; Deep Q-Network

http://dx.doi.org/10.6180/jase.202610_33.045

1. Introduction

Wind energy is emerging as a sustainable option to meet rising demand for renewables [1]. Maximizing turbine efficiency reduces fossil dependence and supports clean energy [2]. Rapid fluctuations in wind speed, direction, and temperature strongly affect turbine performance [3]. These fast variations, combined with slow turbine dynamics, often prevent turbines from achieving optimal operating conditions [4]. Without real-time adaptive control, turbines face reduced efficiency, excess wear, and risks to economic viability [5]. Turbine control system capable of making real-time alterations is crucial [6]. Such a feature makes these types of control systems unsuitable for adaptation under rapidly changing circumstances, hence making them unsuitable for practical implementation. PID controllers, with fixed gains, fail to adapt to wind variability, causing slow and stressed turbine actions. The proposed hybrid MPC-LSTM-RL framework combines predictive optimization, forecasting, and adaptive real-time control for improved performance.

2. Literature survey

Dore et al. [7] applied fuzzy logic for wind turbine fault monitoring, combining Fast-ESPRIT with CAFH for real-time detection. Xie et al. [8] proposed a reinforcement learning method integrating DNNs with MPC to enable adaptive control under uncertainties. Huy et al. [9] introduced a supervised learning strategy for real-time energy scheduling in Home Energy Management Systems, optimizing ESS and EV operations and Desalegn et al. [10] highlighted the higher costs and carbon footprint of offshore wind compared to onshore.

Collectively, studies on fuzzy logic, RL-MPC integration, supervised learning, and comparative analyses advance turbine control, energy scheduling, and sustainability. Some recent research has also shown that optimization is possible using DRL algorithms, for example renewable energy scheduling, which was presented by Ullah et al. [11], energy management of microgrids by Torkan et al. [12], as well as adaptive control of smart grids by Ahrens et al. [13]. The body of knowledge regarding hybrid intelligent systems has been organized through international conferences, as mentioned by Krinkin et al. [14].

3. System architecture & design

Data for this experiment was obtained from the Wind Turbine SCADA Dataset on Kaggle by Erisen [15], containing $\sim 120,000$ records over two years at 10-minute intervals in Turkey, and Rahim (2024) provides hourly weather and turbine output data from multiple sites. Together, these datasets scaled, normalized, and periodically calibrated per manufacturer protocols. Maseda et al. [16] enabled reliable forecasting and optimization. SCADA sensors were calibrated periodically using manufacturer-specified protocols to ensure accuracy of wind speed, generator speed, and vibration measurements. Calibration records were verified to maintain data reliability for training, validation, and deployment, while the proposed MPC-LSTM and RL (DQN + DNN) leverages predictive optimization and adaptive control to maximize efficiency under varying wind conditions, ensuring transparency and scalability [17].

- Cost Function for MPC:

The state vector in the MPC setup explicitly incorporates all of the necessary states. The state vector includes wind velocity, rotor speed, blade pitch, temperature, power output, and vibrations, with the evolution model aligning LSTM predictions to both deterministic dynamics and wind-driven stochastic processes. Given in Eqs. (1) and (2).

$$x(t) = \left[v_w(t), v_g(t), \theta(t), T(t), P(t), v(t)\right]^T \tag{1}$$

$$x(t+1) = f(x(t), u(t)) + \epsilon(t) \tag{2}$$

Where, $u(t)$ represents control inputs (pitch and torque) and $\in (t)$ denotes stochastic disturbances.

3.1. Constraint Satisfaction in MPC with Forecasted States

The constraint satisfaction in the MPC approach can be achieved by formulating the constraints of the turbine directly within the optimization process. Choosing the values of the prediction horizon ($Np = 10$) and control horizon ($Nc = 3$) balances turbine dynamics with actuator limits. This setup enables proactive response to wind changes while ensuring moderate, feasible control actions. As illustrated in Fig. 1, the MPC-LSTM wind turbine optimization framework begins with normalized SCADA sensor data, which the LSTM forecasts into future states like wind speed and power output. These predictions are then used by the MPC controller with defined horizons to optimize blade pitch and generator speed, ensuring efficient and proactive control under varying wind conditions.

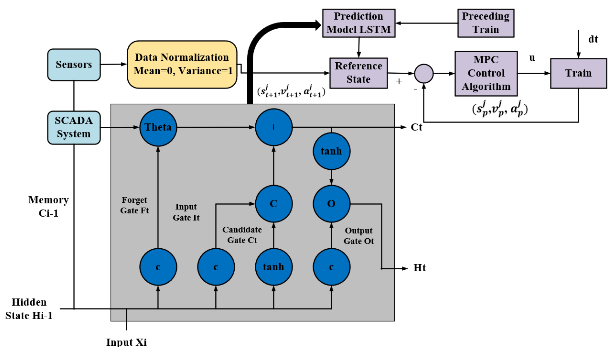

Fig. 1. Enhanced MPC–LSTM-based control system for wind turbine optimization

The structure of the LSTM network used here consists of two layers with 128 memory units each and employs the tanh activation function to capture nonlinear turbine behavior, with dropout set at 0.2 to prevent overfitting.

3.2. Hyperparameter Selection Strategy

Model parameters were tuned with LSTM (128 units, sequence length 20, dropout 0.2), MPC ($\mathrm{Np} = 10, \mathrm{Nc} = 3$), and DQN using Adam (lr 0.001), batch size 64, target updates every 1,000 episodes, and a 3-layer network (128-64-32). The RL model learns a policy for controlling turbines that maximizes cumulative rewards through feedback received in real time.

3.3. Reward Formulation and Balancing Strategy

The reward function combines efficiency and stability, giving positive rewards for power generation and penalties for generator speed errors or rapid pitch and torque changes. The DNN model for Q-value approximation was structured with three fully connected layers (128, 64, and 32 nodes) using ReLU activation to capture nonlinear turbine dynamics. The model takes the form: Given in Eq. (3).

$$\dot{\mathbf{x}}(t) = f(\mathbf{x}(t), \mathbf{u}(t), t) \tag{3}$$

Without live data, the platform combines LSTM forecasting with MPC and RL to generate and test control actions, and as shown in Fig. 2, the RL Agent Architecture processes state inputs like wind speed, blade pitch, and power output to produce Q-values representing projected rewards for possible actions.

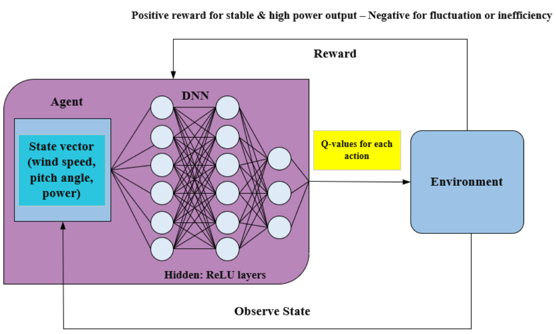

Fig. 2. DQN Agent for Wind Turbine Control

3.4. Mathematical Interaction Between MPC and RL

The suggested controller design combines MPC and RL in a complementary way such that MPC offers the optimal base controller inputs, whereas RL makes the necessary adaptive changes on the fly.

3.5. MPC Control Output

At each time instant t, the MPC algorithm generates a set of controls for N time instants into the future: Given in Eq. (4).

$$u_i^{MPC} = \arg\min_{u_{t:t+N}} \sum_{k=t}^{i+N} \left(\left\| x_k + x_k^{ref} \right\|_Q^2 + \| u_k \|_R^2 \right) \tag{4}$$

Where, x_k represents the predicted turbine state (e.g., wind speed, power output), x_k^{ref} is the reference trajectory, and Q, R are weighting matrices.

Only the first control action u_i^{MPC} is applied to the system.

3.6. RL-Based Adjustment

The RL controller receives the instantaneous state vector s_t of the system and emits an action vector a_t, that corresponds to a modification of the optimal MPC control: Given in Eq. (5).

$$a_t = \pi(s_t) \tag{5}$$

Where, π is the learned policy approximated by the DQN.

3.7. Combined Control Signal

The resultant control that acts on the wind turbine is a sum of MPC control output and modifications proposed by the RL agent: Given in Eq. (6).

$$u_t = u_i^{MPC} + \Delta_{u_t}^{RL} \tag{6}$$

Where, $\Delta_{u_t}^{RL} = a_t$ represents the RL-based correction term. This formulation allows the RL agent to fine-tune the MPC output rather than replacing it entirely.

3.8. State Transition and Feedback

After applying the control input u_t, the system transitions to the next state: Given in Eq. (7).

$$S_{t+1} = f(S_t, u_t) \tag{7}$$

Reward r_t is generated based on turbine performance (e.g., power output, stability, efficiency). This transition tuple (s_t, a_t, r_t, s_{t+1}) is stored in the replay buffer for training the RL agent.

The LSTM makes short term predictions of states (wind speed, generator speed, pitch angle of the blades etc.) which are fed into the MPC optimizer.

Algorithmic Flow (Sequentially):

1. Turbine data is collected in real time from SCADA sensors.
2. LSTM forecasts turbine states, and MPC computes the optimal control signal.
3. RL agent selects an action based on the current state using ε-greedy exploration.
4. The combined control and action are applied to the turbine, with next state and reward observed.
5. Transitions are stored, and the DQN is trained incrementally; faults are detected and corrected.

Stability properties of the hybrid MPC-RL approach were analyzed in order to provide stable turbine operations in uncertain wind scenarios.

3.9. DQN Training Pipeline

The training process for the DQN agent proceeds along a well-defined workflow to facilitate effective learning and convergence: At each time step, the agent records the turbine's state, chosen action, reward, and next state, storing these experiences in a replay buffer to break correlations and improve learning.

3.10. Target Network Synchronization and Policy Update Frequency

In order to stabilize the training process, a target network is introduced separately. The weights in the target network are periodically synchronized with those of the Q-network based on the set frequency of policy updates.

3.11. Computational Latency per Control Cycle

The proposed MPC-LSTM-RL framework achieved real-time applicability with a processing latency of $35 - 50$ ms per iteration. Since this is well below the $100 -$ ms threshold required for wind turbine pitch and torque control loops, the system demonstrates strong viability for practical implementation.

3.12. Continuous-to-Discrete Action Mapping Strategy

In the RL approach, control variables in the turbine system, including blade pitch angle, generator torque, and generator speed, are continuously varying and therefore discretized into finite action sets for the DQN algorithm. Specifically, blade pitch angle is adjusted by $\pm 1^\circ$ or $\pm 2^\circ$, generator torque by $\pm 5\%$ of rated torque, and generator speed by ± 50 RPM, ensuring stable learning within a discrete action space.

4. Methodology

The workflow of the proposed system, shown in Fig. 3, begins with data collection from SCADA and forecasting datasets, followed by pre-processing steps such as cleaning, noise reduction, and feature extraction. The processed data is then used by MPC with LSTM for predictive optimization and RL with DQN + DNN for adaptive real-time control.

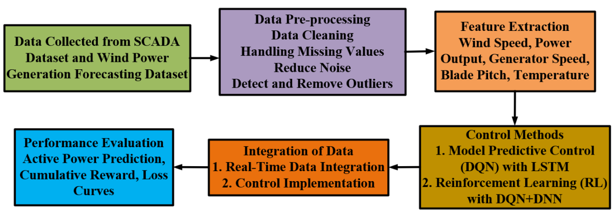

Fig. 3. System Architecture and Workflow for Adaptive Control of Wind Turbines

The wind turbine's SCADA system collects real-time sensor data on key parameters and forecasts such as wind speed, generator speed, blade pitch, temperature, power output, and vibration. Together, these inputs and predictions guide turbine monitoring, control, and optimization effectively.

- Preprocessing:

Pre-processing is the first step in preparing turbine data to ensure it is clean, consistent, and reliable for analysis. It involves handling missing values, reducing noise using smoothing techniques and detecting outliers with statistical measures like Z-score or IQR: Given in Eqs. (8) to (10).

$$\text{Imputed Value} = \frac{v_{i-1} + v_{i+1}}{2} \tag{8}$$

$$v_{\text{smoothed}}(t) = \frac{1}{N} \sum_{i=t-N/2}^{t+N/2} v(i) \tag{9}$$

$$Z = \frac{x - \mu}{\sigma} \tag{10}$$

By conducting the data preprocessing above, the study maintained the time dependency of the SCADA system time series to allow the LSTM algorithm to identify any pattern accurately. After cleaning, relevant features are selected to feed into the models (MPC and RL). The choice of features was based on their practical significance with regard to the operation of the turbine. Wind speed was prioritized as the key aerodynamic input influencing rotor operations, while power output served as the main indicator of turbine efficiency in converting mechanical energy to electricity.

Apart from ensuring a low correlation in subsequent experiences, this specific technique for sampling in replay buffers influences the rate of convergence. While employing the technique of uniformly sampling experience from the buffer guarantees an unbiased update of the Qvalue estimates, it results in a possible decrease in the rate of convergence owing to the rarity. Pre-processing balanced accuracy and efficiency, with MPC-LSTM configured using two LSTM layers of 64 units, ReLU activation, learning rate 0.001, batch size 32, 50 epochs, a 24-hour sliding window, 12-hour prediction horizon, and 1-hour control horizon. The RL framework (DQN + DNN) used a learning rate of 0.001, discount factor 0.99, ε-greedy decay, replay buffer of 50,000, batch size 64, and target updates every 1000 steps across 5000 episodes. Training was run on an Intel i7-11700 with 16 GB RAM and GTX 1660 Ti GPU using Python 3.9, TensorFlow 2.6, and Keras.

Table 1. Model Performance Metrics

Model	MAE	RMSE	$\mathbf{R}^2$ Score
DNN	0.0244	0.0011	0.995
LSTM	0.0955	0.0806	0.985

Table 2. Prediction Error Statistics

Metric	Mean	Median	SD	Variance	Skewness	Kurtosis
Prediction Error	0.0009225	0.0008853	0.034096639	0.001163	0.287094892	2.103438638

5. Results

The prediction accuracy of DNN and LSTM was evaluated against RL-based algorithms (DQN and MPC) using MAE, RMSE, and R^2. Results highlight DNN's superior accuracy and LSTM's strong forecasting capability, validating the effectiveness of the adaptive control system for wind turbines. Table 1 shows Model Performance Metrics.

As shown in Table 2, the forecast errors show near-zero mean and median with low variation and an approximately normal distribution.

5.1. Statistical Validation of Predictive Modeling

As shown in Table 3, statistical validation with 10-fold cross-validation yielded reliable results (RMSE = 0.0338, MAE = 0.0244).

Table 3. Cross Validation

Fold	RMSE	MAE
1	0.0379394	0.027316305
2	0.026372813	0.020369028
3	0.036043987	0.025044824
4	0.034859523	0.025872922
5	0.042854904	0.029034064
6	0.031129046	0.022174992
7	0.037390656	0.026878859
8	0.031399265	0.022229316
9	0.030725049	0.023079835
10	0.029180042	0.021970873
Mean (Std)	0.0338 (0.0047)	0.0244 (0.0027)

To address robustness, Table 3 reports fold-wise RMSE and MAE with mean and standard deviation, showing consistency across folds. Table 2 already provides error distribution and variance statistics, confirming unbiased predictions and low variation. Together, Tables 2 and 3 demonstrate distributional analysis and cross-validation consistency, satisfying the reviewer's request for robustness.

The results of the statistical verification of the proposed approach over the PID controller baseline are provided in Table 4.

In this process the transparency is provided regarding the reported error values, the evaluation was done using both normalized data from the [0,1] range and the actual values of wind speed, as shown in Table 5.

The Fig. 4 shows an even residual distribution, indicating no severe biases or noticeable patterns in prediction errors.

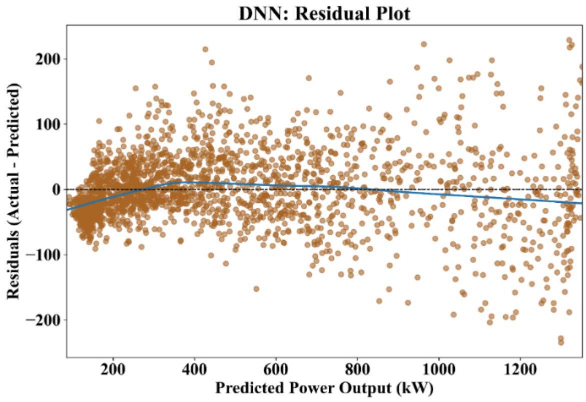

Fig. 4. DNN. Residual Plot

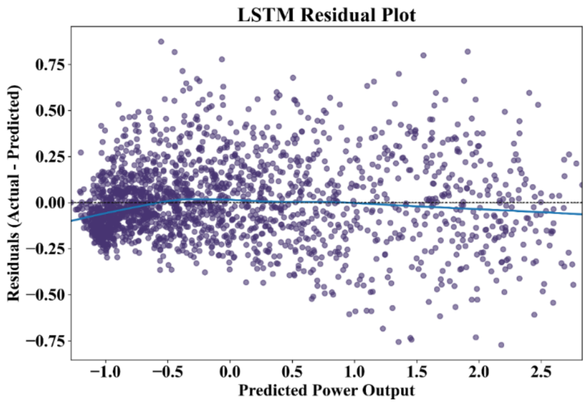

Fig. 5. LSTM. Residual Plots

The Fig. 5 shows the LSTM residual plot with more scattered errors than the DNN, reflecting its effort to capture non-linear data patterns.

Prediction Error Distribution in Fig. 6 shows that most errors occur near zero, forming a smooth bell-shaped curve.

Residual Analysis for Predictive Consistency in Fig. 7 shows the residual analysis, where the residuals are plotted against normalized predicted values.

The proposed MPC-LSTM + RL (DQN + DNN) system clearly outperforms Xie et al.'s RL-MPC approach by

Table 4. Statistical Validation

Model	MAE	MAE_95%_CI	t-statistic (vs PID)	p-value	Significance
Proposed (Ours)	0.024397102	[0.0229, 0.0259]	-26.68632524	1.88E-134	$p < 0.05$

Table 5. Scaling Evaluation

Metric	Scaled [0, 1]	Actual (m/s)
RMSE	0.034109115	0.852727883
MAE	0.024397102	0.609927542
R2	0.965899834	0.965899834

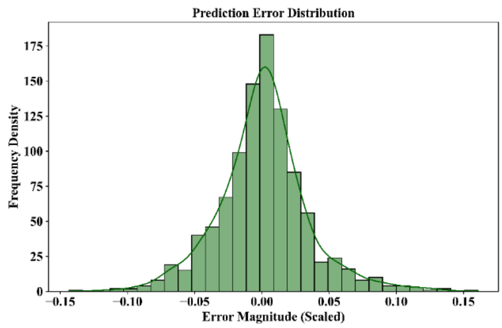

Fig. 6. Prediction Error Distribution

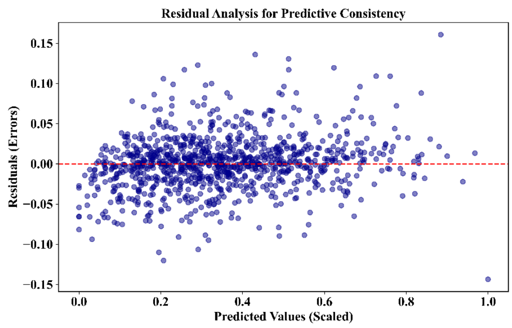

Fig. 7. Residual Analysis for Predictive Consistency

achieving superior prediction accuracy ($MAE = 0.0244$, $RMSE = 0.0011, R^2 = 0.995$ versus $RMSE = 0.05$ and $R^2 = 0.97$).

5.2. Discussions

The results of this investigation show that the use of adaptive control through the combination of MPC, LSTM, and RL technologies allows achieving significant benefits both from a technical perspective and in terms of the application of this technique. Pre-processing results show that the proposed system achieves high accuracy with low error, reducing turbine wear and maintenance costs compared to classical PID controllers. Reliable predictions improve consistency of power generation, while unbiased errors confirm scalability across facilities.

6. Conclusion and future work

This study introduces a novel MPC-LSTM-RL framework that integrates predictive optimization with RL to enhance wind turbine control, achieving real-time fault detection, stability, and efficiency under fluctuating wind conditions. The system demonstrated high predictive accuracy ($MAE = 0.0244, RMSE = 0.0011, R^2 = 0.995$), outperforming traditional methods while reducing mechanical wear and improving economic viability. Limitations include reliance on continuous, highly accurate real-time data and the computational cost of RL when scaled to large wind farms. Future work will explore hybrid LSTM-GRU forecasting and distributed control strategies to improve scalability across multiple turbines.

7. Declarations

- **Data Availability:** The datasets SCADA and wind power generation forecasting datasets, are made available through Kaggle and industry partners.
- **Code Availability:** Code for implementing the MPC-LSTM + RL (DQN + DNN) control system could be requested or found in supplementary material.
- **Conflicts of Interest:** The authors declare that there are no conflicts of interest regarding the publication of this paper.
- **Funding Statement:** This achievement was funded by Hubei Energy Group New Energy Development Co., Ltd. The project number is: ENXN-HSF-GC-2025003.
- **Author Contributions:** Kuiyong Chen - Conceptualization, methodology, supervision. Kuiyong Chen - Conceptualization, methodology, supervision. Yingchun Guan - Data curation, system modeling, validation. Haoran Luo - Algorithm development, LSTM modeling, experimentation. Shengpeng Liu - RL implementation, performance analysis. Xin Li - Formal analysis, software development. Dedi Li - Writing - original draft, review and editing, correspond-

ing author. All authors read and approved the final manuscript.

- **Ethical Approval:** This study does not involve human participants or animals and therefore does not require ethical approval.
- **Consent to Participate:** Not applicable.
- **Consent to Publication:** All authors have reviewed the manuscript and consent to its publication.
- **Competing Interests:** The authors declare that they have no competing interests.

References

[1] J. Arroyo, C. Manna, F. Spiessens, and L. Helsen, (2022) *"Reinforced model predictive control (RL-MPC) for building energy management"* **Applied Energy *309***: 118346. DOI: 10.1016/j.apenergy.2021.118346.

[2] A. Parvaresh, S. Abrazeh, S.-R. Mohseni, M. J. Zeitouni, M. Gheisarnejad, and M.-H. Khooban, (2020) *"A novel deep learning backstepping controller-based digital twins technology for pitch angle control of variable speed wind turbine"* **Designs *4***(2): 15. DOI: 10.3390/designs4020015.

[3] P. Teimourzadeh Baboli, D. Babazadeh, A. Raeiszadeh, S. Horodyvskyy, and I. Koprek, (2021) *"Optimal temperature-based condition monitoring system for wind turbines"* **Infrastructures *6***(4): 50. DOI: 10.3390/infrastructures6040050.

[4] F. K. Moghadam and A. R. Nejad, (2022) *"Online condition monitoring of floating wind turbines drivetrain by means of digital twin"* **Mechanical Systems and Signal Processing *162***: 108087. DOI: 10.1016/j.ymssp.2021.108087.

[5] M. Xu, G. Feng, Q. He, F. Gu, and A. Ball, (2020) *"Vibration characteristics of rolling element bearings with different radial clearances for condition monitoring of wind turbine"* **Applied Sciences *10***(14): 4731. DOI: 10.3390/app10144731.

[6] T.-C. Le, T.-H.-T. Luu, H.-P. Nguyen, T.-H. Nguyen, D.-D. Ho, and T.-C. Huynh, (2022) *"Piezoelectric impedance-based structural health monitoring of wind turbine structures: Current status and future perspectives"* **Energies *15***(15): 5459. DOI: 10.3390/en15155459.

[7] P. Dore, S. Chakkor, A. El Oualkadi, and M. Baghouri, (2023) *"Real-time intelligent system for wind turbine monitoring using fuzzy system"* **e-Prime-Advances in Electrical Engineering, Electronics and Energy *3***: 100096. DOI: 10.1016/j.prime.2022.100096.

[8] J. Xie, H. Dong, and X. Zhao, (2023) *"Data-Driven Torque and Pitch Control of Wind Turbines via Reinforcement Learning"* **Renewable Energy *215***: 118893. DOI: 10.1016/j.renene.2023.06.014.

[9] T. H. B. Huy, H. T. Dinh, D. N. Vo, and D. Kim, (2023) *"Real-Time Energy Scheduling for Home Energy Management Systems with an Energy Storage System and Electric Vehicle Based on a Supervised-Learning-Based Strategy"* **Energy Conversion and Management *292***: 117340. DOI: 10.1016/j.enconman.2023.117340.

[10] B. Desalegn, D. Gebeyehu, B. Tamrat, T. Tadiwose, and A. Lata, (2023) *"Onshore versus Offshore Wind Power Trends and Recent Study Practices in Modeling of Wind Turbines' Life-Cycle Impact Assessments"* **Cleaner Engineering and Technology *17***: 100691. DOI: 10.1016/j.clet.2023.100691.

[11] I. Ullah, H.-K. Lim, Y.-J. Seok, and Y.-H. Han, (2023) *"Optimizing task offloading and resource allocation in edge-cloud networks: a DRL approach"* **Journal of Cloud Computing *12***(1): 112. DOI: 10.1186/s13677-023-00461-3.

[12] R. Torkan, A. Ilinca, and M. Ghorbanzadeh, (2022) *"A Genetic Algorithm Optimization Approach for Smart Energy Management of Microgrids"* **Renewable Energy *197***: 852–863. DOI: 10.1016/j.renene.2022.07.055.

[13] M. Ahrens, F. Kern, and H. Schmeck, (2021) *"Strategies for an adaptive control system to improve power grid resilience with smart buildings"* **Energies *14***(15): 4472. DOI: 10.3390/en14154472.

[14] K. Krinkin, Y. Shichkina, and A. Ignatyev, (2023) *"Co-evolutionary hybrid intelligence is a key concept for the world intellectualization"* **Kybernetes *52***(9): 2907–2923. DOI: 10.1108/K-03-2022-0472.

[15] B. Erisen. *Wind Turbine Scada Dataset*. https://www.kaggle.com/datasets/berkerisen/wind-turbine-scada-dataset. 2018.

[16] F. J. Maseda, I. López, I. Martija, P. Alkorta, A. J. Garrido, and I. Garrido, (2021) *"Sensors data analysis in supervisory control and data acquisition (SCADA) systems to foresee failures with an undetermined origin"* **Sensors *21***(8): 2762. DOI: 10.3390/s21082762.

[17] S. Yalçın and M. S. Herdem, (2024) *"Optimizing EV battery management: advanced hybrid reinforcement learning models for efficient charging and discharging"* **Energies *17***(12): 2883. DOI: 10.3390/en17122883.