EAI Endorsed Transactions
on Energy Web
Research Article EAI.EU

Analysis and Design of Wind Turbine Monitoring System Based on Edge Computing

Xiaoju Yin[1,a], Yuhan Mu[1,b], Bo Li[2,c*], Yuxin Wang[3,d*]

[1] School of Renewable Energy, Shenyang Institute of Engineering, Shenyang 110136, China
[2]College of Information, Shenyang Institute of Engineering, Shenyang 110136, China
[3]College of Computer and Information Engineering, Tianjin Agricultural University, Tianjin 300380, China

Abstract

INTRODUCTION: A wind turbine data analysis method based on the combination of Hadoop and edge computing is proposed.
OBJECTIVES: Solve the wind turbine health status monitoring system large data, time extension, energy consumption and other problems.
METHODS: By analysing the technical requirements and business processes of the system, the overall framework of the system was designed and a deep reinforcement learning algorithm based on big data was proposed.
RESULTS: It solves the problem of insufficient computing resources as well as energy consumption and latency problems occurring in the data analysis layer, solves the problems in WTG task offloading, and improves the computational offloading efficiency of the edge nodes to complete the collection, storage, and analysis of WTG data.
CONCLUSION: The data analysis and experimental simulation platform is built through Python, and the results show that the application of Hadoop and the edge computing offloading strategy based on the DDPG algorithm to the system improves the system's quality of service and computational performance, and the method is applicable to the distributed storage and analysis of the device in the massive monitoring data.

Keywords: monitoring system, Hadoop, edge computing, big data.

Received on 17 November 2023, accepted on 06 April 2024, published on 11 April 2024

doi: 10.4108/ew.5742

1. Introduction

1.1. Progress of related research at home and abroad

With the proposal of carbon peak and carbon neutrality, China's wind power industry is growing rapidly, and the vigorous development of wind power is in line with the country's future energy strategic planning. As an important wind power equipment that converts wind energy into electricity [1], the operation safety of wind turbines has attracted much attention. In order to get more stable wind speed and efficient power generation efficiency, WTGs are developing in the direction of large-scale, and their blades are becoming more and more large-scale, from gear box type to direct-drive type, fixed speed to variable speed, fixed pitch to variable pitch, land-based to ocean-based, and the power is also raised to as high as MW level, and the structure of WTGs is also developing in the direction of towering, and the cost of wind turbines, technical difficulties, and safety risks increase with the increase in their height. The cost, technical difficulties and safety risks of wind turbines increase with their height. With the increase in height of WTGs, the ratio of horizontal and vertical dimensions is becoming more and more obvious, and WTGs are subject to strong external load forces when encountering natural disasters such as earthquakes, typhoons, etc., which are prone to large vibration changes, and wind turbine breakage or even collapse accidents. In order to guarantee the safe, efficient and reliable operation of wind turbines, wind farms are constantly developing towards digitalization, and the number of various wind turbine intelligent terminal devices installed has increased significantly, and their internal structure often fails, leading to frequent accidents in wind farms, so real-time monitoring of

[a] lnsolyxj@163.com, [b]2275936994@qq.com, [c*]Corresponding author: 498806972@qq.com, [d*]261235654@qq.com

data on the status of wind turbines, timely discovery of abnormal status of wind turbines, and the realization of predictive maintenance of wind turbines have become new problems to be solved at present problem [2]. Therefore, the development of WTG monitoring system has important engineering application value.

Efficient calculation, analysis and storage of large amounts of data from wind turbines is a major challenge. Cheng Jiangzhou et al [3] established a wind turbine fault diagnosis and risk prediction model by using improved Bayesian network for the problem of frequent failures of wind turbines caused by severe weather; Zhang Jinhua et al [4] established an optimal dispatch model of wind farms, which effectively reduces the loss of wind farms; Song Wei et al [5] designed a wind turbine bearing fault diagnosis method based on improved noise reduction self-encoder for the problem of weak bearing vibration signal characteristics and difficult diagnosis; Han Pingping et al [6] used the characteristics of Hadoop for the problem that the current massive data of wind turbines cannot be analyzed and stored efficiently, and applied it to the storage and analysis of wind power data. As numerical simulation requires high computational power and is time and resource consuming, learning from past experiences and lessons learnt, this paper utilizes the edge network environment in conjunction with a distributed system infrastructure (Hadoop) [7]-[8] to provide a reliable and effective solution for multi-source data in monitoring systems.

1.2. The main research ideas and content of this paper

In this paper, a WTG monitoring system based on edge computing and Hadoop is proposed, and the overall framework and content of the system are designed according to the requirements of the system. With multiple WTG targets, the WTG intelligent terminal equipment has limited computational resources, and certain WTG terminal equipment executes the modelling tasks with frequent task congestion, lagging, and other abnormal operating conditions, which leads to its execution delay and energy consumption being particularly large. The delay and energy consumption are particularly large. Therefore, servers with computing power are introduced into the data analysis layer of the system as edge nodes to share the computational pressure of terminal devices to improve the task processing rate and reduce energy consumption.

In order to solve the computational offloading decision problem in the process of wind turbine data modelling, a task offloading algorithm for wind turbine vibration modelling based on deep reinforcement learning to achieve multi-data optimization is proposed [9]. Edge computing mainly uses the calculation unloading strategy to process and analyze various collected data. The tasks in edge computing are often highly dynamic [10]-[11]. In the edge computing environment, the energy cost cost of computation offloading becomes smaller at the edge server [12]-[14].

Finally, we analyze and compare the task execution time and task processing efficiency of the system under various computational offloading strategies, obtain the optimal computational offloading scheme, and improve the wind turbine health status monitoring system. In summary, this paper studies the wind turbine status monitoring system based on the combination of Hadoop and edge computing.

2. System requirements analysis

Currently, different types of sensors are installed on wind turbines to collect real-time data such as wind direction, wind speed, nacelle vibration, and temperature and humidity to achieve the monitoring of wind turbine operating conditions. As the operating time of wind turbines advances, the data generated by wind farms is getting bigger and bigger, which is beyond the capacity of data processing and storage of the current monitoring system. In the face of massive wind turbine data, the traditional relational database is difficult to adapt to the development of WTG monitoring data will be biased.

In order to improve the accuracy of the fault diagnosis of the WTG condition monitoring system and reduce the energy loss of the system, the system requirements are analyzed to achieve the following functions:

(a) Reduce the total delay and total energy consumption of the WTG monitoring system when performing tasks.

(b) Real-time monitoring of WTG data.

(c) Capable of realising the function of storing and managing a large amount of raw data from intelligent sensors.

(d) Data analysis of WTGs to determine equipment health and remaining life.

3. Framework design of WTMS

3.1. System framework design

Using the system architecture of "layered design", the statistics and analysis of wind turbine data are processed, and the big data analysis structure is combined with edge computing to achieve the batch collection, analysis and application of wind turbine multi-source data, and the overall framework of the system is shown in Figure 1.

a) Equipment layer: The equipment layer is divided into the climate change adjustment power generation area (Area A) and the equipment fault diagnosis and analysis area (Area B). Zone A collects the temperature and humidity measured by the climate measurement controller, as well as the data on the anemometer and wind direction meter, and delivers them to the platform layer. Zone B delivers the data at the faults on the tower, nacelle, hub and blades through the controller of the wind turbine to the coordination layer to analyze the fault data, characteristics and response information, and carry out screening.

b) Coordination layer: the equipment operation management and fault analysis platform in this layer analyses the fault information from Zone B, sends the obtained response information and response characteristics to the fault analysis platform and database respectively, and uses the calling information, data, power generation, and equipment information transmitted back from the platform layer to be sent to Zone B for fault diagnosis.

c) Platform layer: collects data from the equipment layer, transports the data to the fault analysis platform and files it.

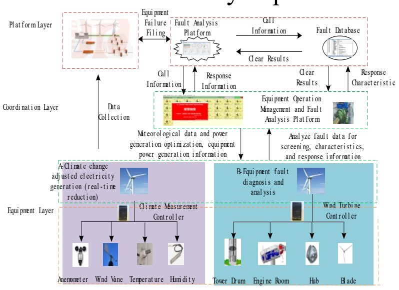

Figure 1. Schematic diagram of the overall system framework

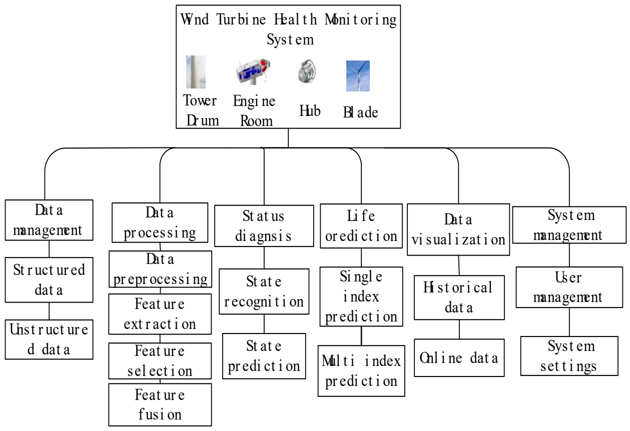

Figure 2. Block diagram of the main functional modules of the system

3.2. System functional design

Each functional module of the wind turbine monitoring system is shown in Fig. 2, which contains the functions of data management and analysis of the tower, nacelle, hub and blades under various working conditions, diagnosis of equipment status, and prediction of equipment health.

(a) Data Management

The module classifies and stores the raw data collected by various types of sensors distributed on wind turbines, in which relational data and non-relational data are stored separately to achieve the import management functions of structured and unstructured data. The module can distinguish structured data and unstructured data according to the type of sensors; the unstructured data is stored in HBase and managed by copying, deleting, transferring, etc., while structured data are stored in MySQL to achieve backup, transferring and deleting functions [15].

(b) Data Processing

In the edge computing environment, the distance between the WTG equipment and the edge server is closer, the data transmission and information exchange is faster, and the energy cost cost of computation offloading at the edge server is small, based on the distributed computing of Hadoop and the deep learning algorithm of edge computing, to play the role of the edge node in the WTG health status monitoring system, and to use the deep reinforcement learning method to formulate the offloading decision and the resource allocation algorithm.

(c) Condition Diagnosis

Diagnosis of the health status of the WTG.

(d) Life Prediction

This function is mainly used to predict the life of WTGs, determine whether the current WTG has reached the life dacay stage, and estimate the remaining life of the equipment.

(e) Data Visualization

Realize real-time monitoring of WTG status, real-time display of WTG status change trends, and timely presentation of WTG equipment maintenance decision.

(f) System administration

System management is divided into user management and system setting, in which user management facilitates users to directly carry out maintenance or update according to various data returned from the system, and system setting makes correct adjustments to WTGs according to real-time updated data to improve the life of WTGs.

4. Construction and analysis of the system model

4.1. System modelling

The system model required for data processing: the WTGs perform a single computational unloading using multiple edge nodes, and the edge nodes can interact with neighbouring edge nodes with real-time information. Let the set of vibration modelling subtasks for WTGs be:

$$R_w = \left\{ R_w^1, R_w^2, \ldots, R_w^L \right\} \tag{1}$$

Among them, R_w contains several sub-tasks, such as vibration correlation analysis, establishment of vibration prediction model for each WTG, model training and evaluation, and calling the optimal model, etc., L is the number of sub-tasks, and w is the serial number of WTGs. The dependency relationship between the sub-tasks of WTG vibration modelling subtasks is expressed as:

$$D_w = \left[e_{i,j}^w \right]_{L\times L}, d_{i,j}^w \in \{0,1\} \tag{2}$$

When $d_{i,j}^w = 1$ denotes that the subtask R_w^j depends on subtask R_w^i, the input of subtask R_w^j is the output of subtask R_w^{i}, when $d_{i,j}^w = 0$ means subtask R_w^j does not depend on subtask R_w^{i}.

During the task offloading process, whether the subtask data is cached or not affects its execution delay and energy consumption, a cache identifier is defined to indicate whether the subtask data is cached at the edge node or not. An offload identifier is also defined to indicate whether the WTG subtask needs to be offloaded to the edge node for computation. When subtask R_w^i is computed at the local smart terminal device, its local computation time and energy consumption are computed as:

$$T_w^i = \frac{c_w^i}{f_w^i} \tag{3}$$

$$E_w^i = k\left(f_w^i\right)^2 c_w^i \tag{4}$$

In the formula: c_w^i is the data volume of the subtask R_w^i ; f_w^i is the computing resources allocated to the subtask R_w^i by the local smart terminal device, and k denotes the energy consumption coefficient.

If the local end device lacks sufficient computational resources, all or part of the subtasks can be offloaded to an edge node with more computational capability for computation. The delay of subtask data transmission to the edge node is calculated as:

$$T_w^{i,up} = \frac{c_w^{i,up}}{r_w^i} \tag{5}$$

$$E_w^{i,up} = p_w^i T_w^{i,up} = \frac{p_w^i c_w^{i,up}}{r_w^i} \tag{6}$$

In the formula: $c_w^{i,up}$ is the amount of data uploaded by the subtask to the edge node, r_w^i is the channel rate at which the data is uploaded; p_w^i is the communication energy power consumed by the subtask to upload to the edge node.

The latency of subtask data processing at the edge node is calculated as:

$$T_w^{i,ha} = \frac{c_w^{i,ha}}{f^i} \tag{7}$$

$$E^{i,ha} = p^{i,ha} T^{i,ha} = \frac{p^{i,ha} c_w^{i,ha}}{f^i} \tag{8}$$

In the formula: $c_w^{i,ha}$ is the amount of data processed by the subtasks offloaded to the edge node, f^i is the computation rate of the edge node, $p^{i,ha}$ is the energy power consumed by the subtask offloaded to the edge node.

The cache hit rate is used to reflect the average probability of a subtask R_w^k requesting a cache file in a given time, denoted by the variable p, which has the mathematical expression:

$$P = \frac{\sum_{k\in L}\sum_{f\in F}\sum_{j\in(L\cup N)} g_k * p_{k,f} * c_{j,f}}{L} \tag{9}$$

In the formula: g_k denotes the probability that a subtask R_w^k requests a cached file; $p_{k,f} = p\left(f \mid k\right)$ denotes the conditional probability that subtask R_w^k requests a cached file f ; and $c_{i,j}$ denotes the state of whether a cached file f is cached at edge node j, the number of subtasks is L, the set of cached content files is F, and the number of edge nodes content with caches is N.

For the wind turbine vibration modelling optimization problem, the computational offloading fusion content caching strategy can be modelled as a multi-source data optimization on the device side, realizing the cache space and the edge side of the device, with the objective function expressed as:

$$\min C_{ij} = \sum_{i=1}^{w}\left[\left(1-z_i\right)C_i^1 + z_i C_i^{off}\right] \tag{10}$$

$$\max p = \frac{\sum_{k\in L}\sum_{f\in F}\sum_{j\in(L\cup N)} g_k * p_{k,f} * c_{j,f}}{L} \tag{11}$$

$$s.t.\begin{cases} C1: 0 \le z_i \le 1, \forall i \in w \\ C2: 0 < f_i \le F_j^{\max}, \forall i \in w \\ C3: \sum_{i\in w} p_i d_i \le H_m, \forall i \in w \end{cases} \tag{12}$$

The offloading decision of the WTG terminal equipment with serial number i is defined as z_i ; p_i is the transmit power of the WTG terminal equipment of i ; d_i denotes the spatial distance between the WTG terminal equipment and the edge node, C_w^1 is the cost cost at the equipment end, and C_w^{off} is the cost cost for the edge to collaborate with the computation at the equipment end; the constraint C_1 denotes that the task offloading decision of each WTG equipment is to be either locally computed or offloaded to the edge node for processing; the constraint C_2 denotes that the sum of computation power at the edge end is to be less than the maximum computing power of any system in the network; the constraint denotes that the total data volume of the equipment at the edge is to be less than the maximum cache space. The constraint $F_j^{\max}$ indicates that the sum of computational power at the edge is less than the maximum computational power of any system in the network; and the constraint C_3 indicates that the total data volume of the devices at the edge is less than the maximum cache space H_m .

The system is based on the optimal solution of offloading decision and computational resource allocation to achieve the purpose of reducing energy consumption. The optimal offloading policy search is achieved according to multiple optimization objectives based on time delay, energy efficiency and cache hit rate, and this multi-objective optimization function is non-convex and the optimal solution becomes more difficult to solve as the number of terminal devices increases. Applying deep reinforcement learning to this type of decision problem has obvious advantages.

4.2. Computational offloading optimisation scheme based on edge computing

Deep reinforcement learning

Deep Learning (DL) belongs to a kind of machine learning, which is to generalize the regular characteristics from the sampled information and then obtain the relevant features, during which it needs to be achieved by exploring the distributed characteristics of the information.

Reinforcement Learning (RL) belongs to a kind of unsupervised learning machine learning, Reinforcement Learning models can be further classified as policy or value function-based learning models, Reinforcement Learning has outstanding advantages in making intelligent decisions, and is able to find the optimal effect of actions in different environments [16].

DQN (Deep Q-Network, DQN) algorithm is improved based on the classical reinforcement learning algorithm Q-learning. However, the application of DQN algorithm is relatively limited, and it can't handle the continuous action control well, which can effectively solve the continuous state space and discrete action space problems. To address this problem on the basis of the DQN algorithm is extended, the DDPG algorithm is proposed, the algorithm can effectively deal with linear data.

Wind turbine health monitoring task unloading based on DDPG

Deep Deterministic Policy Gradient (DDPG) is a DRL specific algorithm based on policy gradient. The principle of DDPG algorithm is shown in Figure 3.

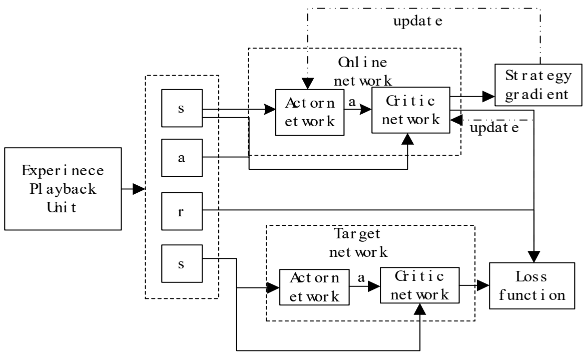

Figure 3. DDPG structure diagram

The problem of modelling wind turbines is optimised based on the deep reinforcement learning algorithm DDPG. The intelligences are deployed at the edge computing server and are responsible for calculating the optimal offloading scheme for the set of subtasks, which is able to make the best decisions in a network environment where the system resources vary over time, taking into account the interactions between the subtask decisions. The DDPG is used to formulate reasonable offloading decisions for the subtasks of the WTG health status monitoring system and to allocate computational resources for the tasks that are offloaded to the edge servers for execution, so that the task execution completion delay is minimized.

4.3. Algorithm simulation validation and performance evaluation

To evaluate the performance of the proposed DDPG offloading strategy, this paper creates an edge computing network environment for simulation and analysis using Python. Q-learning as well as DDPG deep reinforcement learning networks are built through the machine learning open-source platform TensorFlow to achieve the simulation analysis of the offloading of the wind turbine vibration modelling task. Compare the effects of all local computation of the modelling task, computation completely offloaded to the edge server, and computation using Q-learning as well as DDPG. In local computation, all the task data collected by all the WTG end-devices execute the computation on the local devices without offloading the computation. In complete offloading, all task data collected by all WTG terminal devices are offloaded to the edge nodes to perform computation, and the edge node computation resources are evenly distributed. The simulation parameters are set as shown in Table 1.

Table 1. Simulation parameter settings

parametric	set value
Number of wind turbines	10
Number of wind turbine terminal	10
The distance between the equipment and the base station	100 /m
Task data size	500/ kB
Number of tasks	10
Data computing power provided by end devices	1/ (GHz·s^{-1})
Communication energy power consumption of nodes	0.5/ w
Amount of data uploaded to edge nodes	100-500 /kB
Computing power of edge nodes	5/(GHz/s^{-1})
Energy power for task offloading to edge node computing	0.1/ w
Terminal equipment energy consumption factor	10-20
MEC Cache Capacity	1%
Actor Network Learning Rate	0.01
Critic Network Learning Rate	0.01

5. Results analysis and comparison

As the number of WTG end-devices continues to increase the total energy consumption of fully local computation, fully offloaded computation, Q-learning based and offloaded based DDPG algorithms are increasing. As the number of terminal devices increases, the edge nodes and terminal devices cannot provide sufficient computational resources for the modelling task of each WTG device. When the number of WTG terminal devices is less than 5, the total energy consumption of both Q-learning-based and DDPG-based algorithms offloading is small and the difference is not significant. When the number of WTG end devices is greater than 5, the offloading strategy based on the DDPG algorithm produces lower total energy consumption than the other three offloading strategies. Overall, the computational offloading strategy based on the DDPG algorithm is more effective in reducing the total energy consumption compared to the three strategies of completely local computing, completely offloading computing, and based on the Q-learning algorithm. The effect of the number of WTG terminal devices on the total energy consumption is shown in Fig. 4.

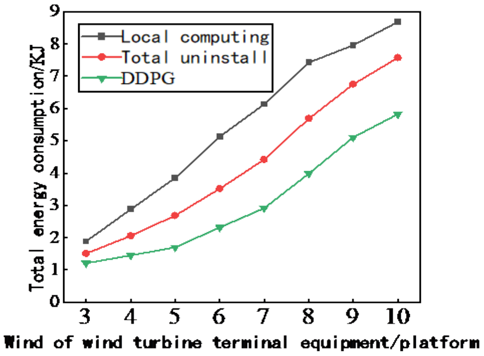

Figure 4. The influence of the number of fan towers on the total energy consumption of edge computing

As the computational power of the edge nodes increases, more computational resources can be provided to the WTG end devices, and the total energy consumption for completing the offloading of the WTG modelling task is decreasing. As shown in Fig. 5, the impact of edge node computing power on total energy consumption. When the computational power of the edge node is less than 2.5 GHz/s, performing local computation has a more obvious effect on reducing energy consumption than the other three offloading strategies. The computational power of the edge node is weaker, and the processing efficiency of performing local computation at the WTG terminal equipment will be higher compared to offloading the task to the edge node. When the computing power of the edge node is greater than 3GHz/s, the offloading strategy based on DDPG is more effective in reducing energy consumption than completely local computing, completely offloading, and offloading strategy based on Q-learning. This indicates that when the computational capacity of the edge node is sufficient, the edge node can share some of the subtasks of the WTG end-device modelling task, and the computational offloading strategy can be used to efficiently use the edge node computational resources and significantly reduce the energy consumption. The DDPG-based offloading strategy is better than the complete offloading and Q-learning-based offloading strategies in reducing energy consumption. In summary, the proposed DDPG-based computational offloading strategy for WTG modelling significantly outperforms the fully local, fully offloaded and Q-learning-based offloading strategies in reducing the total energy consumption when the edge computing capacity is sufficient [17].

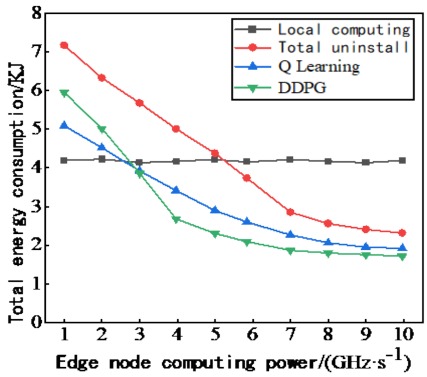

Figure 5. The impact of edge node computing power on total energy consumption

In order to further evaluate the comprehensive performance of the computational offloading strategies for WTG modelling based on the DDPG algorithm, the number of WTG terminal devices continues to increase within the processing range of the edge nodes, and the processing time of each computational offloading strategy for multiple WTGs is compared. As shown in Fig. 6, as the number of WTG terminal devices increases, the task data to be processed also increases, so the computation time of these four strategies is increasing. If the processing is done only by the WTG terminal equipment with weak computational capability, the processing time is too long when performing tasks with a large amount of data.

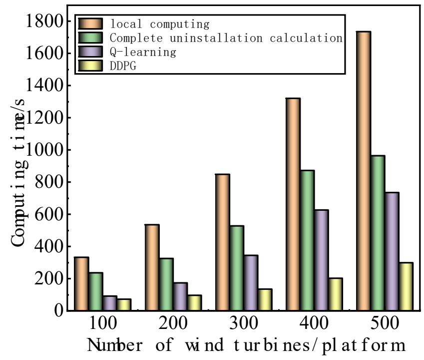

Figure 6. Comparison of computation time of each algorithm

Comparison of the computation time of the four strategies shows that when the number of WTG equipment is 300, the computational speed of the computational offloading strategy based on the DDPG algorithm is 5.1 times faster than that of performing local computation, 3.4 times faster than that of complete offloading computation, and 1.8 times faster than that of the computational offloading strategy based on the Q-learning algorithm, and the DDPG-based WTG modelling proposed in this paper The computational offloading strategy based on DDPG proposed in this paper reduces the task execution time and improves the task processing efficiency. In summary, the DDPG-based computational offloading strategy for WTG modelling is significantly better than other offloading strategies in terms of computational speed.

6. Summary

This paper applies Hadoop and edge computing offloading strategy based on DDPG algorithm in the wind turbine monitoring system to get the following conclusions: meets the demand of large-scale wind turbine data in storage and analysis; solves the problem of data, computing and storage in wind turbine operation; through the task offloading decision in the data analysis layer in the wind turbine monitoring system and server resource allocation solving problem Joint modelling is carried out to maximize the cache hit rate and minimize the energy cost cost of user equipment. Tests show that the computation offloading strategy based on the DDPG algorithm improves the computational efficiency of the system, reduces time consumption and energy loss, and has a good application value for the diagnosis and analysis of equipment at the edge.

However, there are still some parts of the health monitoring of WTGs that are worthy of our continued research. As the system is affected by the complex external environment and different operating conditions, we can consider mixing other optimization algorithms for parameter optimization in order to improve the prediction accuracy and stability of the system, so that it can be applied to other complex operating conditions of WTGs; and we can consider adding optimization objectives such as user service quality in the later stage, in order to investigate the performance of the algorithm and improve its performance. We can consider adding optimization objectives such as customer service quality to explore better performance optimization algorithms in the future.

Acknowledgements.

China University Innovation Fund -- New Generation Information Technology Innovation Key Project, No. 2022IT017; Education Reform Project of China Institute of Labor Electronics in 2022, No. CIEL2022060; Cooperative Education Program of Ministry of Education, No.220900287260151, No. 202102602013; Key Project of Graduate Education and Teaching Research and Reform of Tianjin Agricultural University, No. 2021-YA-6; Talent Support Program of Tianjin Agricultural University, No. Y0400907; School-enterprise cooperation technology researched development project (TNHXKJ2022074)

References

[1] Li Yao, Zhu Caichao, Tao Youchuan, et al. Research Status and Development Tendency of Wind Turbine Reliability[J]. China Mechanical Engineering, 2017, 28(09): 1125-1133.

[2] Dong Yuting, Wang Haiyun, Tang Xinan. Status of Wind Turbine Condition Monitoring System[J]. Electric Machines & Control Application, 2013, 40(01): 17-21.

[3] Cheng Jiangzhou, Feng Xinyi, Feng Mengting, et al. Wind Turbine Fault Diagnosis and Risk Prediction Considering Meteorological Factors[J]. Science Technology and Engineering, 2022, 22(22): 9645-9615.

[4] Zhang Jinhua, Wu Wenjing, Wang Zhuoran, et al. RESEARCH OF WIND FARM OPTIMAL SCHE-DULING BASED ON REDUCING WIND FARM LOSSES[J]. Acta Energiae Solaris Sinica, 2018, 39(04): 1085-1096.

[5] Song Wei, Lin Jianwei, Zhou Fangze, et al. Wind turbine bearing fault diagnosis method based on an improved denoising AutoEncoder[J]. Power System Protection and Control, 2022, 50(10): 61-68.

[6] Han Pingping, Zhang Xiangmin, Ding Ming, et al. Application of Data Storage and Analysis Technology of Hadoop to Wind Power Grid-connected System[J]. Proceedings of the CSU-EPSA, 2018, 30(01): 43-50.

[7] Wang Lintong, Zhao Teng, Zhang Yan, et al. Storage optimization and parallel query method for big data of wind power monitoring based on Hadoop[J]. Electrical Measurement & Instrumentation, 2018, 55(11): 1-6.

[8] Liu Shuren, Song Yaqi, Zhu Yongli, et al. Research on Data Storage for Smart Grid Condition Monitoring Using Hadoop[J]. Computer Science, 2013, 40(01): 81-84.

[9] Zhou Niancheng, Liao Jianquan, Wang Qianggang, et al. Analysis and Prospect of Deep Learning Application in Smart Grid[J]. Automation of Electric Power Systems, 2019, 43(04): 180-191.

[10] Cuervo E, Balasubramanian A, Cho DK, et al. MAUI: Making smartphones last longer with code offload[C]// International Conference on Mobile Systems. DBLP, 2010.

[11] Chae D, Kim J, Kim J, et al. CMcloud: Cloud Platform for Cost-Effective Offloading of Mobile Applications[C]// 2014 14th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (GGGrid). ACM, 2014.

[12] Chen Xiao, Liu Wei, Chen Jing, et al. Research on Computing Offload Strategy in Edge Computing Environment[J]. Fire Control & Command Control, 2022, 47(01): 7-14+9.

[13] Li Qiang, Du Tingting, Tong Zhao, et al. Dependent Task Offload Based on Deep Reinforcement Learning in Mobile Edge Computing[J]. Journal of Chinese Computer Systems, 2023, 44(07): 1463-1469.

[14] Zhang Bingjie, Yang Yanhong, Cao Shaozhong. Review of Computing Offloading Schemes for Multi-access Edge Computing[J]. Journal of Frontiers of Computer Science and Technology, 2023, 17(09): 2030-2046.

[15] Wu Jun, Chen Zuoyi, Yan Zhe, et al. Development of Ship Bearing Health Condition Monitoring System Based on Hadoop[J]. Journal of Ordnance Equipment Engineering, 2020, 41(01): 140-144.

[16] Zeng Feng, Zhang Zheng, Chen Zhigang. Computation offloading and resource allocation strategy based on deep reinforcement learning[J]. Journal on Communications, 2023, 44(07): 124-135.

[17] Han Xiaofei, Song Qingyun, Han Ruiyan, et al. Survey on Mobile Edge Computing Offloading Technology[J]. Telecommunication Engineering, 2022, 62(09): 1368-1376.