

Design and Development of Wind Power Refined Simulation Software

Xuanhui Zhao^{1, a}

¹School of Electrical Engineering, Shandong University, Jinan 250061, Shandong Province, China;
^a 18340018687@163.com,

Abstract. A challenge to the steady operation of the system as well as the design, operation, and maintenance of wind power is posed by the wind power industry's rapid development and the rate at which wind power is incorporated into the power system. This work creates high-quality simulation software for wind power to address the complicated modeling requirements brought on by significant wind power penetration. The software implements the functions of fine modeling, offline simulation, real-time simulation, and data analysis and uses Python, Shell, and C++ language programming as its primary programming languages. Additionally, to confirm the software's functionality, this study employs OpenFAST single-unit and three-unit wind field samples. The reality demonstrates how easily real-time simulation software can be implemented and how well it does so. Software also has a cheap time and economic cost, and its application value is considerable.

Keywords: Software development; Wind power simulation; Real-time simulation; Data analysis.

1. Introduction

Wind power and solar energy have steadily come to the attention of all nations due to the resource issue. The installed capacity of wind power is increasing quickly due to the development of wind power technology. The total installed capacity of wind power reach 923 GW by the end of 2022 worldwide, while the installed capacity of wind power in China reach 395.6 GW[1].

However, the increased use of wind energy has brought up several problems. Large-scale wind power grid connections have the potential to oscillate the electricity grid, obstruct its capacity to run gradually and break down protection measures. Large-capacity wind power equipment, in particular far-sea wind turbines, has a significant maintenance cost and necessitates challenging maintenance decisions. It is difficult to rapidly switch new wind farms and turbines to outdated control methods, and choosing and validating new control procedures is challenging[2]. Sophisticated simulation tools are needed to mimic renewable energy equipment, and meet market demand.

Currently, there is a variety of software available for wind power simulation, including PSASP[3], PSCAD[4], Power Factory/DIGSILENT[5], GH Bladed[6], OpenFAST[7][8], HAWC2[9], and RSCAD[10]. Power Factory/DIGSILENT, PSASP, and PSCAD offer a range of component models for ideal power grid simulation; nonetheless, it is challenging to develop precise models of pneumatic, mechanical modules to achieve fine turbine simulation. In terms of aerodynamics and mechanics, GH Bladed, OpenFAST, and HAWC2 with excellent simulation models have been ideal. However, the electromagnetic part adopts a simplified model and is difficult to expand. The more advanced RTDS program, RSCAD, has a vast component library and can construct electromagnetic part models of turbines for real-time simulation, but it is more challenging to build precise pneumatic and mechanical modules. In conclusion, there are a few issues with the fine simulation of wind power in the current simulation software. A cost-effective and practical fine simulation software is urgently required.

This research creates a new sort of wind power fine simulation software as a result of the aforementioned circumstance. To achieve the fine model building of wind turbines, the program incorporates the incoming wind, pneumatic, mechanical, and electromagnetic modules. The precise model construction of the wind farm is made possible by the incorporation of the wake module.

The software integrates an offline simulation package to realize offline simulation. It can interact with the hardware simulators in real-time for real-time simulation. The data analysis module also

offers a range of data analysis techniques. The software can simplify the simulation operation steps and improve the efficiency of modeling, simulation, and data mining.

This paper is divided into four sections: an introduction, a description of the software architecture and function realization, a simulation of the software function, and a summary and prospect.

2. Main Functions of Simulation Software

The simulation software primarily performs four tasks: precise modeling, offline simulation, real-time simulation, and data analysis. This section will go into detail about each function.

2.1 Wind Power Fine Modeling

The wind farm is a multi-coupled complex system with rapid and slow time scales that include electromagnetic transient on fast time scales and the wind field, aerodynamics, and mechanical dynamics on slow time scales[10]. It is challenging to develop a refined model while the current software tends to simplify the model. The development of precise models for inlet wind, pneumatic, mechanical, and electromagnetic systems is made possible by the simulation software.

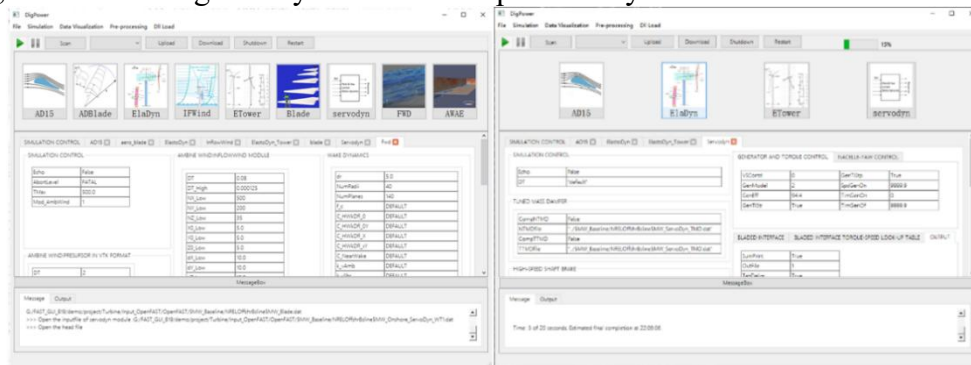


Fig. 1 Model construction interface and offline simulation

The program uses a variety of sub-interfaces to display, alter, and store parameters due to the numerous parameters and complicated time scale of the inflow wind, pneumatic, and mechanical modules. Users can directly build a new project, or change the existing model by selecting Read Template Project from the menu bar. Figure 1 depicts the interface for reading parameters.

2.2 Off-line Simulation

The fundamental use for simulation software is offline simulation, which can be used for design verification. The software incorporates an offline simulation program and pre-sets engineering templates for wind farms and turbines. Users can generate the required parameters by the external program, select the project, and conduct simulations.

Figure 1 depicts the offline simulation execution interface. The simulation's progress is displayed in the upper right corner, and the progress bar automatically disappears once the simulation is complete. The lowest portion of the interface is an information output box.

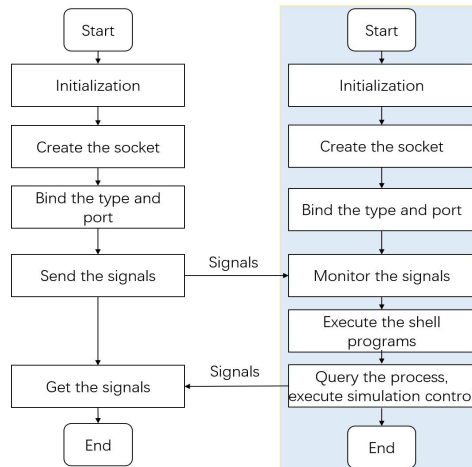


Fig. 2 Real-time simulation structure and process

2.3 Real-Time Simulation

The vital function of the software is real-time simulation, which may be used to make immediate forecasts and adjustments to power grids and wind farms. To provide a passable simulation of wind farms, the software can connect numerous hardware real-time simulators[12].

The simulation procedure is displayed in Figure 2. Start simulators and simulation software, then click Scan to get the simulators' IP to establish connections. The project is built on software (the turbines and the wind field are changed by drop-down box) and uploaded to the simulators. After the simulation starts, the user can control it on software (sending commands to simulators by UDP communication). After the simulation, the software can obtain the results and analyze them.

2.4 Data Analysis

Data analysis is one of the often-used features. Users can integrate various result data, explore the characteristics, and expand the applications. The software's real-time image drawing, data comparison, graph drawing, data storage, and other features are set in the menu bar. Figure 3 depicts the simulation software's real-time picture interface and data analysis interface.

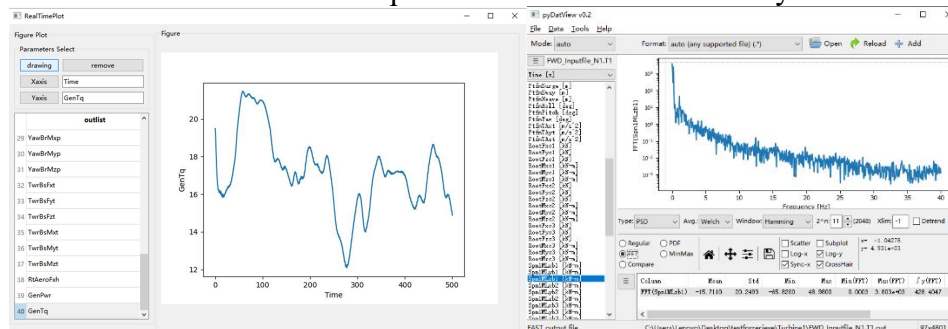


Fig. 3 Real-time image rendering and data analysis

3. The architecture and implementation of the Simulation software

The complexity of software development can be reduced, and secondary development can be facilitated, by a sound software architecture [13]. The four components of the software are software interfaces, project construction, simulation, and data analysis. The simulation module, which can be separated into offline simulation and real-time simulation, is the software's core. Figure 4 illustrates its precise structure. This section describes the implementation of the architecture and functionality.

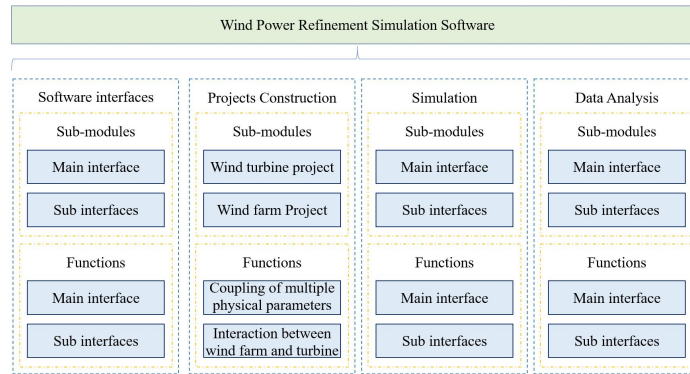


Fig. 4 The structure of software

3.1 Software Interfaces and Implementation

The interface is a window for human-computer interaction. The QT framework is used to create the simulation software. QT is a cross-platform, object-oriented, open-source C++ framework for creating graphical user interfaces. It offers a wide range of UI components and sophisticated APIs, which can make development easier. The software includes a functional interface and numerous sub-interfaces. The functional interface serves as the foundation for function implementation, while the sub-interfaces are used to generate each module's parameters and to analyze and observe data.

The software function interface is seen in Figure 1 and is made up of the menu bar, function area, module area, parameter area, and information display area, in that order. The buttons in the menu bar and function area can be used to perform operations such as simulator scanning, project delivery, simulation control, etc. It is possible to switch the 16 sub-interfaces by buttons in the module area. Above the parameter area is the label bar. Below, you can see and change the module's parameters. Program output data is shown in the information display area.

3.2 Project Construction and Implementation

Software activities are made simpler by normalizing project files. The program creates template projects like wind farms and windmills. Wind farm engineering covers wind turbine parameters, wind farm wake, and other parameter files. Wind turbine engineering contains several parameter files of multiple modules, such as pneumatic and mechanical. The project may be finished by users using the software's read, modify, save, and issue features. The simulation project comes with a unique Info.prj file that details each project's type and the paths to the necessary input files. Figure 12 depicts the wind farm project's file organization.

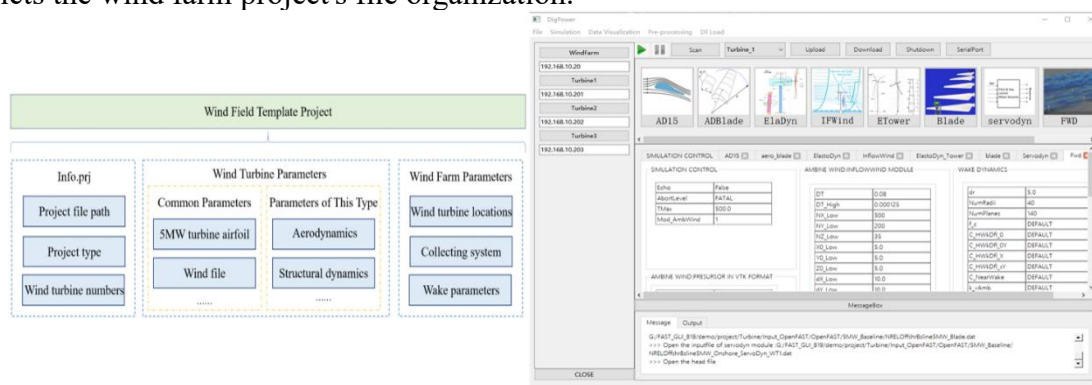


Fig. 5 Wind farm project structure and reading

By reading the Info.prj file, the simulation program may read the project data, and the parameters of each module and show them in the interface for changing and saving.

3.3 Off-line Simulation and Implementation

The offline simulation program is the result of the secondary development of the OpenFAST program. The refined electromagnetic model is used instead of the simplified model to realize the refined off-line simulation. In the software, the turbine and the wind field are switched by the drop-down box. After the project is established, the offline simulation package is called for simulation.

3.4 Real-time Simulation and Implementation

The simulation software is deployed in the upper computer of the Windows system, and the upper computer is connected to the hardware real-time simulator through the switch. The software interacts with the simulator through UDP communication protocol and controls the real-time simulation of the simulator. The real-time simulation process mainly includes simulator scanning, project construction, and simulation control, which are detailed in this section.

3.4.1 Hardware Simulator Scan

Before starting a real-time simulation, the software must get the simulator's IP address to connect. UDP communication has strong real-time performance, small resource requirements, and high reliability in the local area network[14]. Therefore, this paper adopts UDP broadcast communication to obtain the IP address of the hardware simulators. Click the software Scan button to set the port to initiate listening. The simulator side program broadcasts its type and IP when it starts. The software obtains the address and establishes a connection for control.

3.4.2 Project Creation and Delivery

After the software obtains the simulator IP, the user can import a simulation template project to build wind turbine and wind farm models. The software first reads the Info.prj file of the project to obtain the relative path of the parameter files, read them, and display them. Then the user can make changes and save, as shown in Figure 5. The project can be selected by the Upload button and sent to the simulators over an SSH connection. The user clicks the start button and sends commands to the simulators through UDP. All types of simulators are connected and co-simulated.

3.4.3 Simulation Control

The software interacts with the simulator service program to realize the start and stop control of the real-time simulation. In the process of real-time simulation, you can click the software pause/continue button for simulation control. The software sets the socket, sends the signal to each simulator through UDP protocol, and the service program receives the signal for start and stop control[15]. After the simulation, the output file is stored in a specific path, and the simulation software can connect to the simulator to download it to the upper computer for data analysis and visualization.

3.5 Data Analysis and Implementation

The data analysis function mainly includes two parts: real-time image and data analysis. When simulating the output data, you can select the real-time plot button and select the result file for image rendering. The software obtains the data label information and generates the parameter bar in the sub-interface. Users can select different horizontal and vertical coordinate parameters to draw and observe dynamic images.

In addition, to further improve the ability of data mining, the simulation software integrates the PyDatview program, which can realize multiple functions of data analysis, including multi-file reading, multi-graph comparison, subgraph drawing, PDF/FFT data analysis, image file saving, and so on.

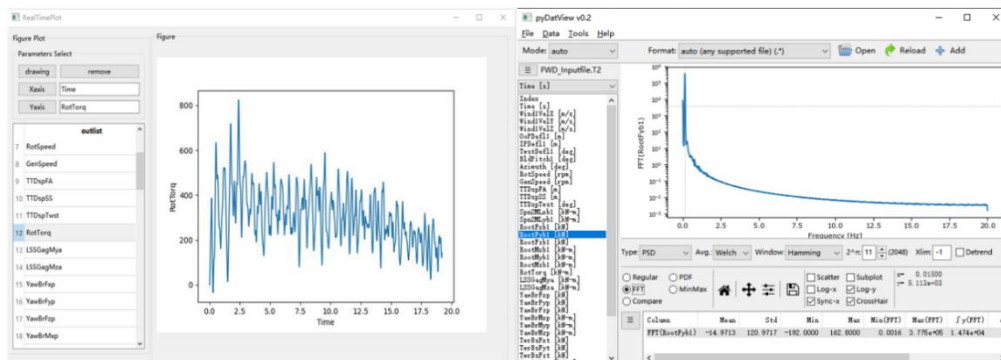


Fig. 6 Real-time image rendering and data analysis

4. Example verification

The software functions mentioned above are verified by using OpenFAST's wind turbine and FAST's three-turbine wind field [7][8].

4.1 Off-line Simulation

Aiming at the software offline simulation function, this paper uses a 5MW onshore wind turbine to verify the case. The software reads the parameter file and saves the changes to perform offline simulation. The parameter setting and execution process are shown in Figure 1. Select the output file and select the horizontal and vertical parameters to generate a live image, as shown in Figure 6.

4.2 Real-Time Simulation

Aiming at the real-time simulation function of the simulation software, this paper adopts the wind field calculation example of three wind turbines provided by FAST.Farm for verification. After the model is established, the software sends instructions to control the hardware simulator for simulation. After the simulation, the output is obtained, including one wind field file and several fan files. The data analysis module can analyze the obtained output results, input multiple files, and draw and save multiple graphs, as shown in Figure 6.

5. Summary and Prospect

This paper introduces in detail the main functions, framework, and implementation of the new wind power fine simulation software. Simulation software based on QT development, with Python, C++, and Shell smooth implementation of wind power fine modeling, offline simulation, real-time simulation data analysis, and other functions. The simulation software has a clear structure, easy operation, good expansibility, and simplified simulation steps. However, the software also has some shortcomings, such as real-time simulation data communication problems. This paper plans to use UDP communication in LAN to design a perfect communication module embedded in the upper computer system.

References

- [1] Global Wind Energy Council. Global wind report-annual market update 2022[R]. Global Wind Energy Council, 2023.
- [2] Zhang Liying, Ye Tinglu, Xin Yaozhong, Han Feng, Fan Gaofeng. Problems and Measures of Power Grid Accommodating Large Scale Wind Power. Proceedings of the CSEE, 2010,30(25):1-9.
- [3] LI Yalou, Li Fang, Liu Hechuan et al.. Framework Design of Simulation System for Integrated Energy Systems Based on PSASP. Power System Technology, 2019,43(07):2498-2506.

- [4] Zhang Hua, Xie Kaigui. PSCAD Based Simulation and Analysis on PV Power Station. *Power System Technology*, 2014,38(07):1848-1852.
- [5] ZHAO Haoran, Wu Qiuwei, Ioannis MARGARIS et al. Implementation of IEC Generic Models Type 1 Wind Turbine Generator in DIgSILENT PowerFactory [J]. *Automation of Electric Power Systems*,2013,37(08):26-33.
- [6] E.A.Bossanyi. GH Bladed user manual. German: GH&Partners Ltd,2009.
- [7] Jonkman J, Jr M. FAST user's guide. Colorado: National Renewable Energy Laboratory, Golden, CO,2005.
- [8] Shaler K , Jonkman J . FAST.Farm development and validation of structural load prediction against large eddy simulations. *Wind Energy*.
- [9] Larsen T, Madsen A. How 2 HAWC2, the user's manual. Risø-R-1597(ver. 4-3)(EN). Roskilde, Denmark: Technical University of Denmark, 2012.
- [10] Jia Feng, CAI Xu, Li Zheng et al. Refined Modeling of Wind Energy Conversion Systems and Real-time Co-simulation With Hardware-in-loop. *Proceedings of the CSEE*, 2017,37(04):1239-1251.
- [11] Li Bing. Research on Refined Real-time Simulation Technology of Wind Farm. Shandong University, 2021.
- [12] B. Li, H. Zhao, S. Gao, S. Hu. Digital real-time co-simulation platform of refined wind energy conversion system[J]. *International Journal of Electrical Power & Energy Systems*, 117, 105676,2020.
- [13] Li Wenfan, Liu Zhigang, Wu Wencheng et al. Design of Power System Geographic Wiring Diagram Drawing Software Based on Qt [J]. *Automation of Electric Power Systems*,2013,37(07):72-76+107.
- [14] B. Eckart, X. He, Q. Wu and C. Xie. A Dynamic Performance-Based Flow Control Method for High-Speed Data Transfer. *IEEE Transactions on Parallel and Distributed Systems*, 2010,21(1): 114-125.
- [15] Ye Zhenfeng, Song Jinhai, Li Qiuhua et al. Application of Embedded Linux in Real-time Sampling Unit of Digital Substation [J]. *Automation of Electric Power Systems*,2010,34(10):109-112.