Enhancing Grid Resilience through Real-time Self-Healing: Integrating Python-PowerFactory-IoT with ESP32

Agus Suwardono
Electrical Engineering Department
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
7022232006@student.its.ac.id

Ardyono Priyadi
Electrical Engineering Department
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
priyadi@ee.its.ac.id

Margo Pujiantara
Electrical Engineering Department
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
margo@ee.its.ac.id

Firas Quthbi Sidqi
Electrical Engineering Department
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
Ibhtuqfiras0999@gmail.com

Sjamsul Anam
Electrical Engineering Department
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
anam@ee.its.ac.id

Vita Lystianingrum
Electrical Engineering Department
Institut Teknologi Sepuluh Nopember
Surabaya, Indonesia
vita@ee.its.ac.id

2024 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM) | 979-8-3503-6880-2/24/$31.00 ©2024 IEEE | DOI: 10.1109/CENIM64038.2024.10882635

***Abstract* — The modern power grid demands innovative solutions to improve its resilience and reliability. This research aims to develop a novel approach to real-time self-healing in power distribution networks by integrating Internet of Things (IoT) technology with advanced simulation tools, specifically DIgSILENT PowerFactory. The proposed system utilizes ESP32 microcontrollers equipped with sensors to measure electrical parameters such as voltage and current, collecting real-time data from the network. This data is transmitted via Wi-Fi to a MySQL database on a central server. Python (Jupiter Notebook) scripts facilitate the connection between the database and DIgSILENT PowerFactory, enabling real-time load flow simulations within the network. The results of this system show significant improvements in fault response times and overall grid resilience. Our implementation demonstrates that this integrated approach can effectively reduce downtime and enhance the stability of power distribution networks.**

Keywords—IoT, ESP32, DIgSILENT PowerFactory, real-time self-healing, resilience

I. INTRODUCTION

Power systems are typically engineered to handle frequent, low-impact outages with minimal disruption. However, rare but severe events like extreme weather or natural disasters, while less likely, can have devastating consequences. These events can lead to prolonged outages and significant damage to key infrastructure, such as substations, transmission lines, and power generation facilities.[1]. In recent years, extensive power outages resulting from extreme disasters have been witnessed on both national and international scales. These HILP events can lead to unpredictable losses once they occur [2]. Taking a strong typhoon as an example [3], it typically results in the breakage of distribution lines and the collapse of utility poles, causing the interruption of power supply to critical loads. Consequently, this affects the normal operation of essential public functions such as municipal services, healthcare, and transportation.

The reliability and resilience of power distribution networks are critical for ensuring uninterrupted electricity supply, which is fundamental to modern society's functioning. Distribution network reconfiguration (DNR) involves altering the distribution network's structure by controlling the operation of switches [4]. Traditional power systems face challenges such as aging infrastructure, increasing demand, and vulnerability to faults and disruptions. To address these issues, integrating Internet of Things (IoT) technology with advanced simulation tools offers a promising solution. The IoT enables real-time monitoring and control, while simulation tools like DIgSILENT PowerFactory provide detailed analysis and optimization capabilities.

Despite the advancements in power system management, real-time fault detection and self-healing remain significant challenges. Current methods often involve manual intervention and are time-consuming, leading to prolonged outages and reduced system reliability. Additionally, the integration of IoT with existing power systems is hindered by issues related to data management, real-time processing, and communication reliability. The need for a cost-effective, scalable, and efficient solution to enhance the resilience of power distribution networks is evident [5].

This research aims to develop a real-time self-healing system for power distribution networks by integrating IoT technology using ESP32 microcontrollers with advanced simulation tools. The specific objectives are an IoT-based framework for real-time data collection from the power network using ESP32 microcontrollers. Implement a robust data transmission and management system using a MySQL database and Jupiter Notebook for data analysis. Integrate the collected data with DIgSILENT PowerFactory for real-time fault detection, isolation, and restoration. Evaluate the system's performance in enhancing fault response times and overall grid resilience[6].

This research contributes to the field of smart grid technology by providing a novel integration of IoT and power system simulation tools for real-time self-healing. The use of ESP32 microcontrollers offers a cost-effective solution for continuous monitoring and data collection. The integration with Jupiter Notebook and DIgSILENT PowerFactory allows for sophisticated data analysis and real-time simulation, leading to improved fault detection and restoration processes. This approach not only enhances the reliability of power distribution networks but also paves the way for more advanced smart grid applications [7].

The proposed system offers several benefits, including the enhanced Reliability to Real-time monitoring and rapid fault response improve the overall reliability of the power distribution network. Reduced Downtime: The self-healing capability ensures quick isolation of faults and restoration of service, minimizing outages. Cost-Effectiveness: Using ESP32 microcontrollers provides a low-cost solution for extensive network monitoring. Scalability: The system can be easily scaled to accommodate growing power network demands and complexities. Data-Driven Decision Making: Advanced analytics and simulation enable informed decision-making and optimized power system operations. By addressing the challenges of real-time fault detection and self-healing, this research contributes to the development of more resilient and intelligent power distribution networks, ensuring a stable and reliable electricity supply for the future.

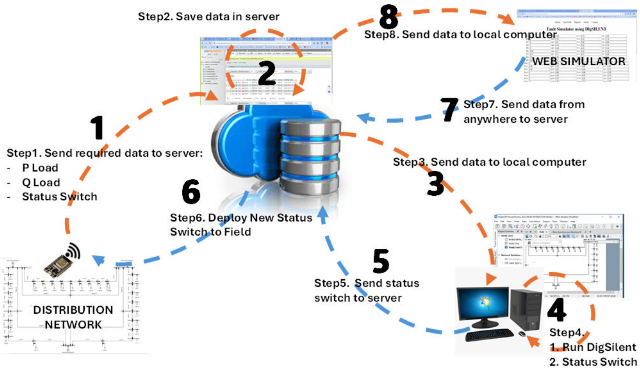

Fig. 1. The proposed system architecture

II. Smart Distribution Network and Self-Healing Control

A. Smart Distribution Network

The smart distribution network, enabled by distribution automation, incorporates sophisticated measurement and communication technologies for system-wide supervision and remote control. It also integrates distributed generators, energy storage systems, and electric vehicles. Unlike traditional distribution networks, the smart network supports a large number of distributed energy resources and enables interaction with the network by end users to facilitate smart electricity consumption. Additionally, it incorporates self-healing and optimization control, ensuring higher energy efficiency and security—key characteristics of a smart distribution network. Furthermore, it offers intelligent monitoring of the network and devices, leading to improved asset management and greater operational efficiency[8].

B. Self-Healing Control

Self-healing is a core function and crucial technology of the smart distribution network, playing a significant role in increasing the dependability and efficiency of the system. It relies on computer and communication technologies to enable proactive monitoring of the system's operational state, effectively identifying internal and external threats, and mitigating the impact of disturbances on network operations. This is achieved through the full coordination of various distributed power sources and grid operating conditions, ultimately improving system security and efficiency. Depending on the operational state, self-healing contains precautionary self-healing under abnormal conditions, fault self-healing during fault situations, and affordable self-healing. during normal operations.

Preventive self-healing control involves adjusting the system's operational status to minimize the impact of potential threats to the distribution network by predicting near-future operational conditions and risks, both internal and external. Fault self-healing control focuses on quickly diagnosing and locating faults using terminals and the control center to achieve fault isolation and automatic load transfer in unaffected areas, minimizing disruptions to the network. In extreme situations, the network can utilize distributed power supply capabilities to implement controlled islanding, initiate black-start procedures, and ensure power supply to critical loads.

The economical self-healing control is applied when the distribution network is in a safe and stable operation state, and the economical reconfiguration control strategy is adopted to improve the operation efficiency and to reduce the loss of the distribution network.

III. Experimental Setup

This paper, the proposed scheme is depict in Figure 1. The network simulate load flow analysis on 30-bus. DIgSILENT PowerFactory software, one of the most powerful and widely used power system analysis software in electrical engineering was used for the model simulation. The 30-bus system for schemes relating power generations, distribution lines, and loads.

The diagram illustrates a data acquisition and transmission system using an ESP32 microcontroller. The ESP32 is connected to sensors that measure current (I), voltage (V), power (P), and reactive power (Q). The sensor data is transmitted wirelessly to a router, which then connects to a server at domain.com. On the server, a PHP script named post-data.php receives the transmitted data and processes it. In this research transmission data from ESP32 through HTTP Post to post-data.php are already secured by API Key to protect cyber attack. The next research author will propose encryption technic to increase the trustworthiness of the proposed system. The processed data is then stored in a MySQL database for further analysis or visualization. Sensors read values from the network using an ESP32, and then this microcontroller transmits those values to a server. Subsequently, the data is captured by a PHP script and stored in a database Figure 1. In this step, Arduino IDE is used to program ESP32 to obtain the current, voltage, power data from network via Wi-Fi to a MySQL.

Fig. 2. Reading sensor

Data is first stored in a MySQL database as Figure 2. This data is then retrieved and processed using Jupiter Notebook, a Python-based environment for data science and machine learning. Jupiter Notebook acts as a bridge between the MySQL database and Digsilent PowerFactory PC, a software used for power system analysis and simulation. The processed data is used as input in Digsilent PowerFactory PC to perform load flow simulations based on a specified power system scheme. The simulation results, such as optimal reconfiguration strategies, are then sent back to Jupiter Notebook and used to update variables or data in the MySQL database. This workflow allows for automated data analysis, simulation, and optimization of power systems as Figure 3

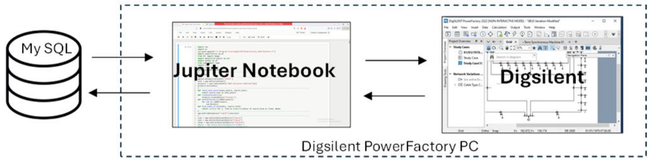

Fig. 3. Data analysis and simulation workflow

The ESP32 sends an HTTP GET request to a server at domain.com, requesting data from a PHP script named get-data.php like Figure 4. This script retrieves data from a MySQL database and sends it back to the ESP32 in JSON format. The ESP32 then parses the JSON data to extract relevant variables. These variables are used to control switches, turning them on or off based on the values stored in the database. This system allows for remote monitoring and control of devices connected to the switches.

Fig. 4. Deploying new status switch

The diagram illustrates a data retrieval and control system using an ESP32 microcontroller. The ESP32 periodically sends an HTTP GET request to a server at domain.com, requesting updated data from a PHP script named get-data.php. This script retrieves the latest data from a MySQL database, including the current status of the switches, and sends it back to the ESP32 in JSON format.

The ESP32, programmed using the Arduino IDE, parses the JSON data to extract the relevant switch status information. This information is then used to update the control signals for the switches, effectively implementing the desired changes to the power system network. By regularly fetching and processing the updated data, the ESP32 ensures that the switches are always operating in accordance with the latest instructions stored in the MySQL database.

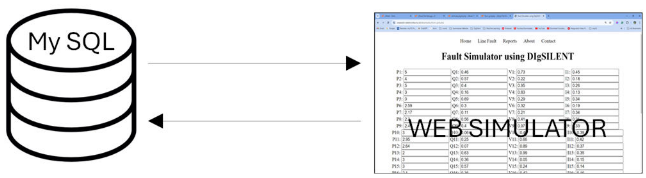

Fig. 5. Web simulator

The diagram illustrates a data flow between a MySQL database and a web-based simulator.

A. MySQL Database:

- Serves as the central repository for storing data.
- This data could include various parameters related to a power system, such as voltage levels, current readings, and switch statuses.

B. Web Simulator:

- Acts as a user interface, providing a visual representation of the data stored in the MySQL database.
- The simulator is likely built using DigSilent, a power system analysis software.
- It displays real-time data from the system, including:
    - Current values
    - Voltage levels
    - Active and reactive power
    - Switch statuses
- The simulator can also be used to simulate different fault conditions or scenarios, allowing for testing and analysis of the power system's behaviour under various circumstances.

C. Data Flow:

- Data is continuously transferred from the MySQL database to the web simulator.
- This ensures that the simulator always displays the most up-to-date information about the power system.
- The web simulator can also send data back to the MySQL database, for example, to update the status of a switch based on user input as Figure 5.

D. Overall Function:

- The combined system provides a real-time monitoring and analysis tool for a power system.
- The MySQL database stores the raw data, while the web simulator offers a user-friendly interface for visualizing and interacting with this data.
- This setup is particularly useful for:
    - Monitoring the health of a power system
    - Identifying potential issues or anomalies
    - Testing different control strategies
    - Training operators

In essence, the MySQL database acts as the data source, while the web simulator serves as a dynamic display and control panel for the power system.

IV. METHODOLOGY

The optimization problem arises from the switching processes used to change the network topology are referred to as the network reconfiguration issue. The primary objective is to determine the optimal operational strategy[9]. In figure 6 bellow, the proposed methodology will be applied.

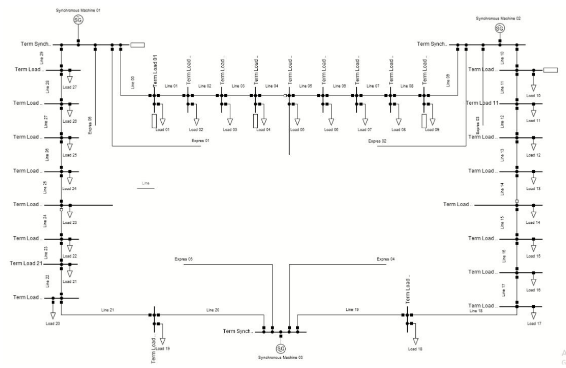

Fig. 6. Proposed Model

Solving this problem efficiently requires selecting the most suitable operational topology from all available configurations. Nevertheless, given the exponential growth in configurations that occurs with the number of switches and their arrangement, it is not feasible to inspect every possible configuration for modern distribution systems. This rapid increase is referred to as a combinatorial explosion [10].

This paper presents a lookup table algorithm to solve the distribution network reconfiguration problem by determining the optimal solution under various constraints. The optimal outcome guides the decision for the new network configuration. The objective is to minimize downtime while maintaining voltage quality and generation capacity, thereby improving the voltage profile for customers and increasing reliability levels.

A. Objective functions

Various aspects are taken into account when defining the objective function for the network reconfiguration problem. One of considered factors is minimize generator load that the objective functions is defined as equation 1, while another key aspect, often explored by researchers, is maximizing reliability indexes.

$$f(x) = P_G - \sum_{i=0}^{n} P_L \tag{1}$$

B. Constraints

A common limitation in network reconfiguration is the current limit of electrical components such as conductors, transformers, protection, and switching devices. Another important and frequently regarded restriction is the voltage limit, which is closely related to power quality. [11]. The implementation of constrain will show up in equation 2 -7 in this paper.

Considering the previous points, constraints are defined as follows.

a) *Load Power should be less than the available power from the generators:*

$$\sum_{i=0}^{n} P_L \leq P_G \tag{2}$$

This ensures that the total demand of the system does not exceed the total power generation capacity.

b) *Line loading should be less than 100% of the rated capacity:*

For each transmission line i→j,

$$S_{ij} \leq S_{ij,max} \tag{3}$$

This constraint ensures that none of the lines exceed their thermal or capacity limits.

c) *Bus voltages should stay within acceptable limits around 1 pu:*

The voltage at each bus should ideally be 1 pu, but practical constraints require a tolerance. Let's assume the acceptable voltage range is between 0.95 pu and 1.05 pu:

$$0.95 \leq V_i \leq 1.05 \tag{4}$$

This ensures that the voltage at each bus remains within a small deviation from the nominal value.

d) *Radial network with express line.*

e) *Each element's current magnitude needs to stay within the permitted bounds:*

$$|I_i| \leq I_{i\,max} \tag{5}$$

f) *Each protective device's current magnitude must remain within permitted bounds..*

$$|I_i| \leq I_{jprot} \tag{6}$$

g) *The voltage magnitude at each node must remain within its acceptable range.*

$$V_{j\,min} \leq V_j \leq V_{j\,max} \tag{7}$$

Where:
P_L - Total Load Power (MW)
P_G - Total Power Generated by Generators (MW)
S_{ij} - Power flow on line i→j
$S_{ij,max}$ - Maximum power flow capacity of line i→j
I_i - current bus i,
$I_{i\,max}$ - max current accepted bus i,
I_{jprot} - current limit threshold of the protection device j,
V_j - voltage magnitude at bus j,
$V_{j\,min}$ - allowable voltage min magnitude at bus j,
$V_{j\,mx}$ - allowable voltage max magnitude at bus j

C. Heuristic Strategies

A stepwise decrement approach is suggested in [12] for the NP-hard problem of the updated plan to add new lines to the power network. In this work, we propose an updated lookup table method embedded in Python, based on network topology analysis.

V. EXPERIMENT

To determine the effectiveness of the proposed lookup table method for active distribution networks considering interval power flow, comparative testing experiments were conducted. In this paper, we use the term cycle to show process data from the reading sensor until deploying the status switch. Table 1 shows the normal services in network, before the fault happens. Every bar shows a radial connection. Network reconfiguration add another generator load until the constraints are met.

Table 1. Normal Configuration

Gen 01	L_00	L_01	L_02	L_03	L_04	L_25	L_26	L_27	L_28	L_29
Gen 02	L_05	L_06	L_07	L_08	L_09	L_10	L_11	L_12	L_13	L_14
Gen 03	L_15	L_16	L_17	L_18	L_19	L_20	L_21	L_22	L_23	L_24

Where : L_xx is notation for line xx

When a fault happens in line 03, the system will have alternate choices to reconfiguration the network. The alternate choice happens in Python software as an input DigSilent Powerfactory in Table 2. In this case, python calculates the reconfiguration only one line and then matches that load after line addition in power generator 2 still under maximum capacity. The remaining generator - gen 03 - is on the same loads.

Table 2. Network reconfiguration fault on line 03

Gen 01	L_00	L_01	L_02	L_03	L_25	L_26	L_27	L_28	L_29		
Gen 02	L_04	L_05	L_06	L_07	L_08	L_09	L_10	L_11	L_12	L_13	L_14

The other case fault is in line 01. The possibilities of a solution are depicted in Table 3. Line Express 01 is involved when generator 02 capacity has already reached the maximum. Line Express 01 replaces line 01.

Table 3. Network reconfiguration fault on line 01

Gen 01	L_00	L_25	L_26	L_27	L_28	L_29							
Gen 02	L_04	L_05	L_06	L_07	L_08	L_09	L_10	L_11	L_12	L_13	L_14		
Gen 02	L_03	L_04	L_05	L_06	L_07	L_08	L_09	L_10	L_11	L_12	L_13	L_14	
Gen 02	L_02	L_03	L_04	L_05	L_06	L_07	L_08	L_09	L_10	L_11	L_12	L_13	L_14
Gen 01	L_00	L_Exp 01	L_25	L_26	L_27	L_28	L_29						
Gen 01	L_00	L_Exp 01	L_02	L_25	L_26	L_27	L_28	L_29					
Gen 01	L_00	L_Exp 01	L_02	L_03	L_25	L_26	L_27	L_28	L_29				

From the simulation above, we note time from beginning to ending. From reading data by sensors, until deploy status switches. Figure 7 shows the duration of every cycle in range between 36 – 40 seconds.

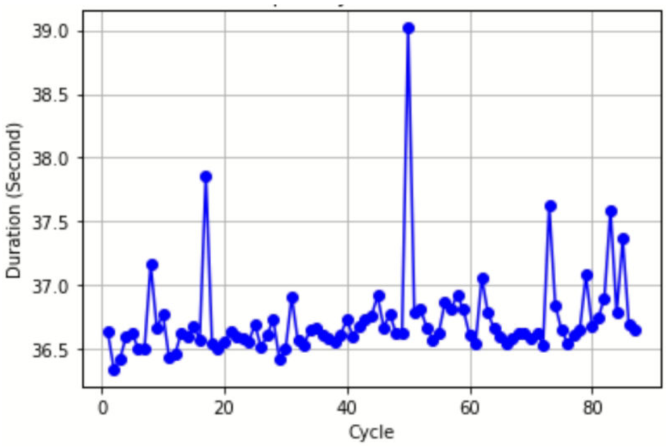

Fig. 7. Graphic Cycle Vs Duration

VI. CONCLUSION

The proposed system architecture can reduce the downtime after faults happen. This action can be done by network reconfiguration that needs less than one minute, entering the range interval between 36-40 seconds. This system can save the last load data before the outage. The last operational data after the outage provides consideration for decision-making during simulation to get the optimal network.

REFERENCES

[1] J.-H. Lin and Y.-K. Wu, "Review of power system resilience concept, assessment, and enhancement measures," *Appl. Sci.*, vol. 14, no. 4, p. 1428, 2024. [Online]. Available: https://doi.org/10.3390/app14041428.

[2] A. K. Erenoğlu, I. Sengor, and O. Erdinç, "Power system resiliency: A comprehensive overview from implementation aspects and innovative concepts," *Energy Nexus*, vol. 15, p. 100311, 2024. [Online]. Available: https://doi.org/10.1016/j.nexus.2024.100311

[3] Y. He, Y. You, Z. He, H. Lu, L. Chen, Y. Jiang, and H. Chen, "Study on the resilience model of power distribution networks under typhoon disasters," in *Proc. 4th Int. Conf. Power and Electr. Eng.*, J. Li, Ed., Singapore: Springer Nature Singapore, 2024, pp. 163–173

[4] Seungchan Joet al., "Self-healing radial distribution network reconfiguration based on deep reinforcement learning," Volume 22, June 2024, 102026 doi:10.1016/j.rineng.2024.102026

[5] Nand Kumar et al. "Self-Healing Networks AI-Based Approaches for Fault Detection and Recovery",2023 Vol. 47 No. 4 (2023), doi: 10.52783/pst.206

[6] Peter, G.; Stonier, A.A.; Gupta, P.; Gavilanes, D.; Vergara, M.M.; Lung sin, J. Smart Fault Monitoring and Normalizing of a Power Distribution System Using IoT. *Energies* **2022**, *15*, 8206. https://doi.org/10.3390/en15218206

[7] Dimara, A.; Vasilopoulos, V.-G.; Papaioannou, A.; Angelis, S.; Kotis, K.; Anagnostopoulos, C.-N.; Krinidis, S.; Ioannidis, D.; Tzovaras, D. Self-Healing of Semantically Interoperable Smart and Prescriptive Edge Devices in IoT. *Appl. Sci.* **2022**, *12*, 11650. https://doi.org/10.3390/app122211650

[8] W. Liu, T. Kang, W. Cheng and F. Zhao, "The modeling of self-healing control system for distribution network based on UML," *2015 5th International Conference on Electric Utility Deregulation and Restructuring and Power Technologies (DRPT)*, Changsha, China, 2015, pp. 1435-1439, doi: 10.1109/DRPT.2015.7432458.

[9] D. P. Bernardon *et al.*, "Automatic reconfiguration of distribution networks using Smart Grid concepts," *2012 10th IEEE/IAS International Conference on Industry Applications*, Fortaleza, Brazil, 2012, pp. 1-6, doi: 10.1109/INDUSCON.2012.6451381.

[10] H. P. Schmidt, N. Ida, N. Kagan and J.C. Guaraldo, "Fast reconfiguration of distribution systems considering loss minimization," IEEE Transactions on Power Systems, vol. 20, no. 3, pp. 1311-1319, Aug. 2005

[11] S. Mishra, D. Das, and S. Paul, "A comprehensive review on power distribution network reconfiguration," *Energy Systems*, vol. 8, no. 2, pp. 227-284, 2017, doi: 10.1007/s12667-016-0195-7.

[12] D. Bienstock and A. Verma, 'Strong NP-hardness of AC power flows feasibility', *Operations Research Letters*, vol. 47, no. 6, pp. 494–501, 2019.