Faculty of Engineering

A Framework for Green Energy Resources Identification and Integration Supported by Real-Time Monitoring, Control, and Automation Applications

Jong Far Chen

Doctor of Philosophy
2025

A Framework for Green Energy Resources Identification and Integration Supported by Real-Time Monitoring, Control, and Automation Applications

Jong Far Chen

A thesis submitted

In fulfillment of the requirements for the degree of Doctor of Philosophy

(Electrical and Electronic Engineering)

Faculty of Engineering
UNIVERSITI MALAYSIA SARAWAK
2025

DECLARATION

I declare that the work in this thesis was carried out in accordance with the regulations of Universiti Malaysia Sarawak. Except where due acknowledgements have been made, the work is that of the author alone. The thesis has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

………………………………………

Signature

Name: Jong Far Chen

Matric No.: 22010314

Faculty of Engineering

Universiti Malaysia Sarawak

Date : 06/05/2025

ACKNOWLEDGEMENT

First and foremost, I would like to express my sincere gratitude to Universiti Malaysia Sarawak for granting me an opportunity to pursue my postgraduate studies in this esteemed campus. My great appreciation also goes to my supervisor, Prof. Musse Mohamud Ahmed, for his exceptional supervision, encouragement, and support in various ramifications throughout every phase of my research journey. I would also like to extend my heartfelt gratitude to my co-supervisor, Prof. Ir. Dr. Andrew Ragai Henry Rigit, for his invaluable guidance and support along the way.

Many thanks to Universiti Malaysia Sarawak for the grant under UNIMAS's Vice-Chancellor High Impact Research Fund (VCHIF), Grant No: UNI/F02/VC-HIRG/85514/P11-03/2022, which funded my research work. Special indebtedness towards Yayasan Sarawak for their financial support to cover my living expenses during the study period.

I also would like to deliver my heartfelt gratitude to my beloved family as the source of motivation, rendering their limitless caring and utmost encouragement in completing the research study. Last but not least, I wish to acknowledge my friends for exchanging ideas, kind assistance, and being willing to discuss problems together, contributing in many ways to accomplish the research work.

ABSTRACT

Sarawak is transitioning to green energy adoption, yet progress is hindered by a lack of comprehensive spatial data to identify optimal locations, inadequate optimization techniques for effective integration of these sites, and insufficient robust Industrial Internet of Things-based real-time monitoring and automation strategies to manage the intermittent nature of green energy resources. To address these challenges, a novel Geographical Information System-based fuzzy Technique for Order Preference by Similarity to Ideal Solution coupled with filtration algorithms was proposed. This two-layered approach effectively filters potential green energy sites. The first layer identified 23 optimal wind energy sites and 138 optimal hydro energy sites. The second layer employed spatial data and the fuzzy Technique for Order Preference by Similarity to Ideal Solution algorithm to refine potential solar energy sites, yielding the top 100 optimal locations. The proposed method demonstrated a 69.01 % alignment when validated against the weighted sum method. Following site identification, an improved Geographical Information System-driven fuzzy Traveling Salesman Problem-Binary Integer Programming algorithm was proposed to integrate these sites into a reliable ring-based system topology, aiming to achieve a zero-carbon footprint. The process involved clustering by divisions and designing optimal electrical power line routing for each cluster, prioritizing minimum total distance, elevation difference, and average ground flash density. Validation against conventional methods and state-of-the-art algorithms confirmed the superior performance of the proposed approach. Additionally, an Industrial Internet of Things-based system utilizing servers, cloud platforms, and Supervisory Control and Data Acquisition systems was developed for real-time monitoring, control, and automation to address green energy intermittency. Hardware prototypes using Raspberry Pi and Industrial Internet of Things components were interfaced with SCADA systems to validate real-world applicability. Experimental results confirmed the effectiveness of the proposed methodologies. In conclusion, the proposed methodologies demonstrate the potential to overcome barriers to green energy implementation, fostering sustainable development in Sarawak. This research offers practical insights for policymakers, energy stakeholders, and researchers advancing green energy initiatives.

Keywords: Fuzzy, green energy resources, geographical information system, integrated green energy systems, industrial internet of things

Satu Kerangka untuk Pengenalpastian dan Penyepaduan Sumber Tenaga Hijau Disokong oleh Aplikasi Pemantauan, Kawalan dan Automasi Masa Nyata

ABSTRAK

Sarawak sedang beralih kepada penggunaan tenaga hijau, namun perkembangannya terhalang oleh kekurangan data spatial yang komprehensif untuk mengenal pasti lokasi optimum, teknik pengoptimuman yang tidak mencukupi bagi penyepaduan efektif tapak-tapak ini, serta strategi pemantauan masa nyata dan automasi berasaskan Internet Perindustrian Perkara yang kurang kukuh bagi mengurus sifat berselang-seli sumber tenaga hijau. Bagi mengatasi cabaran ini, satu metodologi baharu Sistem Maklumat Geografi berasaskan teknik kabur untuk Technique for Order Preference by Similarity to Ideal Solution digabungkan dengan algoritma penapisan telah dicadangkan. Pendekatan dua lapisan ini berjaya menapis tapak tenaga hijau yang berpotensi. Lapisan pertama mengenal pasti 23 tapak tenaga angin optimum dan 138 tapak tenaga hidro optimum. Lapisan kedua menggunakan data spatial dan algoritma kabur untuk Technique for Order Preference by Similarity to Ideal Solution untuk memperhalusi tapak tenaga suria berpotensi, menghasilkan 100 lokasi solar optimum teratas. Kaedah yang dicadangkan menunjukkan tahap keselarasan sebanyak 69.01 % apabila disahkan terhadap kaedah jumlah berwajaran. Selepas pengenalpastian tapak, algoritma Masalah Jurujual Perjalanan-Pengaturcaraan Sistem Angka Perduaan yang dipacu oleh Sistem Maklumat Geografi yang dipertingkatkan dicadangkan untuk menyepadukan tapak-tapak ini ke dalam topologi sistem berasaskan gelang yang boleh dipercayai, bertujuan mencapai sifar jejak karbon. Proses ini melibatkan pengelompokan mengikut bahagian dan merekabentuk laluan talian kuasa elektrik optimum untuk setiap kluster, dengan keutamaan diberikan kepada jumlah jarak minimum, perbezaan ketinggian, dan purata ketumpatan kilat tanah.

Pengesahan terhadap kaedah konvensional dan algoritma terkini membuktikan keunggulan pendekatan yang dicadangkan. Selain itu, sistem berasaskan Internet Perindustrian Perkara yang menggunakan pelayan, platform awan, dan sistem Kawalan Penyeliaan dan Pemerolehan Data dibangunkan untuk pemantauan masa nyata, kawalan, dan automasi bagi menangani sifat berselang tenaga hijau. Prototaip perkakasan menggunakan Raspberry Pi dan komponen Internet Perindustrian Perkara dihubungkan dengan sistem sistem Kawalan Penyeliaan dan Pemerolehan Data untuk mengesahkan kebolehgunaan di dunia sebenar. Keputusan eksperimen mengesahkan keberkesanan metodologi yang dicadangkan. Kesimpulannya, metodologi yang dicadangkan berpotensi mengatasi halangan pelaksanaan tenaga hijau, sekaligus menyokong pembangunan mampan di Sarawak. Penyelidikan ini memberikan pandangan praktikal kepada pembuat dasar, pemegang kepentingan tenaga, dan penyelidik dalam memajukan inisiatif tenaga hijau.

Kata kunci: *Kabur, sumber tenaga hijau, sistem maklumat geografi, sistem tenaga hijau bersepadu, internet perindustrian perkara*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ABC	Artificial Bee Colony
AC	Alternating Current
ACO	Ant Colony Optimization
AES	Advanced Encryption Standard
AFSA	Artificial Fish Swarm Algorithm
AGC	Apollo Guidance Computer
AHP	Analytical Hierarchy Process
AI	Artificial Intelligence
ANP	Analytical Network Process
API	Application Programming Interface
AR	Assurance Region
AS	Automatic Switch
ASRS	Automated Storage and Retrieval Systems
BA	Bat Algorithm
BB	Branch and Bound
BF	Brute Force
BFOA	Bacteria Foraging Optimization Algorithm
BIP	Binary Integer Programming
BOCR	Benefit, Opportunity, Cost, and Risk
BRKGA	Biased Random-Key Genetic Algorithm
CETSP	Close Enough-Traveling Salesman Problem
CGTSP	Clustered Generalized Traveling Salesman Problem
COPRAS	Complex Proportional Assessment

CPLEX	Constraint Programming Linear Programming with Extensions
CS	Cuckoo Search
DAQ	Data Acquisition
DC	Direct Current
DE	Differential Evolution
DEA	Data Envelopment Analysis
DEMATEL	Decision Making Trial and Evaluation Laboratory
DEMs	Digital Elevation Models
DP	Dynamic Programming
ELECTRE	Elimination and Choice Translating Reality
EM-MOPSO	Elitist-Mutated Multi Objective Particle Swarm Optimization
EP	Evolutionary Programming
EPC	Engineering, Procurement, and Construction
ES	Evolutionary Strategies
FA	Firefly Algorithm
FIS	Fuzzy Inference System
FST	Fuzzy Set Theory
FTSP	Family Traveling Salesman Problem
GA	Genetic Algorithm
GDP	Gross Domestic Product
GERs	Green Energy Resources
GEs	Green Energy Locations
GFD	Ground Flash Density
GHI	Global Horizontal Irradiance
GIS	Geographical Information System

GP	Genetic Programming
GPIO	General Purpose Input / Output
GSO	Glowworm Swarm Optimization
GWO	Grey Wolf Optimizer
HES	Hydro Energy Sites
HFLTS	Hesitant Fuzzy Linguistic Term Sets
HFS	Hesitant Fuzzy Sets
HMI	Human Machine Interface
IFS	Intuitionistic Fuzzy Sets
IGESs	Integrated Green Energy Systems
IIoT	Industrial Internet of Things
IP	Integer Programming
ISHDB	Internet of Things Smart Household Distribution Board
LabVIEW	Laboratory Virtual Instrument Engineering Workbench
Li-ion	Lithium-ion
LF	Load Following
LOA	Lion Optimization Algorithm
MA	Monkey Algorithm
MACBETH	Measuring Attractiveness by a Categorical Based Evaluation Technique
MCDM	Multi-Criteria Decision-Making
MILP	Mixed Integer Linear Programming
MLI	Multi-Level Inverter
MS	Manual Switch
NI	National Instrument

NN	Nearest Neighbor
NP	Nondeterministic Polynomial time
NTC	Negative Temperature Coefficient
OPC	Open Platform Communications
OS	Operating System
OWA	Ordered Weighted Averaging
PCNN	Pulse Coupled Neural Network
PDS	Parcel Delivery Services
PI-SVPWM	Proportional Integral-based Space Vector Pulse Width Modulation
PLC	Programmable Logic Controller
POWER	Prediction of Worldwide Energy Resource
PROMETHEE	Preference Ranking Organization Method for Enrichment Evaluation
PSO	Particle Swarm Optimization
PV	Photovoltaic
QL	Quality Learning
RC4	Rivest Cipher 4
RMS	Root Mean Square
SA	Simulated Annealing
SAW	Simple Additive Weighting
SCADA	Supervisory Control and Data Acquisition
SDGs	Sustainable Development Goals
SEB	Sarawak Energy Berhad
SES	Solar Energy Sites
SFLA	Shuffled Frog Leaping Algorithm

STATCOM	Static Synchronous Compensator
TDS	Technical Decision Support
TS	Tabu Search
TSP	Traveling Salesman Problem
TOPSIS	Technique for Order of Preference by Similarity to Ideal Solution
VIKOR	Multiple Criteria Optimization and Compromise Solution
WECS	Wind Energy Conversion Systems
WES	Wind Energy Sites
WLC	Weighted Linear Combination
WSM	Wireless Sensor Network

CHAPTER 1

INTRODUCTION

1.1 Background

Sarawak spans 124,450 km^2 with a population of nearly 3 million (Tang et al., 2023). It has warm, humid climate, with temperatures between 23 ℃ to 32 ℃ (Isia et al., 2022), offers significant potential for green energy. The state receives abundant solar radiation, averaging 4.21 $kWhm^{-2}$ to 5.56 $kWhm^{-2}$ (Kee et al., 2022) daily, and about 4.5 hours of sunshine per day (Arief et al., 2020), making it ideal for solar energy sites. While average wind speeds are modest at 2 ms^{-1} to 3 ms^{-1}, coastal regions experience higher velocities during the Northeast Monsoon, reaching up to 10 ms^{-1} (Lawan et al., 2020). Advancements in turbine technology now allow for effective energy capture even at these lower wind speeds. Additionally, with annual rainfall around 4,600 mm (Huang et al., 2023) and plentiful rivers, hydroelectric power is a viable energy source for the region. generation.

Identifying optimal Green Energy Resources (GERs) requires evaluating various climatic, technical, accessibility, environmental, and social factors. This comprehensive approach ensures thorough assessment of potential sites. Multi-Criteria Decision-Making (MCDM) methods are instrumental in this process, offering a structured framework to balance conflicting criteria and determine the relative importance of each factor (Shao et al., 2020). These methods, which have evolved since the 1700s, are now integral across disciplines like mathematics, engineering, and economics. Common MCDM techniques include the Analytical Hierarchy Process (AHP), Technique for Order of Preference by Similarity to Ideal Solution (TOPSIS), and Preference Ranking Organization Method for

Enrichment Evaluation (PROMETHEE), etc. Expert input is crucial in these evaluations to accurately weigh the significance of each criterion (Elkadeem et al., 2021).

Integrating identified green energy sites offers numerous benefits. Currently, power grid combines in Sarawak GERs with fossil fuel-based plants (Durin et al., 2022). Transitioning to a system solely powered by green energy is vital for sustainable development and achieving a zero-carbon footprint. This shift would reduce maintenance associated with combustion processes and eliminate risks tied to flammable materials. Moreover, utilizing inexhaustible resources like solar, wind, and hydro can lower operating costs and minimize environmental and health impacts due to reduced harmful emissions. Given the dispersed nature of GERs in Sarawak, designing optimal electrical power line routes is essential (Jong et al., 2022). This challenge parallels the Traveling Salesman Problem (TSP), which seeks the shortest path connecting multiple points (Pop et al., 2024). Both exact algorithms and approximate methods have been developed to address TSP, each with its own advantages and limitations (S. Wang et al., 2020). Comparing and validating these approaches is crucial when planning power line routes for GER integration.

Integrated Green Energy Systems (IGESs) require advanced monitoring, control, and automation to manage the variable nature of GERs and fluctuating energy demands (Deng & Lv, 2020). The Industrial Internet of Things (IIoT) facilitates real-time tracking of energy production and consumption, enabling optimized management strategies. Technologies like Supervisory Control and Data Acquisition (SCADA), combined with servers and cloud computing, enhance coordination between energy systems and demand, improving the reliability and resilience of IGESs (Albogamy et al., 2022). Developing simulation models that incorporate real-time data can provide insights into system behavior. Validating these models with hardware prototypes, such as the Raspberry Pi 5 ensures practical applicability and effectiveness in real-world scenarios (Eben Upton, 2023).

The research is conducted systematically, beginning with the identification of GERs, integrating them into the system, and proposing effective solutions for real-time monitoring, control, and automation for IGESs. The culmination of these efforts results in comprehensive research works, wherein the proposed IGESs serve as a valuable asset for long-term energy security and accelerate the transition of Sarawak to a greener power provider.

1.2 Problem Statement

The transition towards GERs in Sarawak faces significant challenges due to the absence of comprehensive spatial information regarding suitable locations. A comprehensive assessment framework is essential for systematically identifying potential green energy locations in Sarawak (S. F. Shahrom et al., 2023). The absence of a robust and effective approach to identify large-scale optimal green energy locations severely limits the potential to harness GERs efficiently (Gribiss et al., 2023). This shortfall impedes progress and delays the transition towards a greener and more sustainable future for Sarawak (Rajakal et al., 2023).

Once optimal green energy locations are identified, the subsequent task involves integrating GERs into a comprehensive system. However, there is a dearth of reliable optimization techniques, particularly in designing optimal electrical power lines routing. Critical factors such as distance, elevation difference, and lightning severity should be considered to optimize electrical power lines routing effectively. For instance, minimizing distance reduces costs, while minimizing elevation differences between green energy locations reduces construction and installation costs (Jong et al., 2022). Given high lightning occurrence in Sarawak, minimizing Ground Flash Density (GFD) among green energy locations effectively mitigate the exposure of electrical power lines to lightning strikes (Savira Kamarani et al., 2023).

Furthermore, IGESs require comprehensive monitoring, control, and automation (Prasanna Rani et al., 2023). In fact, the current state-of-the-art research lacks comprehensive IIoT-based real-time dynamic data monitoring, control, and automation strategies due to the intermittent nature of green energies such as solar and wind. These effective energy management practices are crucial for ensuring smooth power delivery and maximizing economic benefits (Lei et al., 2021; Rao et al., 2024). Therefore, it is essential to propose reliable systems to provide valuable insights for the effective utilization of GERs.

1.3 Research Questions

Research questions are listed as follows:

i. How can large-scale optimal green energy locations in Sarawak State be identified using the MCDM method?
ii. How does the integration of identified green energy locations using optimization algorithms enable the design of optimal electrical power lines routing in Sarawak State?
iii. How can effective real-time monitoring, control, and automation strategies be established for the IGESs model, and how can they be validated using a hardware model?

1.4 Research Hypothesis

Research hypotheses are listed as follows:

i. An enhanced GIS-based MCDM approach model can efficiently identify large-scale optimal green energy locations in Sarawak State.

ii. An improved GIS-driven Traveling Salesman Problem (TSP) optimization algorithm model is capable of effectively integrating the identified green energy locations by designing optimal electrical power lines routing in Sarawak State.

iii. An IIoT-based system that utilizes servers, clouds, and SCADA enables real-time monitoring, control, and automation strategies for the IGESs. The development of a hardware model can validate the effectiveness of interfacing hardware with SCADA for real-time monitoring, control, and automation strategies.

1.5 Research Objectives

Research objectives are listed as follows:

i. To propose a novel GIS-based fuzzy TOPSIS and filtration algorithms for identifying large-scale optimal green energy locations in Sarawak State.

ii. To develop an improved GIS-driven fuzzy TSP-BIP algorithm for integrating the identified green energy locations by designing optimal electrical power lines routing in Sarawak State.

iii. To establish an innovative IIoT-based system that utilizes servers, cloud, and SCADA for real-time monitoring, control, and automation for the IGESs simulation model, and to validate its effectiveness using a hardware prototype.

1.6 Research Significance

Research significances are listed as follows:

i. Meet increasing electricity demand: Growing energy needs require sustainable solutions. This research identifies optimal green energy sites to ensure a reliable supply.

ii. Reduce carbon footprint through green energy transition: Transitioning to green energy mitigates environmental risks from fossil fuels.

iii. Facilitate sustainable development and electricity export: Efficient use of renewables can meet local power needs and allow surplus export to neighboring countries, boosting regional revenue.

iv. Enhance efficiency and reliability: Integrating intermittent green energy resources requires careful planning. Research proposes optimized solutions for site selection, considering factors such as minimum distance, minimum elevation difference, and minimum GFD.

v. Establish effective monitoring and control: Implementing IIoT technology enables real-time management of the intermittent properties of GERs effectively.

vi. Realize economic benefits: Transitioning to green energy creates jobs, attracts investments, and reduces long-term energy costs.

1.7 Research Scope

The research is scoped on identifying influential criteria and collecting their corresponding spatial data, which are systematically gathered and stored in a GIS database for analysis. A novel GIS-based fuzzy TOPSIS and filtration algorithm is proposed to identify optimal green energy locations on a large scale in Sarawak. Using GIS tools, the filtration algorithm effectively eliminates unsuitable locations for green energy deployment. The top 100 optimal Solar Energy Sites (SES) are then determined through the fuzzy TOPSIS algorithm and validated using the Weighted Sum Method. The selection of Wind Energy Sites (WES) is based on the filtration process and wind speed assessments, while Hydro Energy Sites (HES) are identified through the same filtration methodology. The integration of these green energy locations (SES, WES, HES) involves clustering based on Sarawak's divisions. An improved GIS-driven fuzzy TSP-BIP algorithm is developed to design optimal electrical power line routing for each cluster, considering three primary parameters: distance, elevation difference, and Ground Fault Detection (GFD). The results obtained from this algorithm are compared and validated against other optimization algorithms. Furthermore, an advanced IIoT-based system is established, incorporating servers, cloud infrastructure, and SCADA for the IGESs. A simulation model of the IGESs is developed using MATLAB Simulink with a ring system topology. Real-time dynamic data are modeled and integrated into the IGESs for continuous monitoring. Manual control of AC and DC loads is conducted for maintenance purposes, while automation is implemented to manage DC and AC faults using SCADA window scripts. Additionally, a hardware model is developed and interfaced with SCADA to validate the effectiveness of the IIoT-based IGESs simulation model in real-time monitoring, control, and automation strategies.

1.8 Thesis Structure

The thesis structure consists of five chapters. Chapter 1 introduces the research, including the background, problem statements, research questions, hypotheses, objectives, significance, and scope. This provides a clear rationale for conducting the research. Chapter 2 conducts a comprehensive literature review, analyzing state-of-the-art criteria for GERs, MCDM methods, optimization algorithms, and IIoT-based systems for effective real-time monitoring, control, and automation. Rigorous analysis of state-of-the-art research is crucial for identifying research gaps. Chapter 3 presents the methodology, including GIS data collection and construction, GIS-based fuzzy TOPSIS and filtration algorithms, GIS-driven fuzzy TSP-BIP optimization algorithm, and the establishment of an IIoT-based system for effective real-time monitoring, control, and automation strategies with simulation models and hardware models. Chapter 4 analyzes the influential criteria, discusses the results of developed models, and performs validation to measure the reliability and robustness of the models. Finally, Chapter 5 provides the conclusion of the research, summarizing the findings, discussing their implications, and offering recommendations for further directions and contributions to the overall research work.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

This chapter offers a comprehensive review of the current state-of-the-art research on the identification and integration of GERs in Sarawak, along with methods for monitoring, controlling, and automating these resources. Initially, the focus lies on harnessing solar, wind, and hydro energy. The review takes into account crucial criteria from five perspectives (climatic, technical, accessibility, environmental, and social) when identifying a large scale of optimal green energy locations. Following this, an in-depth exploration of MCDM methods is conducted, as these techniques are widely employed in GERs identification. MCDM methods provide decision-makers with the means to evaluate multiple criteria and alternatives, facilitating the identification of optimal green energy locations. Subsequently, the chapter delves into the integration of these identified green energy locations through optimization algorithms. Both exact and approximate algorithms are scrutinized to assess their merits and demerits. Additionally, the applications of TSP utilizing these optimization algorithms are studied and analyzed. Furthermore, the chapter reviews the current implementation of real-time monitoring, control, and automation for IGESs. By synthesizing insights from existing literature, this review aims to discover the valuable research gaps in current research works.

2.2 Components of Influential Criteria for GERs

The categorization of influential criteria into distinct domains, including climatic, technical, accessibility, environmental, and social factors, is widely acknowledged as relevant for identifying optimal green energy locations (Shao et al., 2020). Therefore, all the influential criteria within these five domains have been thoroughly discussed and explored. Additionally, a summary outlining the necessary criteria for SES, WES and HES has been provided.

Solar radiation is the cornerstone for SES, typically quantified by Global Horizontal Irradiance (GHI). GHI reflects the total solar energy incident on photovoltaic arrays and is influenced by geographical factors such as longitude and latitude, as well as atmospheric conditions including humidity and temperature (Karipoğlu et al., 2022). An economically viable solar installation generally requires an average GHI of about 5 $kWhm^{-2}day^{-1}$ (Ayough et al., 2022), which serves as a key indicator of site potential. Air temperature is another critical factor where lower temperatures enhance PV efficiency and prolong panel lifespan, while even modest temperature rises (e.g., 1 ℃) can reduce power output by 0.5 % to 0.6 % due to accelerated material degradation (Grubišić-Čabo et al., 2016). Humidity is measured as a percentage of maximum moisture content, also plays a significant role. High humidity can decrease efficiency by promoting water intrusion between cells and triggering corrosion, thereby increasing system failure risk (Günen, 2021). The duration of sunlight (typically around 7 hours per day) is essential as it directly correlates with total energy yield and is modulated by factors such as altitude and atmospheric clarity (Ruiz et al., 2020). Wind speed can influence solar installations by enhancing the cooling of PV modules, although excessive wind may also impose mechanical stresses (Ullah et al., 2021). Precipitation and cloudiness further affect SES performance; increased rainfall may reduce output by causing soiling or shading effects and cloud cover can diminish PV efficiency by 10 % to 25 % through scattering and refraction of sunlight (Perveen et al., 2019). Minor factors, including variations in air pressure, water vapor, total ozone, and lightning flash density, also contribute to the overall performance of solar panels. For WES, wind energy potential is predominantly dictated by wind speed, which is essential for turbine activation and energy production. Operational wind speeds typically have a cut-in threshold of around 3 ms^{-1}, with optimal performance near 6 ms^{-1}, and a cut-off limit around 25 ms^{-1} to prevent damage (Saint-Drenan et al., 2020). Alongside wind speed, wind power density is a critical metric that represents the average annual power available per unit area of a turbine's swept zone; a minimum threshold of approximately 200 Wm^{-2} is often required for viable energy extraction (Asadi & Pourhossein, 2021). Air density, influenced by temperature and altitude, affects the mass of air interacting with the turbine blades and thus the energy captured; higher air density typically improves power output. Turbulence intensity is another important subcriterion, as excessive turbulence can lead to rapid fluctuations in wind speed and direction, adversely affecting turbine stability and efficiency (Asadi & Pourhossein, 2021). Temperature and atmospheric pressure further modulate air density and wind behavior, while relative humidity has a subtler impact by slightly reducing air density. Wind direction is also pivotal, as optimal turbine orientation must align with prevailing wind patterns to maximize energy capture (Lawan et al., 2020). For HES, climatic factors focus on water resource characteristics. The water flow rate is the foremost criterion, as it directly determines the kinetic energy available for conversion into electricity and influences both turbine efficiency and reservoir management (Temel et al., 2023). Water temperature is also critical; colder water is preferred since it sustains optimal turbine performance and minimizes adverse impacts on efficiency, whereas higher temperatures can diminish output and pose risks to aquatic ecosystems (Simonović et al., 2021). Precipitation patterns are essential in ensuring a reliable and steady supply of water, which in turn supports consistent flow rates and sustainable energy production. Regions with ample and evenly distributed rainfall are generally more suitable for hydroelectric development, as they facilitate reservoir replenishment and stable power generation (Jafari et al., 2021).

Technical evaluations for SES begin with the slope, a key parameter affecting solar panel orientation and performance. Studies indicate that a slope below 11 % is recommended, with an ideal target of less than 1 % to maximize solar radiation capture and minimize shading effects (Doorga et al., 2019). In addition, land cover or land use analysis using GIS and satellite data helps differentiate between natural and anthropogenic surfaces; open fields with minimal obstructions are preferred over urban or densely vegetated areas (Rios & Duarte, 2021). Aspect is defined by the latitude of the sites, azimuth, and overall orientation relative to the sun, further influences the amount of solar radiation received. Elevation is determined via Digital Elevation Models (DEMs), plays a supportive role as higher altitudes often experience clearer skies and stronger solar irradiance (Noorollahi et al., 2022). Soil type is another critical factor; stable, flat surfaces such as concrete or asphalt are ideal for ground-mounted photovoltaic systems, while soils with large pore sizes may undermine mounting stability (Deveci et al., 2022). Lastly, technical considerations extend to the availability of skilled manpower and technical expertise for the installation, maintenance, and optimization of PV systems, ensuring that the chosen sites can be effectively developed (C. N. Wang et al., 2022). In the technical assessment of WES, slope remains a crucial parameter. A maximum slope of 5 ° is recommended to mitigate turbulence, which can disrupt wind flow and reduce turbine efficiency (Effat & El-Zeiny, 2022). Elevation is equally significant; higher elevations are generally favored as they offer reduced obstructions and more stable wind patterns, whereas lower elevation sites tend to experience higher turbulence (Asadi & Pourhossein, 2021). Land cover and the overall site area are vital; open, unobstructed landscapes enable turbines to harness wind energy more effectively, optimizing power capture (Ullah et al., 2021). The technical viability of WES also hinges on the local availability of skilled personnel who can support the installation, operation, and routine maintenance of wind turbines, thereby ensuring operational stability and reducing downtime (C. N. Wang et al., 2021). Finally, soil conditions including drainage, stability, load-bearing capacity, and erosion resistance are essential for designing secure and durable turbine foundations, necessitating thorough geotechnical investigations (Shorabeh, Firozjaei, et al., 2022). Technical evaluations for HES center on the characteristics of water resource and reservoir. Key factors include the size of reservoir, shape, and storage capacity, which are critical for ensuring a consistent water supply and stable power generation (Haas et al., 2022). An understanding of both upstream water availability and downstream environmental impacts is necessary for sustainable reservoir management (Marcelino et al., 2021). Water level metrics, such as dead water level, normal water level, and initial water level, play important roles in operational planning, while the head (the vertical distance between the water source and turbine intake) is directly related to the potential energy output; a higher head generally yields greater energy (Urošević & Marinović, 2021). The feasibility of hydro implementation is further enhanced by the presence and connectivity of significant water bodies, including rivers, streams, or lakes. Turbine efficiency, which reflects the conversion of kinetic energy from flowing water into electricity, is a crucial indicator of site potential (Xiong et al., 2021). Additional technical considerations include assessing the stability of the dam toe, the viability of diversion weirs, the reservoir's pondage capacity, and adequate water depth for turbine installation. Lastly, the surrounding land slope, land use patterns, and the characteristics of the river zone including sediment transport and flow dynamics are evaluated to address environmental impacts and ensure construction feasibility (Kuriqi et al., 2021).

Accessibility is a critical factor in the siting and operation of GERs such as solar, wind, and hydro installations. Proximity to existing electrical power lines is paramount; locations within 50 km of power networks tend to reduce transmission losses and lower infrastructure costs, enhancing overall efficiency (Karipoğlu et al., 2022). Similarly, access to main roads is essential to facilitate the transport of personnel and equipment during construction and maintenance, though maintaining a safe distance of around 1 km from major roads is generally recommended to mitigate associated risks (Saraswat et al., 2021). Furthermore, the positioning of GERs relative to residential areas plays a dual role. On one hand, proximity can shorten electrical power lines and reduce grid losses, thereby improving cost-effectiveness (Ayough et al., 2022). On the other hand, due to the large land requirements of green energy installations, a buffer of approximately 0.5 km to 1.5 km from residential zones is advised to balance accessibility with safety considerations (Saraswat et al., 2021). Additionally, assessing local electricity demand by examining industrial, commercial, residential, and agricultural load pattern, is crucial for determining optimal GER locations (Bohra & Anvari-Moghaddam, 2022). Finally, economic factors such as construction costs, annual income levels, electricity prices, and governmental subsidy policies further influence the feasibility and overall viability of green energy projects (Deveci et al., 2022).

Environmental criteria for green energy systems emphasize minimizing negative impacts and safeguarding ecosystems. For solar and wind energy, although these technologies generally have minimal adverse effects, careful site selection is necessary. For instance, installations near water bodies such as rivers or lakes can risk leakage and runoff contamination, potentially harming water quality and aquatic life (Colak et al., 2020). To mitigate such risks, a 500 m buffer zone from water bodies is advised. Moreover, both solar and wind systems generate heat through their materials, which may reach levels of concern if not properly managed. The placement of these installations in protected areas, such as national parks, wildlife sanctuaries, or other ecologically sensitive zones, should be strictly avoided to prevent disruption of natural habitats and endangered species (Zahid et al., 2021). Additionally, aesthetic and social factors necessitate a buffer of around 2 km from residential areas to balance infrastructural benefits with community acceptance. For HES, environmental assessments focus on ensuring adequate land area for constructing dams, reservoirs, and power plants without encroaching on sensitive ecosystems. Site selection must consider the impact on aquatic life, particularly regarding fish migration and spawning grounds, with potential mitigation through the incorporation of fish passages or fish-friendly turbine designs (Kuriqi et al., 2021). It is also critical to evaluate flood zones and conduct geological surveys to assess stability and prevent adverse effects on both the environment and nearby communities (Urošević & Marinović, 2021).

The implementation of green energy systems must prioritize social criteria to ensure that projects foster community well-being and social sustainability. Literature indicates that social factors such as job creation, community involvement, and the preservation of cultural heritage are essential for garnering local support (Y. Wu et al., 2021). Population density plays a critical role; higher densities can amplify electricity demand and potentially intensify conflicts over energy infrastructure, while lower densities may present fewer social disruptions but still require careful evaluation of local values and environmental impacts (Deveci et al., 2021). Moreover, establishing clear regulatory boundaries is vital, as national and local policies, subsidies, and legal frameworks related to land usage and contractual agreements influence the financial and operational viability of these projects (Wolfshohl & Sweers, 2021)). Engaging with communities through surveys, interviews, and discussions is imperative to assess public opinion and cultural significance, as resistance from local inhabitants can undermine project success (Urošević & Marinović, 2021; C. N. Wang et al., 2021). Additionally, economic indicators such as annual income levels, electricity prices, and governmental subsidy policies also play a role in shaping the social acceptance and overall feasibility of green energy initiatives (Deveci et al., 2021).

2.3 Merits and Demerits of Existing MCDM Methods

MCDM is an established academic discipline that provides a structured framework for making informed decisions involving multiple criteria or objectives. Several factors are often considered in many real-world decision-making scenarios, each with varying levels of importance or weighting (Emovon & Oghenenyerovwho, 2020). MCDM provides a methodological approach to assess and compare options based on multiple criteria and identify the optimal alternative.

Saaty introduced the Analytic Hierarchy Process (AHP) in the 1970s, which is one of the most widely recognized approaches in MCDM. AHP decomposes a complex decision problem into a structured hierarchy comprising a goal, a set of criteria, and a series of alternatives (Mastrocinque et al., 2020). Decision-makers perform pairwise comparisons among the criteria and alternatives, and through normalization (Abdul et al., 2022), derive relative weights that lead to an overall ranking of options. This method is particularly valued for its systematic and transparent framework that simplifies complex decision problems into manageable components. However, a significant drawback of AHP is its reliance on subjective judgments during the pairwise comparison process. Such subjectivity may introduce biases that can affect the consistency and reliability of the outcomes (Colak et al., 2020). Analytic Network Process (ANP) represents an evolution of AHP by incorporating the interdependence among decision elements. Unlike the strictly hierarchical structure of

AHP, ANP models the decision problem as a network where goals, criteria, sub-criteria, and alternatives are interconnected, allowing for the capture of feedback and interrelationships among factors (Ilbahar et al., 2019). This feature makes ANP especially useful in complex scenarios where criteria do not operate independently. However, this advantage comes with increased computational complexity and reliance on subjective evaluations. The detailed pairwise comparisons and construction of a supermatrix can be resource-intensive and may complicate the decision process, particularly in cases where interdependencies are not pronounced (Alizadeh et al., 2020).

Benefit, Opportunity, Cost, and Risk (BOCR) framework provides a distinct perspective by simultaneously considering four critical aspects of decision-making. In this approach, alternatives are evaluated based on current benefits, future opportunities, costs incurred, and associated risks (İ. Kaya et al., 2018). BOCR is effective in highlighting the trade-offs inherent in any decision, as it explicitly links cause and effect. Despite its ability to clarify the decision process, BOCR faces challenges in establishing consensus on the weights for each component. Moreover, its focus is sometimes restricted to comparing only a couple of dimensions, potentially oversimplifying complex scenarios that require a more nuanced analysis (Wei, 2021). Complex Proportional Assessment (COPRAS) method is based on the principle of assessing the relative significance of alternatives through the normalization of a decision matrix and the integration of weight coefficients (Stefano et al., 2015). COPRAS offers a systematic procedure that accounts for both beneficial and non-beneficial criteria, leading to a comprehensive ranking of alternatives. Its strength lies in the balanced consideration of multiple factors; however, its reliance on accurately defined weights and precise data values can make the computational process quite complex. The sensitivity of the method to these input parameters often poses a challenge in practical applications where data uncertainty is prevalent (Mohanrasu et al., 2023; Patil et al., 2022).

Data Envelopment Analysis (DEA) is primarily employed to evaluate the efficiency of comparable decision-making units by analyzing the relationship between inputs and outputs (Dutta et al., 2022). The method can be implemented using input-oriented or output-oriented models, offering flexibility in assessing the relative performance of different units. One of the key strengths of DEA is its ability to tailor weight assignments to reflect optimal performance for each unit. However, this flexibility can lead to the assignment of overly favorable weights, which may result in inflated efficiency scores. Moreover, DEA typically focuses on efficient units, thereby limiting its broader applicability in contexts where comparative analysis across a wider range of options is required (Fotova Čiković et al., 2022; C. N. Wang et al., 2022). ELECTRE family of methods was initially introduced in 1968 and subsequently refined by Bernard Roy, is designed to address selection and ranking problems using an outranking approach (Mary & Suganya, 2016). ELECTRE employs concordance and discordance indices to evaluate and compare alternatives. Among its variants, ELECTRE III is particularly noted for its effectiveness in ranking, although other versions are more commonly used for selection tasks (Z. S. Chen et al., 2021). Despite its clear theoretical basis, the method is sensitive to the assigned weights and can encounter ambiguity in determining preference thresholds. Such sensitivity often necessitates careful calibration to ensure that the final rankings accurately reflect the intentions of decision-makers (Z. S. Chen et al., 2021).

Fuzzy logic operation is based on the fuzzy set theory introduced by Lotfi Zadeh in 1965 and it is powerful for managing uncertainty and vagueness in decision-making processes (Y. Liu et al., 2020). By utilizing various membership functions, such as triangular, trapezoidal, S-shaped, and Gaussian, fuzzy logic enables the incorporation of imprecise and qualitative data into quantitative models. Extensions such as Type-2 fuzzy sets and Hesitant Fuzzy Sets (Farhadinia & Xu, 2019) further enhance the method's ability to handle ambiguity. While fuzzy logic is particularly advantageous for its capacity to manage uncertainty, the challenge remains in defining appropriate linguistic terms and membership functions, which can be inherently subjective and difficult to standardize (Lu et al., 2022). Geographic Information System (GIS) may not be regarded as an MCDM method, but the integration of GIS with decision models has significantly enhanced spatial analyses, especially in areas such as renewable energy and land-use planning (Zhou et al., 2020). GIS platforms combine hardware and software to manage, analyze, and visualize spatial data in both vector and raster formats (Zambrano-Asanza et al., 2021). This integration allows decision-makers to overlay various criteria on geographic maps, facilitating the identification of optimal locations. However, the primary limitation of GIS lies in its focus on spatial issues, which restricts its application to non-geographic decision problems (Martínez-Martínez et al., 2022; Razeghi et al., 2023).

Goal Programming was developed by Charnes in 1955, and it offers a strategy for tackling decision problems characterized by multiple, often conflicting, objectives. Instead of striving solely for maximization or minimization, goal programming is oriented towards achieving predefined target values by minimizing the deviations from these targets (Farsi et al., 2023). This method is particularly well-suited for situations where a compromise solution is necessary. Its structured approach facilitates a balanced evaluation of competing criteria; however, the effectiveness of goal programming is highly contingent upon accurately setting target values and assigning weights. If not properly calibrated, these factors can lead to unclear utility functions and suboptimal decision outcomes (Arjomandi et al., 2021; Jain & Potdar, 2021). The concept of Linguistic Quantifiers was first introduced by Zadeh in 1983, offers an innovative approach to integrating qualitative assessments into decision models (Genç et al., 2020). Linguistic quantifiers allow decision-makers to express satisfaction levels using natural language terms such as “few” or “many,” thereby bridging the gap between qualitative judgment and quantitative analysis. Despite the benefits of capturing subjective preferences, the absence of a standardized interpretation for these terms poses significant challenges, potentially leading to inconsistencies in weight assignment and overall evaluation (Yalçin & Pehlivan, 2019; Yu et al., 2022).

The Ordered Weighted Averaging (OWA) operator was introduced by Yager in 1988, represents an aggregation method where weights are assigned based on the ordered position of individual criterion scores rather than their inherent attributes (Firozjaei et al., 2019). This operator is particularly useful in balancing conjunctive and disjunctive behaviors within a single framework, allowing for compensation between criteria. While OWA facilitates a flexible and balanced aggregation of data, determining the appropriate set of ordered weights remains a challenging and somewhat subjective process, which can impact the reliability of the final evaluation (Z. S. Chen et al., 2019; Csiszar, 2021). The Preference Ranking Organization Method for Enrichment Evaluation (PROMETHEE) is a user-friendly MCDM approach that employs pairwise comparisons to derive rankings for a set of alternatives (T. Chen et al., 2020). PROMETHEE is available in two primary versions: PROMETHEE I, which provides partial rankings, and PROMETHEE II, which produces a complete ranking of alternatives (Andreopoulou et al., 2018). This method is particularly effective when the decision problem involves a limited number of alternatives with conflicting criteria. However, the outcomes of PROMETHEE are notably sensitive to the selection of preference functions and weight assignments, which may lead to ambiguity in the final rankings unless carefully managed (Karczmarczyk et al., 2018; Y. Wu et al., 2019).

Rank Correlation Analysis utilizes Spearman's rank correlation coefficient to compare the similarity between rankings generated by different MCDM methods (Zafar et al., 2021). This approach is especially useful for evaluating ordinal data and can help in assigning weights based on the relative consistency among various methods. Although it offers a straightforward means of assessing the comparability of different ranking schemes, its applicability may be limited when addressing decision problems that involve more than two criteria, thereby constraining its utility in more complex settings (Kou et al., 2020; Sałabun & Urbaniak, 2020). One of the earliest and simplest methods in MCDM is Simple Additive Weighting (SAW), first proposed by Churchman and Ackoff in 1945 (Vafaei et al., 2022). SAW involves assigning weights to each criterion and computing an overall score for each alternative by summing the products of the criterion weights and the corresponding performance ratings. Its ease of implementation and intuitive calculation have ensured its continued use in various applications. However, SAW is highly sensitive to the choice of weights and may fall short when it comes to handling trade-offs in situations where criteria interact in complex or nonlinear ways (Abrams et al., 2018).

Technique for Order Preference by Similarity to Ideal Solution (TOPSIS) was developed by Yoon and Hwang in 1981, and it is based on the concept that the optimal alternative should have the shortest Euclidean distance from the positive ideal solution and the longest distance from the negative ideal solution (Alghassab, 2022). TOPSIS provides a clear and computationally efficient framework for ranking alternatives by comparing their geometric proximity to these idealized benchmarks. Despite its intuitive appeal and ease of use, TOPSIS is significantly dependent on the accuracy of weight assignments and the assumption of monotonicity in the utility of each criterion, which can limit its effectiveness in certain decision environments (Alghassab, 2022; Bilgili et al., 2022). Multiple Criteria Optimization and Compromise Solution (VIKOR) method has gained recognition for its ability to derive a compromise solution in situations involving conflicting criteria. VIKOR evaluates alternatives based on their closeness to an ideal solution using specific distance metrics while considering the trade-offs among criteria (Gul et al., 2016). This method is particularly useful in complex decision scenarios where a balance between competing objectives is necessary. However, its performance is sensitive to the selection of weights and the setting of preference thresholds, which may lead to ambiguity in the final rankings if not appropriately managed (Abdul et al., 2022; Akram et al., 2021).

Lastly, the Weighted Linear Combination (WLC) method provides a straightforward approach for multi-criteria assessments, particularly in spatial analyses such as land suitability evaluations (Lim & Afifah Basri, 2022). In WLC, each criterion score is multiplied by its assigned weight, and the results are summed to yield an overall suitability score for each alternative. While this method is simple and easy to implement, its outcomes are highly dependent on the predetermined weights and assumptions regarding the importance of each criterion. Such sensitivity can sometimes undermine the robustness of the evaluation, especially when complex interactions between criteria exist (Barzehkar et al., 2019; Estelaji et al., 2023).

2.4 Deployments of MCDMs Strategies in GERs

An assessment of green energy technologies for electricity generation in Turkey was conducted by Boran et al. (2012) using intuitionistic fuzzy TOPSIS. It highlights the crucial role of green energy in reducing greenhouse gas emissions and meeting global energy needs, emphasizing Turkey's favorable geographical position for exploiting renewable resources. Specifically, it evaluates green energies (solar, hydro, wind, and geothermal) for their long-term viability in Turkey's energy sector. The findings reveal hydropower as the leading renewable energy technology in Turkey, followed by wind power, geothermal, and photovoltaic. Consequently, the paper suggests that future green energy policies in Turkey should prioritize hydropower and wind power based on this assessment. Moreover, the research highlighted the suitability of intuitionistic fuzzy TOPSIS for MCDM in evaluating green energy technologies, given its ability to accommodate the uncertain perceptions of decision-makers. While it examines multiple green energy options, its limitations include a lack of validation and analysis and a focus solely on determining types of green energy without considering location selections. Şengül et al. (2015) develops a decision support framework for ranking green energy supply systems in Turkey. It employs the fuzzy TOPSIS method to analyze and rank these systems. Weight values for criteria are determined using Interval Shannon's Entropy methodology, with sensitivity analysis conducted at 0.1, 0.5, and 0.9 cutting levels. The research finds that HES is the most desirable green energy supply system in Turkey, followed by geothermal sites and WES. It discusses the growing demand for energy due to population growth, urbanization, and industrialization, emphasizing the significance of green energy in meeting this demand. Insights are provided for policymakers regarding optimal resource allocation and investment in renewable energy systems.

Strengths include sensitivity analysis, while limitations include a focus solely on determining green energy types without exploring location selection analysis.

The selection process for solar panels in PV systems was thoroughly examined by Sasikumar & Ayyappan (2019). It proposes an integrated approach that combines fuzzy AHP and TOPSIS to balance subjective and objective criteria in panel selection. Previous MCDM methods for green energy and solar power selection are reviewed, emphasizing the need to address both subjective and objective parameters, while the proposed model integrates fuzzy logic with MCDM to handle incomplete data and conflicting goals, enhancing decision-making accuracy. A case study demonstrates the proposed model's application in selecting the best solar panel based on various criteria, effectively quantifying tangible sub-criteria and improving decision-making accuracy. The strengths include consideration of risk indices, and limitations involve the sole focus on selecting types of solar panels without considering location selections. A novel methodology was introduced by Rani et al. (2020) for selecting GERs by integrating fuzzy sets theory and decision-making methods. The approach involves developing new divergence measures for fuzzy sets and extending the fuzzy TOPSIS method. A detailed step-by-step process of the extended fuzzy TOPSIS approach is presented, including criteria weight calculation, alternative estimation, and optimal energy source determination based on linguistic assessments from decision experts. The research also extends a new fuzzy decision-making method to evaluate and rank renewable energy sources, employing a survey to identify nine important criteria. Future work includes extending the proposed method to other fuzzy set types and applying it to evaluate sustainable ecosystem management strategies, medical treatment assessment, and other decision-making problems. The study contributes a novel approach to evaluating and ranking GERs under uncertainty, addressing the limitations of subjective decision-making and providing a systematic methodology for assessing and comparing green energy options. It also highlights potential future applications of the proposed method in various decision-making contexts. The limitations involve the concentration on determining the types of GERs to be used without analyzing location selections.

Solangi et al. (2021) considers the identification and overcoming of obstacles hindering the adoption of green energy technologies in Pakistan. Emphasizing the importance of green energy technologies for sustainable development, particularly in rural areas, the paper suggests green energy technologies as a solution to reduce energy import dependency. Strategies like capital subsidies and green energy targets are evaluated to overcome barriers. The research guides future efforts in deploying green energy technologies for sustainable energy planning in Pakistan, utilizing a comprehensive methodology to assess and prioritize barriers, sub-barriers, and strategies. The insights are applicable not only to Pakistan but also to other developing countries facing similar challenges in green energy technology development. Strengths include comparative validation of results, while limitations lie in the focus on highlighting barriers without exploring location selection analysis. An intuitionistic fuzzy TOPSIS methodology was employed by Bilgili et al. (2022) to analyze sustainable green energy options within the constraints of sustainable growth and established criteria in Turkey. Assessing seven options including wind, solar, geothermal, biomass, wave hydro, and hydrogen, against 25 criteria, it endeavors to rank the optimal level for selecting green energy sources in Turkey. The research findings indicate that solar energy emerges as the most suitable GERs for sustainable growth in Turkey, with the evaluation of capital or investment cost emerging as a crucial criterion. These outcomes offer valuable guidance for Turkey's green energy choices amidst considerations of economic growth and environmental concerns. A key contribution of this study lies in its provision of a flexible and effective approach to navigating complexity and uncertainty in green energy selection, offering vital perspective into optimal green energy choices for sustainable development in Turkey. A limitation is its primary focus on determining specific types of GERs without exploring location selections.

A rising demand for electrical energy was addressed by Ponce et al. (2022) in manufacturing, driven by market shifts and population growth, with a focus on leveraging solar energy to meet these needs. Introducing a fuzzy TOPSIS approach, it evaluates solar panel companies within a manufacturing context using the S4 framework, encompassing sensing, smart, sustainable, and social aspects. The application of the proposed fuzzy TOPSIS method in a case study involving three decision-makers and solar panel companies demonstrates its efficiency in selecting the most suitable solar panel company. The authors stress the importance of decision-making in solar energy deployment and operation, highlighting the significance of selecting S4 features and employing a multi-criteria methodology to comprehensively evaluate solar energy systems based on specific company requirements. Furthermore, the paper discusses prospects, including automating the method through computational programs to aid manufacturing companies in selecting the best-suited solar panel company. The strength of the research lies in its evaluation of solar panel companies, while limitations include a lack of location selection analysis and results validation. Hasti et al. (2023) explores the PV farm potential in Kurdistan Province, Iran, employing GIS-based site-selection methods. It utilizes the ANP, AHP, and TOPSIS to assess spatial suitability. Results indicate 11.19 % of the region is suitable for PV farms, with 10.05 % having medium suitability and 1.14 % highly suitable. These areas could generate 3.2 % of Iran's total electricity consumption and meet Kurdistan's demand entirely. Additionally, the construction and operation of these farms could create significant job opportunities and reduce carbon emissions. Despite strengths like utilizing multiple MCDM methods, limitations include the absence of a fuzzy logic comparison, and no exact solar site locations are provided.

Suitable SES in Tehran province, Iran, was evaluated by Saeidi et al. (2023) using TOPSIS decision analysis and fuzzy Boolean logic based on GIS. Utilizing GIS for data processing, the study collected essential information for effective resource assessment. Through MCDM, approximately 95 % of the region was deemed unsuitable for SES construction, while 5 % was identified as suitable. Nine locations totaling 22 km^2 were selected for SES construction, with Ghiyeh Naserieh in the southern part of the province chosen as the preferred location using the fuzzy TOPSIS method. The research addressed Tehran's pollution issues, utilizing GIS-generated thematic maps to identify and prioritize potential SES based on various criteria including geographical, environmental, economic, social, technical, safety, and infrastructure factors. However, limitations were noted regarding data accuracy and completeness, as well as potential drawbacks of the methods used, and it is only focused on specific location analysis. Almasad et al. (2023) presents a site suitability analysis for implementing SES in Saudi Arabia using a fuzzy MCDM-based approach. The research identifies suitable sites by combining fuzzy AHP and PROMETHEE II. The model considers 12 factors divided into technical and economic criteria to minimize construction costs and maximize power output. The resulting suitability map indicates that 65.1 % of the studied area is "most to highly suitable" for solar power projects. Validation of the model's predictivity shows that 90.6 % of future projects fall within these suitable areas. The paper also includes a sensitivity analysis to examine the effect of economic factors on the suitability results. The paper details the process of criteria selection, weighting, and alternatives evaluation, highlighting the impact of climatic and economic factors on suitability. Fuzzy AHP and PROMETHEE II are used to generate the final suitability map, with validation showing high prediction accuracy for future PV projects. The strengths include determining exact locations for solar sites and conducting validation and analysis, while the limitation is the consideration of a low number of solar sites and the absence of a filtration framework.

Overall, the research studies addressing green energy technologies and decision support frameworks, multiple GERs are prioritized, and sensitivity analysis is incorporated. However, both Boran et al. (2012) and Şengül et al. (2015) are limited by a lack of validation and the omission of location selection analysis. In the realm of solar panel selection for PV systems, integration of fuzzy logic with MCDM techniques is a key strength as shown by Sasikumar & Ayyappan (2019), while Rani et al. (2020) introduces an extended fuzzy TOPSIS approach to rank GERs under uncertainty, yet both studies focus solely on selecting types rather than considering geographic factors. Comprehensive methodologies for assessing green energy adoption and providing flexible selection strategies are further highlighted by Solangi et al. (2021) and Bilgili et al. (2022), respectively, though they too overlook location analysis. On the other hand, practical applications such as the efficient selection of solar panel companies via a fuzzy TOPSIS approach was developed by Ponce et al. (2022), the evaluation of PV farm potential using multiple MCDM methods was designed by Hasti et al. (2023), and the use of GIS to address pollution issues in specific regions performed by Saeidi et al. (2023) demonstrate innovative approaches but are constrained by limitations including data accuracy, absence of fuzzy logic comparisons, and a narrow focus on location-specific analyses.

2.5 Strengths and Limitations of Current Optimization Algorithms

Optimization algorithms are essential in modern applications like routing, energy management, scheduling, logistics, and manufacturing. They aim to find optimal or near-optimal solutions for specific objectives. The two main types are exact and approximate algorithms. Exact algorithms tackle NP-hard problems to determine the best solution, while approximation algorithms quickly identify near-optimal solutions for computationally challenging NP-hard problems. Figure 2.1 illustrates an overview of these algorithms.

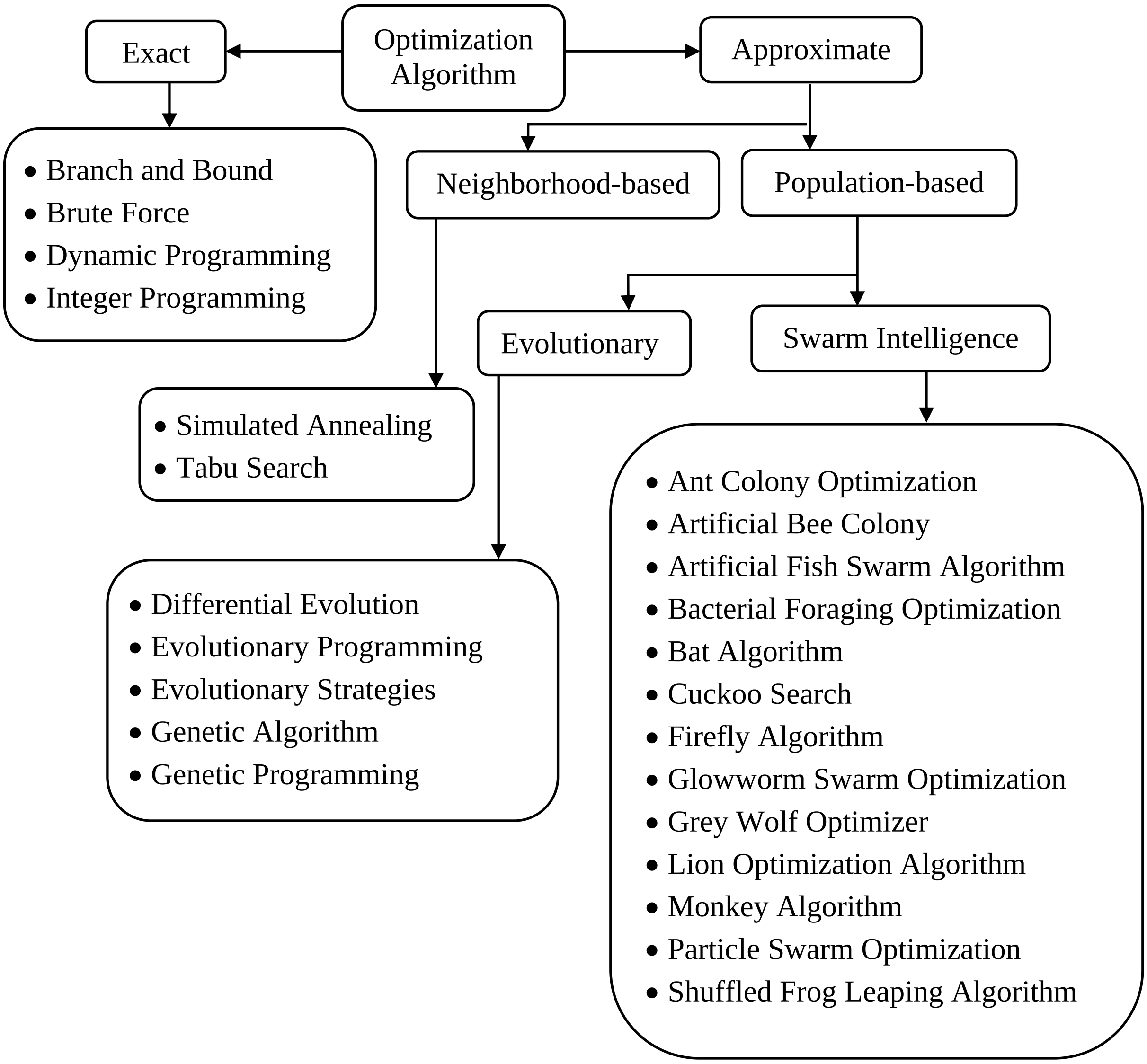


Figure 2.1: Overview of Optimization Algorithms

One of the classical approaches, Branch and Bound (BB) was developed in the mid-1960s, where it partitions a complex problem into several independent subproblems using a bounding function (Narendra & Fukunaga, 1977). By iteratively exploring promising branches while discarding regions that cannot yield better solutions, BB efficiently addresses discrete and combinatorial optimization challenges such as crew scheduling and network flow. Its strength lies in the rapid verification of bounds that often preclude further partitioning, though its performance is heavily dependent on the quality of the bounding function (S. Wang et al., 2020). Brute Force (BF) algorithm is an exhaustive search technique that sequentially compares a target pattern against every possible substring in a text (Heule & Kullmann, 2017). Its straightforward approach guarantees that every occurrence of the pattern is found, which is beneficial for small-scale problems and educational purposes. However, due to its complete enumeration of possibilities, BF suffers from severe computational inefficiency in large texts or complex pattern searches (Ptaszynski et al., 2019). Dynamic Programming (DP) leverages the optimal substructure property of problems by breaking them into overlapping subproblems and solving each recursively in a bottom-up manner. This method, which efficiently reuses computed results through memorization, is widely used in combinatorial optimization (D. Wang et al., 2024). Despite its elegance and power, DP can face challenges such as high memory consumption and potential stack overflow issues when dealing with large-scale problems (D. Liu et al., 2021). Integer Programming (IP) is a form of linear programming where decision variables are constrained to integer values. This formulation, especially in its binary (BIP) (Akay et al., 2021) and mixed (MILP) (C. Li et al., 2022) variants, is inherently NP-hard, making it suitable for problems where only discrete decisions are acceptable. While IP offers guaranteed convergence and global optimality under well-defined conditions, its inability to naturally incorporate nonlinear variables can limit its applicability (Mansini et al., 2015).

Simulated Annealing (SA) was inspired by the metallurgical process of annealing, as it probabilistically accepts not only improvements but also occasional degradations in solution quality to escape local optima (Delahaye et al., 2019). By gradually lowering a "temperature" parameter, SA transitions from exploration to exploitation. Its adaptability to both discrete and continuous problems is a major asset, though the need for careful calibration of its cooling schedule and high computational time may restrict its use in time-critical applications (İLHAN, 2021). Tabu Search (TS) enhances traditional local search methods by incorporating adaptive memory structures (tabu lists) that record previously visited solutions to prevent cycling. This mechanism encourages the exploration of new regions in the solution space and aids in overcoming local optima (Chou et al., 2021). Its effectiveness depends on carefully tuning its parameters (e.g., memory length and aspiration criteria) to balance intensification and diversification within the search (Gmira et al., 2021).

Differential Evolution (DE) is a population-based, stochastic optimization algorithm that is particularly effective for continuous, high-dimensional problems (W. Deng et al., 2021). It utilizes mutation and crossover operators to iteratively improve candidate solutions, with its performance critically dependent on the choice of control parameters such as mutation and crossover rates . Although DE is robust in exploring the search space, it is vulnerable to premature convergence if these parameters are not optimally set (S. Li et al., 2020). Evolutionary Programming (EP) simulates the evolution of numerical parameters within fixed program structures, rather than evolving the structure itself (Zhan et al., 2022). Originally used for evolving finite-state machines, EP now addresses a variety of optimization tasks by iteratively mutating and selecting candidate solutions. Its simplicity and ability to handle diverse data representations make it versatile (Slowik & Kwasnicka, 2020), though it typically demands high computational resources and careful experimentation to refine parameters. Evolutionary Strategies (ES) apply principles of natural selection and self-adaptation to iteratively refine candidate solutions (Slowik & Kwasnicka, 2020). ES focuses on mutation and selection to adjust distribution parameters dynamically, which makes it well-suited for black-box optimization problems where gradient information is unavailable (Y. Wang et al., 2021). However, the method is computationally intensive and sensitive to parameter tuning, with some variants (like Covariance Matrix Adaptation) still lacking a complete theoretical foundation. Genetic Algorithms (GA) mimic biological evolution through operations such as selection, crossover, and mutation applied to a population of candidate solutions (Katoch et al., 2021). Widely used in both single- and multi-objective optimization, GA effectively explores large and complex search spaces. Despite its flexibility and robustness, GA can suffer from slow convergence in high-dimensional spaces and may require additional strategies (e.g., tournament or ranking selection, elitism) to mitigate premature convergence and handle constraints (Z. Wang & Sobey, 2020). Genetic Programming (GP) extends the GA framework to evolve entire computer programs or tree-like structures, which enhances its capability to handle problems with undefined or highly complex solution spaces (Lin et al., 2020). By evolving the structure of candidate solutions rather than merely adjusting parameters, GP automatically detects features and models of nonlinear relationships. However, the increased flexibility comes at the cost of higher computational demands and sensitivity to parameter settings, which can lead to convergence on suboptimal solutions (Zhang et al., 2021).

Ant Colony Optimization (ACO) is inspired by the foraging behavior of ants, which deposit pheromone trails to communicate and reinforce promising paths (Fidanova, 2021). In ACO, solutions are constructed based on pheromone intensity and heuristic information, making it particularly effective for discrete problems such as routing and scheduling. The algorithm's performance is contingent on meticulous parameter tuning to balance exploration and exploitation, and it may require significant computational resources for large problem instances (Skinderowicz, 2022). Artificial Bee Colony (ABC) algorithm models the cooperative foraging behavior of honeybees by dividing the swarm into employed, onlooker, and scout bees (Alaidi et al., 2021). Each bee explores potential solutions and shares information about high-quality regions in the search space. While the algorithm is relatively simple to implement and effective for both continuous and combinatorial problems, its success is sensitive to parameter settings and may require numerous iterations to converge, especially in rugged landscapes (Kaya et al., 2022). Artificial Fish Swarm Algorithm (AFSA) is a swarm intelligence method that simulates the social behaviors of fish, such as foraging, swarming, and trailing, to navigate the search space. Each artificial "fish" evaluates its position and moves toward areas of higher "food concentration" (i.e., better solutions) (Pourpanah et al., 2023). AFSA is noted for its fast convergence and robustness across various domains, yet it can be prone to premature convergence and high computational complexity when dealing with very complex optimization problems (Zhao et al., 2023). Bacteria Foraging Optimization Algorithm (BFOA) draws inspiration from the chemotactic behavior of bacteria like E. coli. It simulates processes such as swimming, tumbling, reproduction, and elimination-dispersal to explore the search space (Guo et al., 2021). BFOA's biologically motivated framework provides a unique approach to global optimization, often outperforming traditional methods in certain applications. However, its performance is highly dependent on parameter settings, and like many bio-inspired methods, it may require hybridization with other techniques to enhance its robustness and convergence speed (Rahkar Farshi & Orujpour, 2021). Bat Algorithm (BA) is a nature-inspired metaheuristic that simulates the echolocation behavior of bats. Bats adjust their positions based on echo feedback while searching for the solution space. The algorithm initializes bat parameters, updates the best global solution, and incorporates randomness to balance exploration and local exploitation (Agarwal & Kumar, 2022). Its straightforward implementation and adaptability to both continuous and binary problems have led to applications in feature selection, fault diagnosis, and antenna positioning . However, BA may experience slow convergence and can become trapped in local optima if not properly enhanced through adaptive strategies (Cui et al., 2019).

Cuckoo Search (CS) is a metaheuristic inspired by the brood parasitism of cuckoo birds. Each solution is analogous to a cuckoo egg, and the algorithm relies on three main rules: egg laying (generating new solutions), nest selection (choosing higher-quality solutions), and nest abandonment (replacing poor solutions) (Cuong-Le et al., 2021). This process allows CS to explore the search space effectively. Although it has proven versatile in applications ranging from scheduling to structural design, CS is sensitive to parameter settings, which can affect its ability to consistently locate the global optimum (Jawad et al., 2023). Firefly Algorithm (FA) mimics the flashing behavior of fireflies, where the brightness of each firefly represents the quality of the solution. Fireflies are attracted to brighter peers, and their movement is influenced by both the attractiveness (a function of distance and light intensity) and random perturbations (Kumar & Kumar, 2021). This dual mechanism facilitates both exploitation and exploration in the search space. FA has been successfully applied in engineering design, neural network training, and feature selection; however, it may face challenges with scalability and convergence speed on large, complex problems (J. Wu et al., 2020). Glowworm Swarm Optimization (GSO) is based on the luminescence behavior of glowworms. Each agent (glowworm) carries a luciferin value that quantifies its fitness. Glowworms dynamically adjust their decision range toward neighbors with higher luciferin levels, leading to the formation of clusters around multiple optima (Cao et al., 2021). This makes GSO particularly useful for multimodal optimization problems. Nonetheless, the algorithm requires careful calibration of its parameters (e.g., luciferin decay and neighborhood range) to ensure robust performance across different problem domains (M. A. S. M. Shahrom et al., 2023). Grey Wolf Optimizer (GWO) is inspired by the social hierarchy and hunting strategies of grey wolves. The algorithm models the leadership hierarchy with alpha, beta, delta, and omega wolves, using mechanisms that simulate encircling and attacking prey to guide the search (Al-Tashi et al., 2020). With only a few tunable parameters, GWO effectively balances exploration and exploitation. It has been applied to diverse problems such as neural network training and economic dispatch in power systems (Seyyedabbasi & Kiani, 2021). Despite its simplicity, the optimizer may encounter difficulties in highly complex or multimodal landscapes, where additional mechanisms might be necessary to avoid local optima (Nadimi-Shahraki et al., 2021).

Lion Optimization Algorithm (LOA) is inspired by the social behavior and territorial dynamics of lions. The algorithm partitions the population into prides and nomads, mimicking real-world behaviors like cooperative hunting, mating, and cub rearing (Yazdani & Jolai, 2016). Through iterative processes involving fitness evaluation, crossover, mutation, and territorial shifts, LOA maintains diversity and explores the solution space comprehensively. While its natural metaphor offers innovative search mechanisms, the algorithm’s convergence speed and sensitivity to parameter tuning remain areas that require careful attention for optimal performance (J. Liu et al., 2020). Monkey Algorithm (MA) is modeled on the behavior of monkeys navigating rugged terrain. It begins with a random population of solutions and employs a multi-phase strategy: climbing to improve local solutions, “Watch-Jump” to escape stagnation, and “Somersault” to introduce diversity by pivoting around a reference point (Y. Li et al., 2022). This combination of intensification and diversification enables MA to escape local optima. However, the algorithm may become computationally expensive in high-dimensional settings and typically demand meticulous parameter tuning to achieve robust performance (Zedan Shaban & Natheer Alkallak, 2021). Particle Swarm Optimization (PSO) is inspired by the collective behavior observed in flocks of birds and schools of fish. Each particle represents a potential solution that adjusts its velocity based on its own best experience (cognitive component) and the swarm’s best-known position (social component) (Gad, 2022). Variants of PSO have been developed to address challenges such as premature convergence and to adapt to multi-objective contexts. While PSO is noted for its fast convergence and simplicity, its heavy reliance on information exchange can lead to stagnation in local optima, particularly in complex or high-dimensional problems (Shami et al., 2022). Shuffled Frog Leaping Algorithm (SFLA) combines the benefits of local search with global information exchange. The algorithm partitions a population of “frogs” into memeplexes, each conducting local searches by leaping toward better solutions (Y. Liu et al., 2022). Periodic shuffling among memeplexes enables global sharing of information, which enhances overall search performance. SFLA has proven effective in combinatorial and continuous optimization problems, but its performance is sensitive to the tuning of its parameters, and achieving an ideal balance between local intensification and global diversification can be challenging (Maaroof et al., 2022).

2.6 Applications of TSP-Optimization Algorithms in Recent Years

S. Wang et al. (2020) focuses on addressing the energy minimization TSP by developing the precise and verifiable methods for its solution. It examines control factors essential for developing new methods and compares them to existing approaches. The study demonstrates the superiority of exact methods over other relaxation methods in tackling the TSP. It also introduces a new method based on Lagrangean relaxation for constructing solution objectives in the energy minimization TSP. The results reveal the difficulty of solving the energy minimization TSP, with Constraint Programming Linear Programming with Extensions (CPLEX) failing to complete cases for all points, while a BB algorithm shows promise in achieving solutions within a shorter runtime. However, even with BB, reaching a satisfactory solution remains challenging due to the significant gap between the best solution found and the optimal solution. The paper highlights the potential impact of additional decision variables and constraints on the energy minimization TSP problem. It suggests that implementing the latest BB algorithm could reduce calculation times for factory processes, thereby improving efficiency. Baniasadi et al. (2020) employed on the Clustered Generalized Traveling Salesman Problem (CGTSP), an extension of the classical TSP that involves dividing nodes into clusters and subclusters. It aims to provide an efficient solution method for CGTSP to make it applicable in practical scenarios, particularly in modern logistics like Automated Storage and Retrieval Systems (ASRS) and drone-assisted Parcel Delivery Services (PDS). The paper introduces a transformation process that converts CGTSP into a constrained TSP and then into a classical TSP, highlighting the flexibility of CGTSP in modeling compared to traditional TSP. This transformation is crucial in demonstrating the potential of CGTSP in addressing real-world logistics challenges. An efficient solution method for CGTSP is meticulously developed, showcasing its superiority over existing methods in terms of solution quality and scalability. Comparative analyses between exact and heuristic solutions obtained through the proposed transformation technique and an IP formulation of CGTSP provide insights into the effectiveness of the proposed method relative to other solution approaches. The paper also outlines potential future research directions for CGTSP, including exploring other cluster-based variations of TSP, investigating constrained TSP formulations, and refining TSP heuristics tailored to the transformed problem structure.

Panwar & Deep (2021) introduces a novel discrete GWO algorithm designed to solve complex discrete TSP. This algorithm integrates the 2-opt algorithm to enhance its performance. The results indicate that the discrete GWO algorithm significantly outperforms alternative approaches. The significance of combinatorial optimization problems is discussed, emphasizing their prevalence and applicability across diverse fields. The TSP is highlighted as a particularly challenging combinatorial optimization problem and a benchmark for algorithm testing. The paper provides an overview of metaheuristic methods and their advantages in addressing combinatorial optimization problems efficiently. Experimental results of the discrete GWO algorithm across different TSP instances are presented, demonstrating its superior performance. Statistical analysis, comparative evaluations with other algorithms, and visualization of results support the effectiveness of the discrete GWO algorithm. The paper highlights its potential for efficiently solving routing problems and complex discrete optimization problems in the future. Additionally, related work is discussed, including applications of the GWO algorithm in engineering problems. Jong et al. (2022) focuses on identifying potential HES in Sarawak and proposes a hybrid AI approach for optimizing transmission line routing to these sites. Using raw data from Sarawak Energy Berhad (SEB), 155 HES are identified and categorized into six districts. A two-stage complex data management approach, including a new spatial mapping technique using GIS spatial tools, is employed to accurately map the data. The proposed hybrid AI approach consists of two parts: the first part employs TSP-GA, while the second part integrates improved fuzzy logic with TSP-GA. The research demonstrates that this hybrid approach significantly improves transmission line routing efficiency compared to conventional methods. Specifically, it achieves improvements ranging from 1.54 % to 18.01 % across different districts in Sarawak. By integrating new HES locations into the system, the research aims to enhance electricity accessibility in remote areas.

Di Placido et al. (2022) explores the Close-Enough TSP (CETSP), a variant of the classical TSP focusing on finding the shortest tour traversing all neighborhoods of a given set of targets. It introduces a novel approach utilizing a GA enhanced with two local search techniques and a mathematical model to address this challenge. A notable feature of this GA is its incorporation of dynamic population sizing and elitism, tailored specifically for the CETSP. Through a comprehensive performance comparison against state-of-the-art heuristics commonly used for similar optimization tasks, the research demonstrates that the GA consistently outperforms existing methods, achieving the smallest percentage gap with respect to the best-known solution. The paper introduces two innovative metrics for classifying problem instances based on their characteristics and their impact on the difficulty of solving the problem. Additionally, it showcases a real-world application of the proposed GA in solar panel diagnostic reconnaissance. Specifically, the GA is used to identify optimal routes for drones conducting reconnaissance flights over photovoltaic fields. By minimizing route length, the GA facilitates more efficient detection of solar panel performance issues, highlighting its practical relevance in addressing complex routing problems in the context of solar panel maintenance. Furthermore, the paper contextualizes its contribution within the broader landscape of research on the TSP and its variants. It emphasizes the significance of the CETSP and the relevance of the proposed GA in advancing solutions for such complex optimization problems. A critical issue of dynamic vehicle shadows disrupting pavement PV arrays, leading to power losses and complex output characteristics was addressed by Mao et al. (2023). To tackle this challenge, it introduces a novel reconfiguration strategy grounded in the TSP framework and SA algorithm. Implementing this strategy results in significant improvements in pavement PV array performance metrics, including reducing local maximum power points, enhancing global maximum power output, and increasing overall power generation efficiency despite real-world shading complexities. A comprehensive comparison with existing techniques highlights the superiority of the TSP-based approach in maximizing output power and achieving irradiance equalization across the PV array. Experimental validation confirms the efficacy of the proposed approach in enhancing maximum power output and mitigating multiple peaks in PV curves, particularly under dynamic vehicle shadowing conditions. Comparative assessments between traditional and proposed configurations under diverse shading scenarios provide compelling evidence of the latter achieving superior performance in optimizing output power.

A novel labeling method was introduced by Tawanda et al. (2023), aimed at solving the TSP, a classic optimization challenge where a finite set of nodes must be visited exactly once, minimizing the total weight of connecting arcs. This method is designed to find the optimal solution within a predetermined number of iterations, tailored to networks of known sizes. It offers flexibility by addressing both symmetric and asymmetric TSP variants and demonstrates computational efficiency, with complexity decreasing as iterations progress. The labeling algorithm allows for the determination of alternative tours within a TSP network, terminating after a fixed number of iterations proportional to the total number of nodes. Numerical analysis has showcased the labeling method outperforming existing algorithms in finding optimal tours within a reduced number of iterations. Future studies outlined by the authors include the development of software to facilitate computational experiments on large-scale TSP instances and comparative analyses with existing methods. This comprehensive approach aims to further validate the effectiveness of the proposed method and its potential applications in real-world optimization problems. Mahmoudinazlou & Kwon (2024) proposes a hybrid GA to address the min-max multiple TSP. The hybrid GA integrates TSP sequences and a DP algorithm to find optimal solutions. It introduces a novel crossover operator aimed at combining similar tours from parent solutions while addressing intersections between tours. Experimental results show that the hybrid GA outperforms existing algorithms, matching, or surpassing best-known solutions in most cases. It particularly improves best-known solutions in several instances and consistently demonstrates equal or superior average performance across a wide range of problems. The paper suggests potential extensions of the hybrid GA to handle multi-depot multiple TSP and explore the integration of drones as agents in the multiple TSP. Additionally, it highlights the importance of heuristic algorithms like GA for addressing NP-hard problems such as the multiple TSP due to the impracticality of exact solutions.

A novel solution was developed by Parlangeli et al. (2024) for the shortest Dubins path problem, specifically focusing on finding the shortest TSP path between three consecutive via-points with prescribed initial and final orientations, and no prescribed orientation at the intermediate point. The solution utilizes simple tools from analytic geometry and presents an efficient algorithm for real-time path planning, which is validated through extensive simulations. The paper emphasizes the importance of robust algorithms for autonomous navigation, particularly in marine environments, and highlights the significance of the Dubins vehicle model for describing vessel motion constraints. It provides a detailed mathematical formulation of the problem and the solution approach, including the computation of tangent points and the derivation of the optimal path length and heading at the via-point. The proposed method outperforms existing approaches in terms of computational complexity and solution accuracy, as demonstrated through comparative analysis and performance evaluation metrics. The paper concludes by discussing the potential applications of the proposed method in autonomous marine or underwater vehicles and dynamic scenarios requiring real-time path planning. Chaves et al. (2024) addresses the Family Traveling Salesman Problem (FTSP) and proposes two methods for its solution: a parallel BB algorithm with efficient local search and an adaptive metaheuristic combining the Biased Random-Key GA with Q-Learning (BRKGA-QL) algorithm. Computational experiments on a benchmark dataset of 185 instances reveal that the BB algorithm with efficient local search was optimal for 179 instances and improved upper bounds in 19 open instances, while BRKGA-QL found optimal solutions in 131 instances and improved upper bounds in 21 open instances. The paper emphasizes the robustness and efficiency of both methods, comparing them with existing literature and introducing the concept of using reinforcement learning to control BRKGA parameters. Computational experiments were conducted on a system with specific configurations and solver settings, showcasing the performance profiles of the methods. Additionally, the paper discusses local search heuristics, their computational efficiency, and the impact of various method components. Future research directions include addressing variant FTSP and analyzing solutions in new instances, potentially from e-commerce data.

A prevalent limitation across multiple works (Baniasadi et al., 2020; Chaves et al., 2024; Di Placido et al., 2022; Mahmoudinazlou & Kwon, 2024; Mao et al., 2023; Panwar &

Deep, 2021; Parlangeli et al., 2024; Tawanda et al., 2023) is their restriction to singular objective functions, limiting applicability to real-world multi-criteria problems. Exceptions such as Jong et al. (2022), which integrated multi-objective optimization for energy systems, still face critiques for omitting critical parameters (e.g., lightning resilience). Strengths include metaheuristic advancements to escape local optima (Di Placido et al., 2022; Mao et al., 2023), flexibility in handling symmetric/asymmetric TSP (Baniasadi et al., 2020; Tawanda et al., 2023), and scalability for large instances (Baniasadi et al., 2020; Chaves et al., 2024). However, gaps persist in cross-algorithm benchmarking, swarm intelligence integration, and validation on ultra-large-scale problems. Technical innovations, such as novel crossover operators (Mahmoudinazlou & Kwon, 2024) and geometric path simplifications (Parlangeli et al., 2024), remain niche due to limited comparative analyses. Collectively, literature highlights the need for frameworks that harmonize multi-objective optimization, scalability, and robust empirical validation to address complex, real-world TSP variants.

2.7 Establishments of Real-Time Monitoring and Control Using IIoT

Chinomi et al. (2017) proposes the design and implementation of a smart monitoring system tailored for modern renewable energy micro-grid systems, employing a low-cost data acquisition system and LabVIEW program. This prototype facilitates the monitoring, analysis, and communication with devices within the micro-grid, measuring diverse parameters including voltage, current, power, power factor, and harmonics distortion. It addresses the limitations of traditional measuring instruments, which are deemed inadequate for remote monitoring and fault detection. The proposed system endeavors to overcome these shortcomings by automatically storing data, enabling remote control, and providing qualitative data display alongside quantitative measurements. LabVIEW software serves as the backbone of the system, featuring a front panel for user interaction and a block diagram for program control. National Instruments (NI) compact Data Acquisition (DAQ) hardware interfaces with external signals, constituting the hardware configuration. The hardware configuration encompasses sensors and DAQ devices for monitoring solar and wind plants, as well as load and storage systems. The LabVIEW program is engineered to measure, monitor, analyze, and display data from these renewable energy systems, offering signal simulation, recording, and real-time fault detection capabilities. Experimental findings validate the accuracy and efficiency of the proposed system in measuring and analyzing renewable energy system parameters. Verification processes include tests for tolerances, phase angle calibration, and total harmonic distortion analysis. Demonstrating proficiency under both normal and fault conditions, the system enables remote monitoring and control. Emphasized advantages include automatic parameter recording, accurate measurement and analysis, and remote-control capabilities, rendering it more cost-effective and efficient compared to traditional devices. However, the proposed system utilized a radial network simulation model, and it has limited control. A thorough investigation into an IIoT-based energy monitoring system tailored for real-time control and surveillance of energy consumption within a switchgear industry was presented by Mudaliar & Sivakumar (2020). Central to this system is the utilization of Raspberry Pi, programmed with Node.js, to gather data from energy meters and store it locally, enabling access through Grafana. A comparison of various IIoT devices has been made, ultimately favoring Raspberry Pi and Arduino due to their simplicity, user-friendliness, and cost-effectiveness. Technical specifics of the system setup, including the utilization of Raspbian OS, InfluxDB for local and cloud-based data storage, and Grafana for data visualization, are applied. Furthermore, the findings through graphical representations of various electrical parameters obtained from existing energy meters are displayed using Grafana. These results provide tangible evidence of the efficiency of the system in facilitating the understanding of day-to-day energy consumption patterns and enabling effective energy conservation measures. Additionally, it provides detailed guidance on the setup procedures for implementing a Raspberry Pi-based monitoring system and acknowledges the invaluable support received from industry personnel in obtaining operational data for the study. However, it lacks a simulation model, and the system is used for monitoring purposes only.

Md Liton Hossain et al. (2020) introduces advancements in Wind Energy Conversion Systems (WECS) utilizing Multi-Level Inverters (MLI), introducing a simplified Proportional Integral-based Space Vector Pulse Width Modulation (PI-SVPWM) to address output waveform ripples and voltage balancing issues. Additionally, a real-time fault detection algorithm integrated into the PI-SVPWM controller is proposed, using an industrial IIoT algorithm and hardware prototype for real-time condition monitoring of WECS. The proposed simplified PI-SVPWM aims to mitigate output waveform ripples and voltage imbalances, while the embedded fault detection algorithm enhances converter reliability. The IIoT algorithm and hardware prototype enable real-time condition monitoring of WECS, emphasizing fault diagnosis, condition assessment, and asset management. Through remote access and periodic assessments, the proposed algorithm facilitates reliable condition monitoring. Contributions include the development of the PI-SVPWM controller, the embedded fault detection algorithm, and the IIoT algorithm and hardware prototype. These advancements are poised to enhance efficiency, reliability, and real-time monitoring of wind energy conversion systems. However, the development of PI-SVPWM is complex, and there is an absence of a simulation model. Gupta et al. (2021) introduces a low-cost, IIoT-enabled data acquisition system designed for monitoring solar

PV systems in harsh environments. It addresses the limitations of existing wired and wireless systems, emphasizing their high cost and limited accessibility. The proposed system utilizes open-access software and cloud services, enabling remote monitoring and data gathering of PV systems. After testing over 28 days in harsh conditions, the system proves to be reliable, cost-effective, and energy-saving. It offers a 58 % energy saving, and increased sensor life compared to traditional systems. The remote accessibility facilitates monitoring from any location, making it suitable for real-time performance optimization. The paper highlights the significance of cost-effective monitoring systems for PV systems and real-time data acquisition for performance evaluation. The proposed system provides a platform for long-term data collection and analysis, contributing to the advancement of IIoT technology in the solar energy sector. The IIoT-enabled data acquisition system offers an economical and efficient solution for monitoring and optimizing PV system performance. It presents a valuable tool for researchers and academia, providing a more affordable and reliable option for sensing and monitoring PV systems in diverse environments. However, the paper is limited to hardware development with monitoring features only, and a simulation model is absent.

A practical implementation of IoT technology in managing household electricity, specifically focusing on the creation of an IoT Smart Household Distribution Board (ISHDB) for monitoring and controlling various smart appliances was introduced by Ahmed et al. (2021). The ISHDB collects and stores voltage, current, and power data, presenting them in a user-friendly manner. It utilizes an Arduino-based prototype, Wi-Fi connectivity to the ThingSpeak cloud, and the Blynk mobile application for real-time monitoring. Cost and time comparisons with existing techniques reveal the superiority of ISHDB in terms of cost-effectiveness and execution time. The paper highlights the growing need for smart home systems to manage electrical energy efficiently, especially with the rising energy demand and transition to prosumer-oriented markets. It emphasizes the significance of cost-effective prototypes for distribution boards, an aspect often overlooked in existing literature. The ISHDB system comprises two core modules: a hardware interface module and a software communication module, utilizing an Arduino Uno microcontroller. Testing with household appliances demonstrates its effectiveness in monitoring electricity usage and controlling power consumption. Safety and protection measures integrated into the system ensure compliance with regulations and accurate signal measurement. The paper suggests further research to extend the functionalities of the system, such as incorporating IoT-aided AI methods for security purposes. Overall, it demonstrates the technical soundness and cost-effectiveness of the ISHDB system for household applications, enabling real-time monitoring of appliance performance from anywhere. However, the developed application is only suitable for household use. Ali et al. (2021) proposes the monitoring, evaluation, and management of the operational performance of an existing micro-grid comprising grid PV systems and the main grid supply. It introduces a customized web-based SCADA system designed to assess the performance of the micro-grid in terms of energy consumption, power quality indices, and energy cost based on the energy tariff in Egypt. The designed system facilitates micro-grid monitoring and load sharing between the main grid supply and the off-grid PV energy system, with online data access and analysis from the inverter web server, multi-level security authentication, and data encryption. Smart wired and wireless technology are utilized for integrated sensing devices and validate the collected data from the SCADA system through performance characterization of the on-grid PV system. Furthermore, other research works based on monitoring and analyzing the collected data in a centralized manner where the operator should exist in front of the SCADA master station, typically in the control room, are analyzed. The developed platform emphasizes the key advantages including cost-saving, operational sustainability, convenience, security, extendibility, and safe resource management. However, the proposed system lacks a simulation model with limited control functionalities.

A development and implementation of an advanced energy management strategy for a hybrid microgrid, aiming to address challenges posed by intermittent GERs was proposed by Ullah et al. (2022). It introduces an energy management system with a real-time monitoring interface to efficiently manage the hybrid microgrid, comprising solar and wind power sources, Li-ion battery storage, backup electrical grids, and AC or DC loads. The energy management system ensures balanced power supply, stable frequency, and voltage profiles, utilizing efficiency control for battery charging and discharging. Simulation results using MATLAB Simulink and Python platforms validate the effectiveness of the proposed energy management system and monitoring interface for stable microgrid operation. It categorizes microgrid control systems into centralized and decentralized models for optimal operation. The proposed energy management system model and real-time monitoring interface aim to optimize energy management, ensure stability, and enhance reliability under varying meteorological conditions and load fluctuations. Performance evaluations demonstrate the effectiveness of energy management systems in managing power balance, frequency regulation, and voltage control within the microgrid. However, the simulation model used a radial network simulation model and lacked validation of the proposed system. Qays et al. (2022) introduces a novel application of SCADA systems using IIoT technology for monitoring hybrid renewable energy systems, which include photovoltaic, wind, and battery energy storage systems. It emphasizes the increasing demand for real-time monitoring in remote and offshore locations. The proposed SCADA system enables real-time monitoring of electrical parameters and remote control of system components. To develop the hardware prototype, the authors utilize low-cost electronic components and an Arduino Integrated Development Environment ATMEGA2560 remote terminal unit. Simulation and experimental results demonstrate the feasibility, reliability, and cost-effectiveness of the proposed system compared to existing techniques. The proposed IIoT-aided SCADA system is shown to be a novel and effective solution, validated through simulation and experimental analyses. Structural diagrams of the hybrid power system, interfacing of the SCADA system with MATLAB Simulink, and the experimental prototype are presented. Comparison with existing models indicates the novelty and potential of the proposed system for remote renewable energy management. However, the proposed simulation model is in a radial topology network and lacks integration of dynamic inputs.

Melo et al. (2023) focuses on designing and assessing a standalone microgrid for San Andres, Colombia, aiming to provide low-cost clean electricity using renewable resources. It addresses challenges related to regulating frequency and voltage within the microgrid due to the intermittent nature of GERs and presents a dispatch strategy-based control system. A 24.57 kW peak load microgrid composed of a PV-wind-storage system is designed and assessed using MATLAB Simulink. Additionally, a Programmable Logic Controller-SCADA (PLC-SCADA) system with a Human-Machine Interface (HMI) in C# is developed for real-time monitoring and automatic data uploading to a cloud database. The paper highlights the negative effects of fossil fuel consumption on the environment and human health, advocating for the replacement of conventional energy sources with renewable ones. A comparative study of five dispatch strategies evaluates their performance based on cost, energy consumption, and emissions, finding the Load Following (LF) strategy optimal for the microgrid. Findings include optimal component sizes, microgrid system stability, and voltage, power, and frequency responses under different dispatch strategies. Development and testing of PLC-SCADA system, SCADA HMI, and cloud database for data storage and analysis are also presented. However, it lacks hardware validation, and only a radial network is used in the simulation model. The challenges in power distribution networks with the rise of renewable energy-based microgrids and substations lacking real-time monitoring was investigated by Ullah et al. (2023). It proposes an IIoT-based monitoring and control system for power substations and distributed smart grids to tackle suboptimal resource allocation, poor load management, grid instability, and lack of real-time decision-making capabilities. By exploring IIoT technology for power parameter monitoring and load management across various sectors, including industrial, domestic, commercial, and electric vehicles, it can mitigate power fluctuations and contingencies. Using HOMER Grid, it analyzes the annualized power production pattern of smart grids and the power consumption pattern of integrated loads for proactive energy management decisions, ultimately aiming to reduce energy costs and carbon emissions. The proposed model is validated through a constructed prototype, demonstrating real-time monitoring and control capabilities to enhance grid stability and energy efficiency. An IIoT-based monitoring and control system for power substations and associated smart grids is introduced to facilitate effective segregation decisions into the power distribution network. By segregating smart grids and managing loads, the proposed system mitigates suboptimal resource allocation and grid instability. Through HOMER Grid analysis, the research investigates power production and consumption patterns, enabling proactive energy management decisions. IIoT technology is leveraged for power parameter monitoring across substations and smart grids, ensuring stable operation and contingency mitigation. Validation through a prototype demonstrates real-time monitoring and control capabilities for effective energy and load management decisions. However, the system lacks a simulation model for further planning and evaluation.

The integration of smart grid technology with renewable energy resources using IIoT, emphasizing the use of Wireless Sensor Network (WSN) for power grid monitoring and control was explored by Murugan & Vijayarajan (2023). It addresses issues such as voltage violation and grid instability by proposing a hybrid approach of encryption methods and algorithms for secured data transmission. The research utilizes Advanced Encryption Standard (AES) and Rivest Cipher 4 (RC4) for encryption, coupled with the Pulse Coupled Neural Network (PCNN) algorithm for high-speed data transmission. Hardware implementation involves Arduino controllers, facilitating efficient smart grid parameter monitoring for uninterrupted power availability. Addressing the limitations of traditional microgrid systems, it highlights the significance of integrating renewables to enhance system efficiency. The Remora Optimization algorithm is introduced for efficient selection of neural network hyperparameters, ensuring global optimal solutions for identifying the shortest path. It demonstrates a hybrid encryption technique, PCNN algorithm, and Remora Optimization algorithm for secured and high-speed data transmission. Hardware implementation using Arduino controllers and MATLAB Simulink assesses system performance, indicating improved efficiency and secure data transmission. The research emphasizes the potential of the proposed IIoT-based secure data monitoring system for renewable energy-fed microgrids, offering valuable perspective into the development of efficient and secure smart grid technologies. Nonetheless, there is an absence of a simulation model for the proposed system. Krishna Rao et al. (2024) explores the utilization of IIoT and smart energy management systems for forecasting solar power generation. It emphasizes the significance of PV forecasting in enhancing real-time control systems, mitigating the impact of uncertainty on PV energy generation, and increasing solar power output. The integration of IIoT with a solar PV system shows potential for remote and in-person monitoring. The hardware and software components for monitoring and controlling solar power systems are provided, including microcontrollers, sensors, and IIoT for data transmission. Implementation of the smart energy monitoring system focusing on real-time data collection from remote locations and its implications for data analysis and system maintenance are analyzed. However, the system lacks a simulation model for further analysis.

2.8 Research Gap

Sarawak faces significant challenges in transitioning from fossil fuels to GERs. A crucial research gap exists due to the absence of a comprehensive assessment framework model for identifying optimal large-scale green energy locations across the state (Almasad et al., 2023). While existing studies focus on determining suitable GER types (Bilgili et al., 2022; Hasti et al., 2023; Sasikumar & Ayyappan, 2019) for specific regions or small areas, none offer a method to thoroughly evaluate potential GERs on a statewide scale (Jong & Ahmed, 2024). The scattered distribution of GERs in Sarawak highlights the significance of harnessing these resources on a large scale to foster sustainable energy development. To address this gap, a novel GIS-based approach integrating fuzzy TOPSIS, and filtration algorithms is proposed. One notable feature of the proposed model is the integration of novel filtration algorithms. These algorithms seamlessly interact with polygon layers and raster maps, facilitating the identification of optimal green energy locations based on decision-maker’s preferences. This enhancement ensures the adaptability of the proposed model to real-world scenarios.

Efficient integration of green energy locations into a bulk power system generation is essential for achieving a zero-carbon footprint and energy sustainability (Jong & Ahmed, 2024). However, reliable methods for GER integration are lacking, indicating another research gap. Robust methods are needed to integrate GERs efficiently, with a preference for the ring topology due to its resilience against power disruptions (Shakil et al., 2020). Three key parameters (distance, elevation difference, and average ground flash density) should be considered. Minimizing the distance between all green energy locations is able to reduce installation, operation, and maintenance costs (Jong et al., 2022). A minimal elevation difference is necessary, as flat or gently sloping terrain is preferred, simplifying the construction process and reducing the need for complex engineering solutions (Jong et al., 2022). Considering the average ground flash density is crucial for mitigating the risk of power line and equipment destruction, thus minimizing downtime and costly repairs. In 2021, lightning strikes on the 275kV Murum Junction transmission line resulted in a double-circuit tripping. This event had significant consequences, causing outages from Miri in the north to Kuching in the south (Sarawak Energy, 2021). Another incident unfolded in 2023 when an unexpected power interruption triggered by lightning strikes affected residents in Sibu (MARILYN TEN, 2023). To address these challenges, an improved GIS-driven fuzzy TSP-BIP algorithm model is proposed. This model designs the optimal power line routing for the identified green energy locations integration with minimum distance, elevation difference, and average ground flash density.

Furthermore, despite state-of-the-art research focused on monitoring, control, and automation for IGESs, a significant research gap remains in developing a comprehensive simulation and hardware model. Many researchers have proposed solutions using hardware without the simulation model (Krishna Rao et al., 2024; Murugan & Vijayarajan, 2023; Ullah et al., 2023). Some researchers have proposed the simulation models (Melo et al., 2023; Qays et al., 2022) but without incorporating dynamic real-time data from actual power utility companies and precipitation data to model the real behavior of power systems. Besides, none of them have considered a ring topology network for IGESs simulation modeling. Bridging this gap is crucial as it provides a more realistic, dynamic, and robust IGESs model, thereby enhancing planning, and management practices in the field. Table 2.1 highlights comparisons of the proposed innovative IIoT-based monitoring, control, and automation system for IGESs against other existing systems.

Table 2.1: Comparisons of the Proposed IIoT-based System with Existing Systems

Source	Simulation Model							Hardware Model			
	Real-Time Dynamic Data Model	**IGESs Model**	**Model Topology**	**Real-Time Dynamic Input to Model**	**Monitoring**	**Manual Control**	**Automation**	**SCADA Interfacing Hardware**	**Monitoring**	**Manual Control from SCADA**	**Automation**
(Chinomi et al., 2017)	No	Yes	Radial	No	Yes	No	No	Yes	Yes	No	No
(Mudaliar & Sivakumar, 2020)	No	No	No	No	No	No	No	Yes	Yes	No	No
(Md Liton Hossain et al., 2020)	No	No	No	No	No	No	No	Yes	Yes	No	Yes
(Gupta et al., 2021)	No	No	No	No	No	No	No	Yes	Yes	No	No
(Ahmed et al., 2021)	No	No	No	No	No	No	No	No	Yes	No	Yes
(Ali et al., 2021)	No	No	No	No	No	No	No	Yes	Yes	No	Yes
(Ullah et al., 2022)	No	Yes	Radial	No	Yes	No	No	No	No	No	No
(Qays et al., 2022)	No	Yes	Radial	No	Yes	Yes	No	Yes	Yes	Yes	No
(Melo et al., 2023)	No	Yes	Radial	No	Yes	Yes	Yes	No	No	No	No
(Ullah et al., 2023)	No	No	No	No	No	No	No	Yes	Yes	No	Yes
(Murugan & Vijayarajan, 2023)	No	No	No	No	No	No	No	Yes	Yes	No	Yes
(Krishna Rao et al., 2024)	No	No	No	No	No	No	No	Yes	Yes	No	No
Proposed System	**Yes**	**Yes**	**Ring**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

2.9 Chapter Summary

The chapter has examined significant criteria encompassing climatic, technical, accessibility, environmental, and social considerations for GERs. The review also analyzed the utilization of MCDM methods to identify optimal green energy locations. Subsequently, existing optimization algorithms used to integrate the identified green energy locations by designing optimal power line routing have been reviewed and evaluated. Additionally, state-of-the-art research works regarding real-time monitoring, control, and automation for IGESs have been explored. Comprehensive literature reviews have been conducted to determine existing research gaps that need to be addressed to ensure the successful implementation of effective green energy infrastructure in the region.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter begins by analyzing the influential criteria of SES, WES and HES. Influential criteria for identifying green energy locations from recent years (starting from 2020) are gathered and ordered based on the number of publications from reputable IEEE and Elsevier high-impact journals. This collection of criteria serves as the groundwork for identifying optimal green energy locations. The research then proposes a novel GIS-based fuzzy TOPSIS and filtration algorithms for identifying large-scale GERs in Sarawak. The GERs identification process comprises two layers. In the first layer, a filtration framework is employed, consisting of two phases. Initially, potential SES and WES from the DIVA-GIS database, and potential HES from the SEB database are gathered. These undergo the first filtration process to exclude locations overlapping with structured data. The optimal HES are then determined, and WES are selected with a minimum wind speed of 3 ms^{-1}. Subsequently, the second phase of filtration employs criteria constraints and raster maps to further refine potential SES locations. In the second layer, a proposed fuzzy TOPSIS algorithm is utilized to identify and rank the top 100 optimal SES. Validation against the weighted sum method is performed to determine its alignment level.

Following GERs identification, optimal power line routing is designed to integrate these green energy locations. Initially, these locations are clustered into 12 divisions. Three influential factors (distance, elevation difference, and average ground flash density) are considered to achieve overall minimal values using a fuzzy TSP-BIP algorithm. Validation is conducted to compare results between fuzzy TSP-BIP and ordinary TSP-BIP algorithms to demonstrate the advantages of fuzzy logic operation. Additionally, fuzzy TSP-BIP is compared with other fuzzy optimization algorithms to validate its superior performance.

Subsequently, IIoT-based real-time monitoring, control, and automation strategies for IGESs are proposed. Real-time dynamic data modeling incorporates load demand data from the grid system operator and solar radiation and temperature data from Solcast. IGESs sizing is based on load demand, and the ring topology network of the IGESs model is designed in MATLAB Simulink. The proposed IGESs model interfaces with dynamic input data, and the SCADA system communicates with MATLAB through a server for effective real-time monitoring, control, and automation. To validate the practical application of real-time monitoring, control, and automation strategies via SCADA in real-world scenarios, a hardware model utilizing Raspberry Pi alongside IIoT components has been interfaced with the SCADA system. This integration serves to demonstrate the effectiveness of SCADA by enabling real-time monitoring, control, and automation functionalities.

3.2 Research Framework

This research framework is organized into three principal sections. The first section focuses on the development of innovative GIS-based fuzzy TOPSIS and filtration algorithms designed to identify large-scale green energy locations. This section is structured into two distinct layers: the first layer is responsible for managing GIS data and implementing a double-phase filtration process, while the subsequent layer employs the proposed fuzzy TOPSIS algorithm to identify the optimal sites. The results derived from this approach are validated against the WSM. The second section introduces a novel GIS-driven fuzzy TSP-BIP algorithm aimed at effectively integrating the identified green energy sites. The performance of this algorithm is rigorously compared and validated against the conventional

TSP-BIP and other fuzzy TSP algorithms. The final section is dedicated to the establishment of an IIoT-based framework for real-time monitoring, control, and automation. This involves the development of dynamic input mechanisms, the formulation of an IGESs model, and the design of robust communication strategies. The practical applicability of these strategies is confirmed through validation with a hardware prototype, thereby demonstrating their potential in real-world scenarios. Figure 3.1 provides a visual representation of the overall research framework.

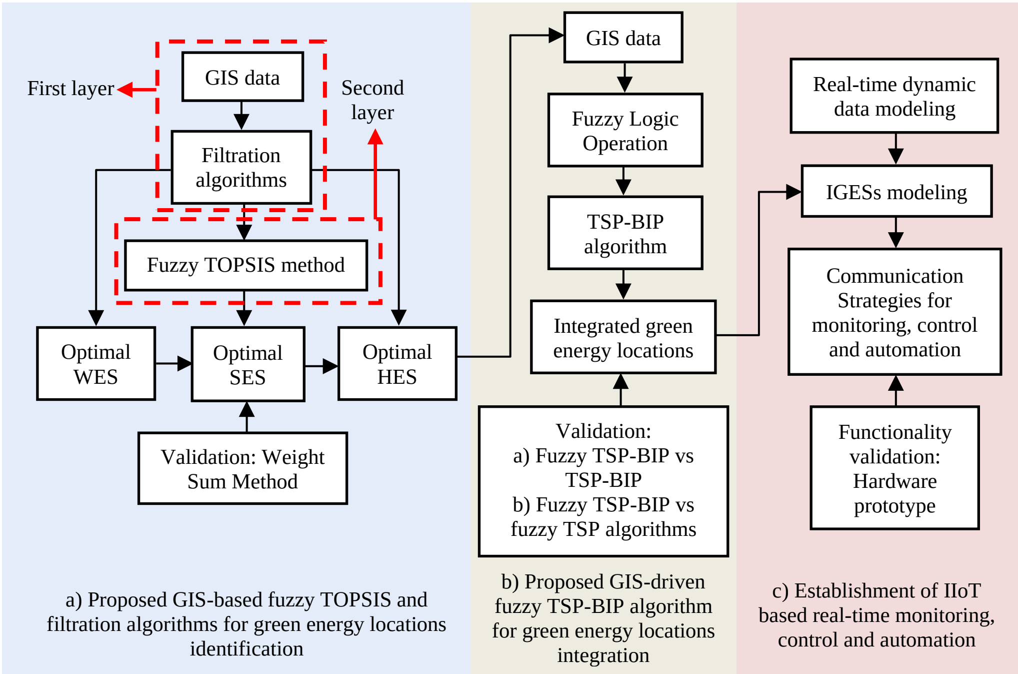


Figure 3.1: Schematic Overview of the Proposed Research Framework

3.3 Proposed GIS-based fuzzy TOPSIS and Filtration Algorithms

The research proposes a novel GIS-based fuzzy TOPSIS and filtration algorithms to identify the green energy locations. This method can effectively screen a large scale of potential green energy sites within a region. Figure 3.2 illustrates the complete flowchart of the proposed model for determining optimal green energy locations.

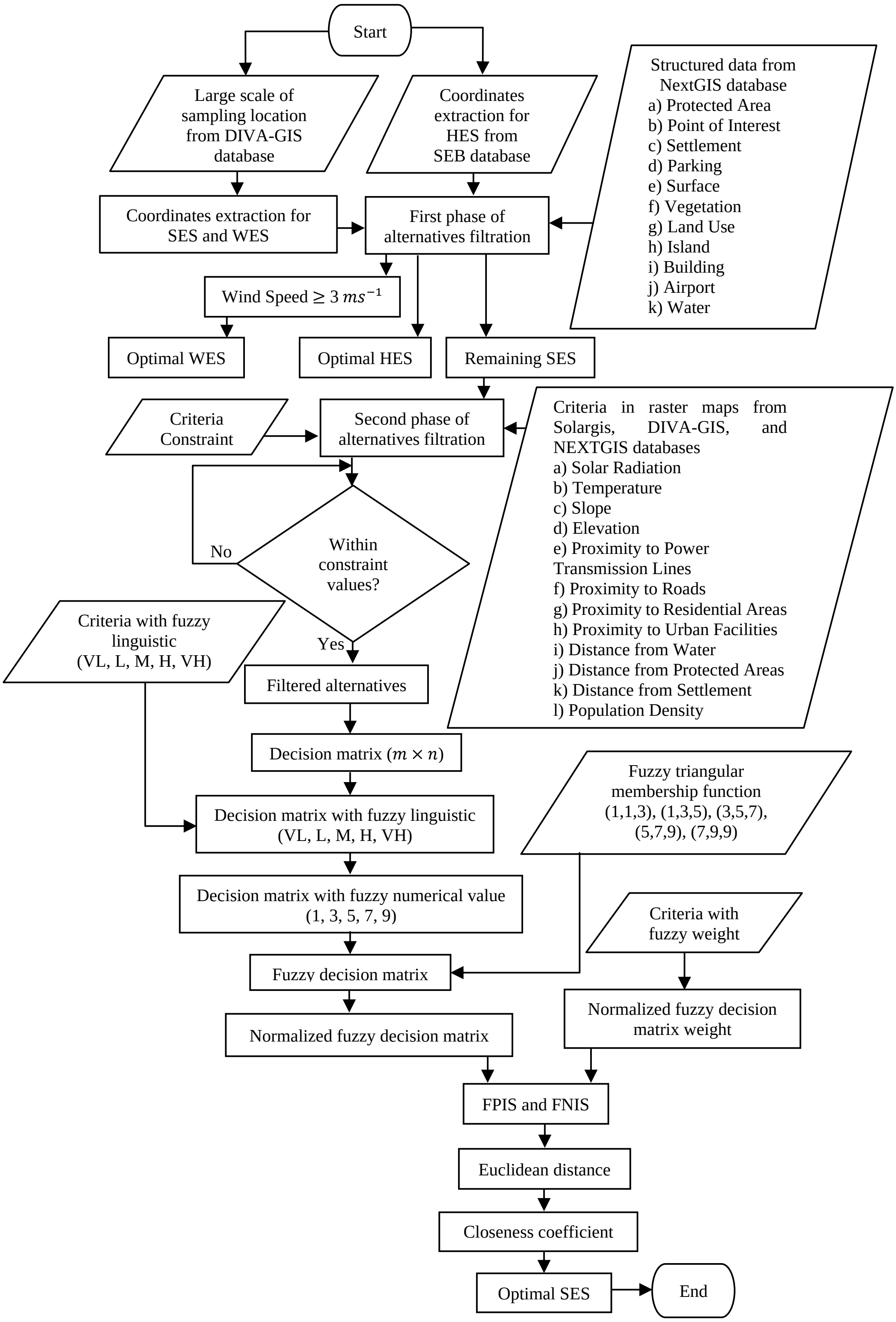


Figure 3.2: Flowchart of Proposed GIS-based Fuzzy TOPSIS and Filtration Algorithms

3.3.1 GIS Data Integration and Filtration Process

In the initial phase of the first layer, a comprehensive dataset comprising 19,237 coordinate-formatted locations is extracted from the DIVA-GIS database (DIVA-GIS, 2023). These locations encompass various small administrative units, including divisions, villages, towns, mukim, and others, which are all considered potential sites for SES and WES. Furthermore, data pertaining to HES is extracted from SEB mapping, yielding a total of 155 potential HES (Jong et al., 2022). Figure 3.3 illustrates the distribution of all SES and WES, while Figure 3.4 displays the potential HES across the Sarawak region.

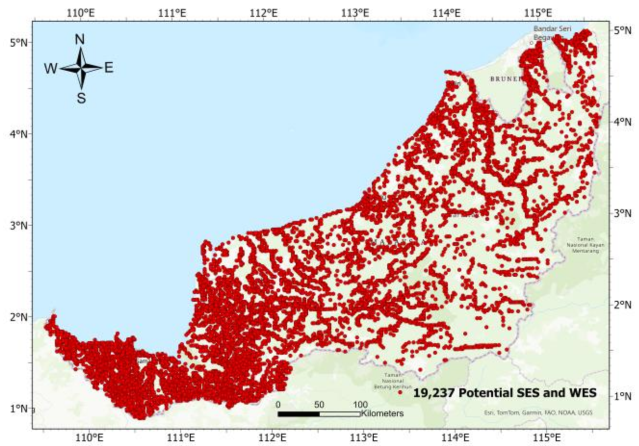


Figure 3.3: Potential SES and WES in Sarawak (DIVA-GIS, 2023)

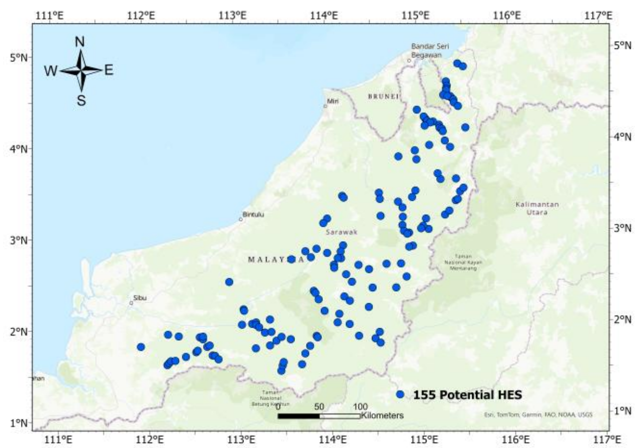


Figure 3.4: Potential HES in Sarawak (Jong et al., 2022)

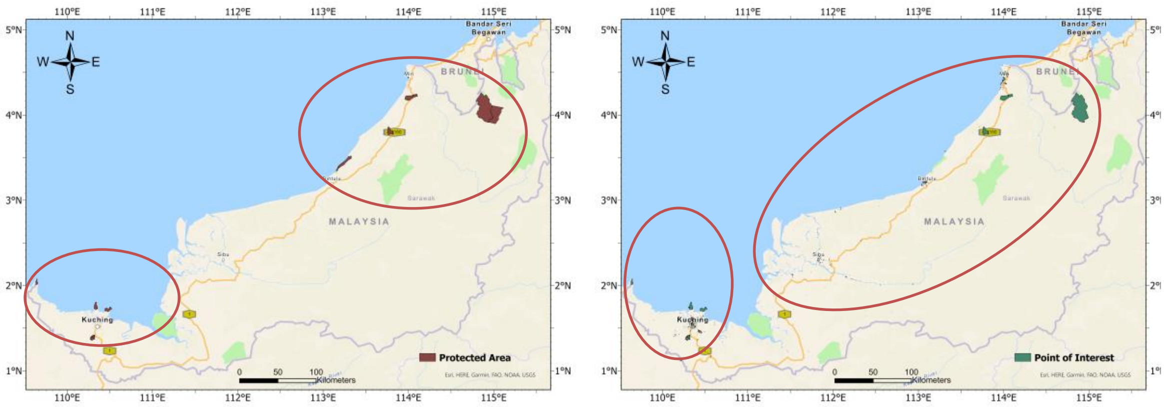
To address the scattered distribution of potential green energy locations across the Sarawak region, structured data play a crucial role in excluding locations within protected areas, points of interest, settlements, parking zones, surfaces, vegetation, land use zones, buildings, airports, and bodies of water. The description of this structured data is provided in Table 3.1, while Figures 3.5 (a) to (k) visualize the mapping of each data category. By excluding coordinates within these designated areas, it ensures that identified green energy locations are not situated in restricted or prohibited zones. Conversely, a specific polygon layer named "island" is employed for the filtration process, eliminating coordinates outside this polygon. This measure confines green energy locations within Sarawak's boundaries, thus mitigating potential land controversy issues. These data are then classified into two categories: exclusion and inclusion areas. Exclusion areas denote regions with restrictions or prohibitions on green energy development, while inclusion areas permit such installations. The utilization of structured data is essential in the first phase of the filtration process for refining and filtering the large scale of 19,237 potential SES and WES, and 155 potential HES within the region.

Table 3.1: Structured Data (NextGIS, 2023)

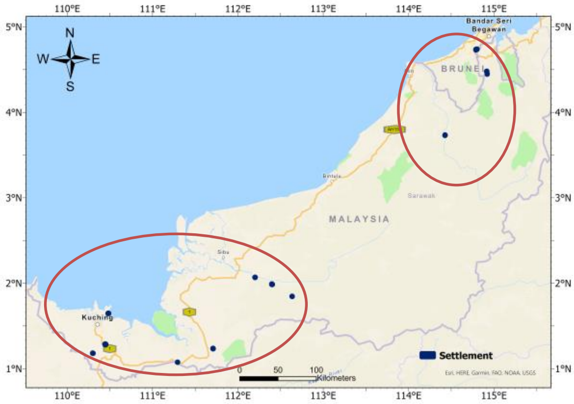
Structure	Description
Protected Areas	Reserved areas for recreational activities, such as parks, gardens, and beaches, protected from further development.
Point of Interest	Noteworthy locations including amenities, offices, shops, tourist attractions, and sports facilities.
Settlement	Inhabited areas where people reside and engage in various activities, including cities, towns, villages, and hamlets.
Parking	Diverse parking facilities, such as multi-story, park-and-ride, surface, underground, private, and designated for customers or official use.

Table 3.1 continued

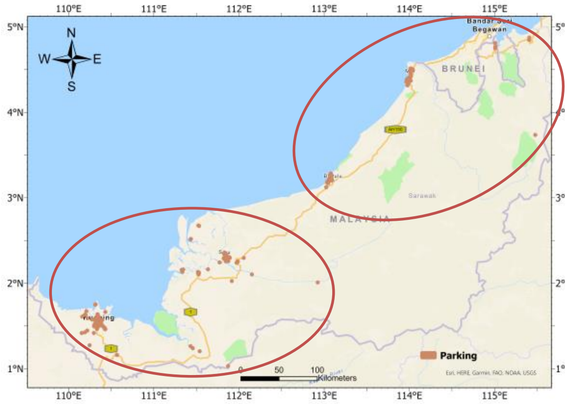
Surface	The uppermost layer of any object or land, encompassing natural surfaces.
Vegetation	Areas characterized by natural plant growth or intentional cultivation, including residential areas, farmland, forests, and grasslands.
Land Use	Classification of land based on human activities and purposes, such as residential, commercial, industrial, agricultural, recreational, and conservation.
Island	The delineation of Borneo Island's boundary.
Building	Constructed structures like houses, apartments, churches, warehouses, and kiosks.
Airport	Facilities designated for public, military, and other purposes related to air transportation.
Water	Areas of water within the boundaries of the local state.



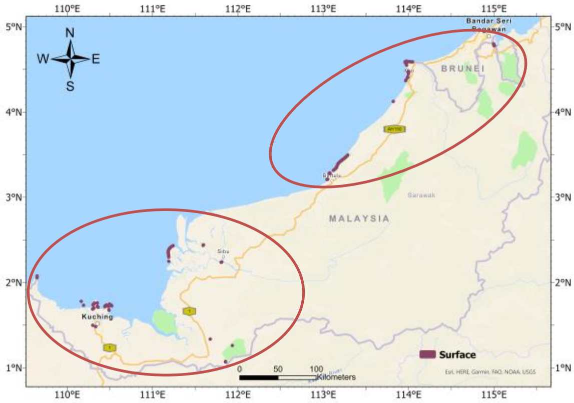
(a) Protected Area (b) Point of Interest



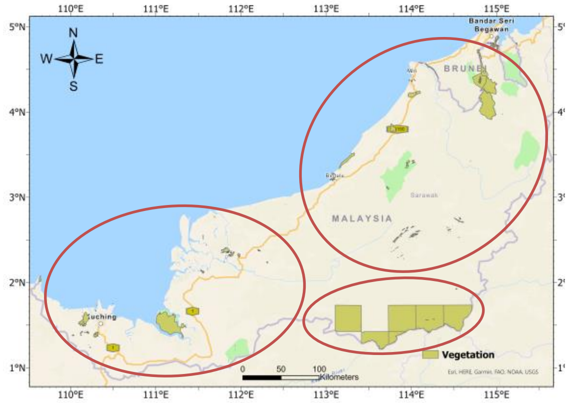
(c) Settlement



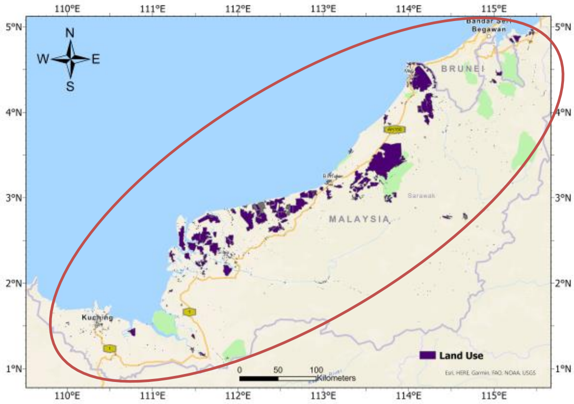
(d) Parking



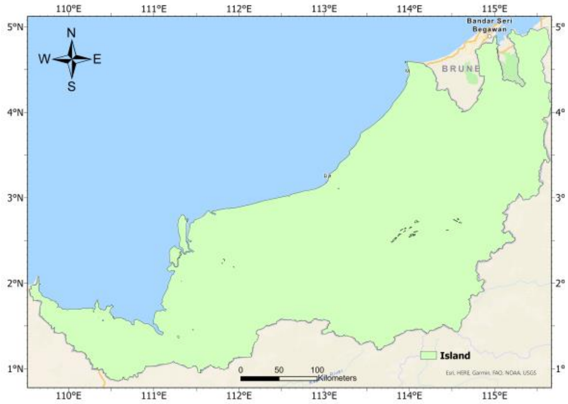
(e) Surface



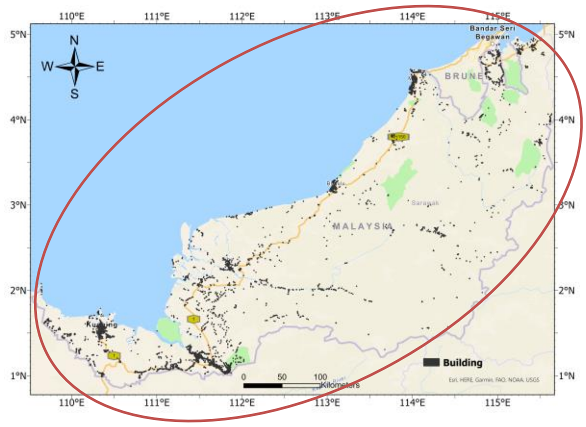
(f) Vegetation



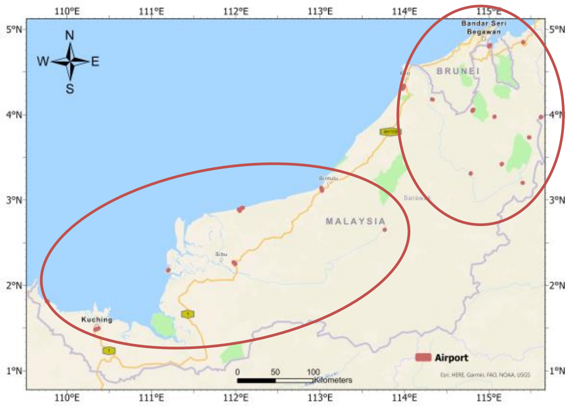
(g) Land Use



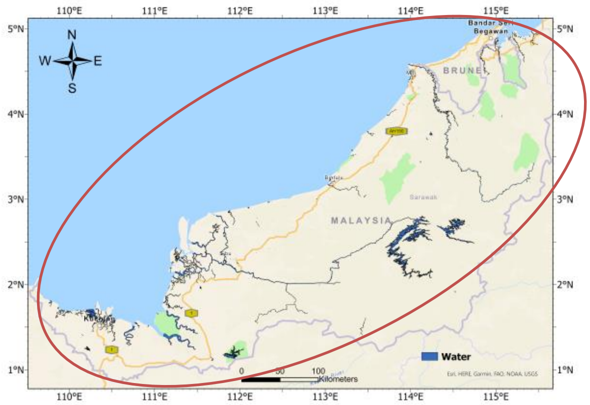
(h) Island



(i) Building



(j) Airport



(k) Water

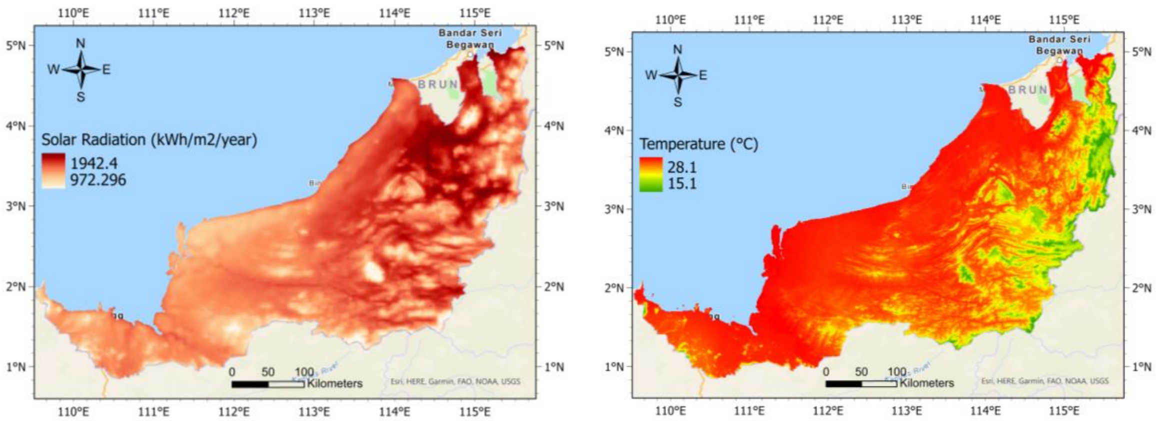
Figure 3.5 (a) to (k): Mapping of Structured Data (NextGIS, 2023)

The first filtration process involves two inputs: the coordinates of green energy locations in the point shapefile layer and the multi-structured data in the polygon shapefile layers. The polygon layers of structured data are attributed as either “inside” or “outside”. The “inside” attribute denotes that points located within a polygon are removed, whereas the “outside” attribute specifies the removal of points situated outside a polygon. The pseudocode algorithm for the initial phase of alternatives filtration, as illustrated in Figure B-1 (Appendix), has been developed and is executable within the ArcGIS Pro Python window. The remaining points after the first filtration are presented in the content layer of ArcGIS Pro. This initial filtration phase yields optimal HES. The optimal WES are obtained from the remaining coordinates out of 19,237, with wind speeds greater than 3 ms^{-1}.

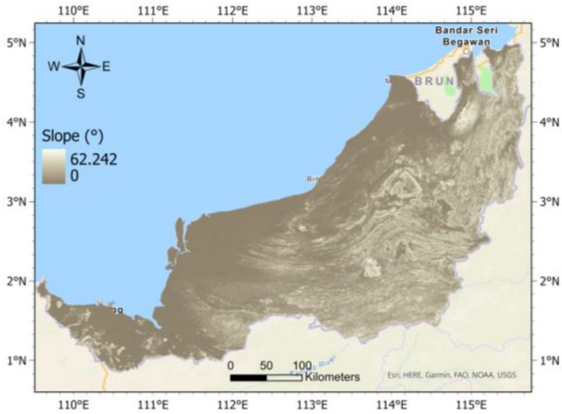
To initiate the second phased of filtration process for identifying SES, all the criteria as outlined in Table 3.2 are utilized to assess and rank potential SES. Among these criteria, solar radiation holds significant importance. Hence, a matured long-term annual average GHI in $kWh\,m^{-2}\,year^{-1}$, spanning from 2007 to 2018 (11 years), is considered. Temperature is also a crucial criterion as it affects the efficiency of solar panels. The slope of the land impacts the installation angle of PV panels, thereby influencing sunlight exposure. Elevation contributes to atmospheric conditions affecting solar radiation levels. Proximity to power transmission lines, roads, residential areas, and urban facilities is vital due to associated costs, installation complexities, and logistical challenges. The distance from water sources is considered as water serves as a cooling agent in solar systems. Similarly, the distance from protected areas is significant to prevent SES installations near prohibited zones, thus averting potential land disputes and environmental impacts. Furthermore, the distance from settlements and population density is crucial to ensure proximity to areas with high power demand. These criteria collectively establish a comprehensive framework for identifying optimal SES, encompassing climatic, technical, accessibility, environmental, and social factors. Each criterion is assigned a unique identifier, ranging from C_1 to C_{12}, for distinction. These criteria data are sourced and extracted from various outlets in raster maps, including Solargis, DIVA-GIS, and NextGIS. Proximity analysis tools in ArcGIS Pro are employed for criteria C_5 to C_{11} to generate necessary spatial raster mapping data. This comprehensive approach ensures the consideration and integration of all relevant criteria into the analysis process. The remaining potential SES from the first phase of filtration must extract their associated values from all the raster maps, as depicted in Figures 3.6 (a) to (l).

Table 3.2: Influential Criteria

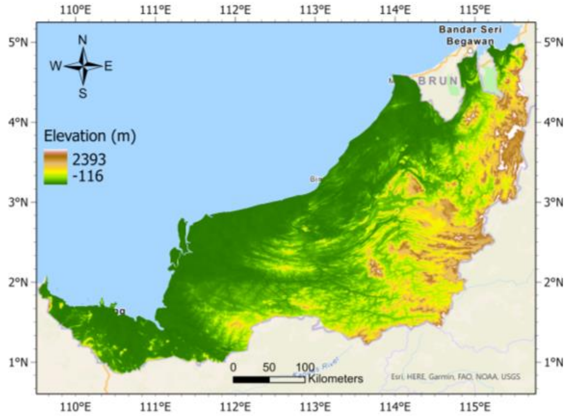
ID	Aspect	Criteria	Source of Raster Map
C_1	Climatic	Solar Radiation ($kWh\ m^{-2}\ year^{-1}$)	(Solargis, 2023)
C_2		Temperature (℃)	(Solargis, 2023)
C_3	Technical	Slope (°)	(DIVA-GIS, 2023)
C_4		Elevation (m)	(DIVA-GIS, 2023)
C_5	Accessibility	Proximity to Power Transmission Lines (km)	(NextGIS, 2023)
C_6		Proximity to Roads (km)	(DIVA-GIS, 2023)
C_7		Proximity to Residential Areas (km)	(NextGIS, 2023)
C_8		Proximity to Urban Facilities (km)	(NextGIS, 2023)
C_9	Environmental	Distance from Water (km)	(NextGIS, 2023)
C_{10}		Distance from Protected Areas (km)	(NextGIS, 2023)
C_{11}		Distance from Settlement (km)	(NextGIS, 2023)
C_{12}	Social	Population Density $\left(\frac{People}{km^2}\right)$	(DIVA-GIS, 2023)



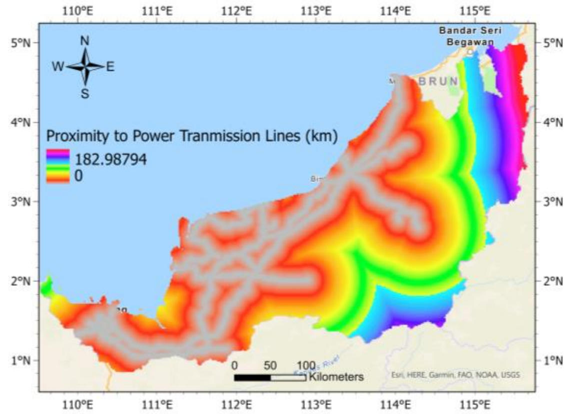
(a) Solar Radiation (b) Temperature



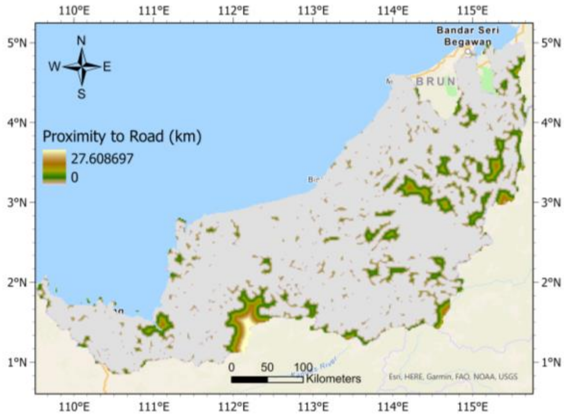
(c) Slope



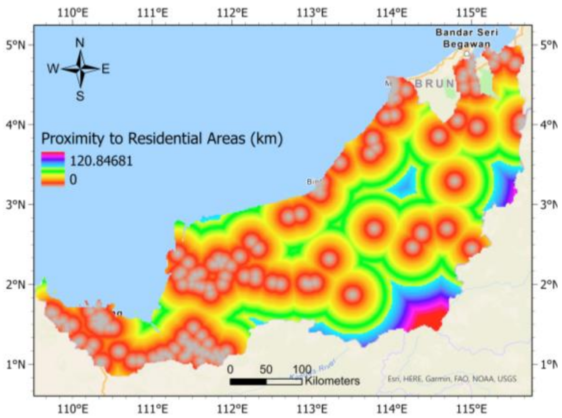
(d) Elevation



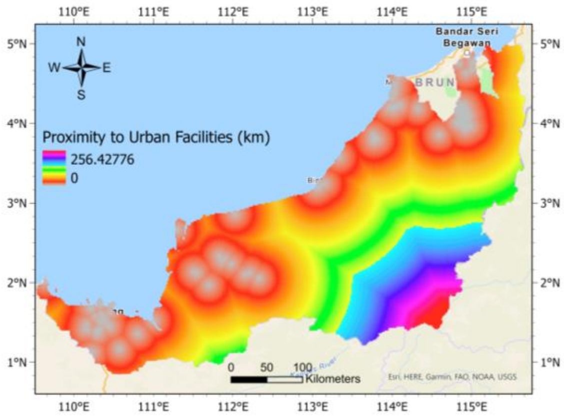
(e) Proximity to Power Transmission Lines



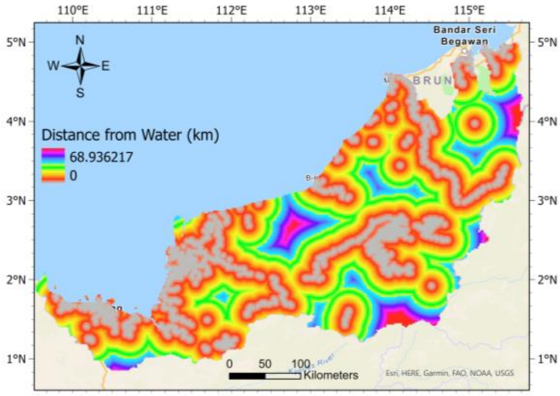
(f) Proxmity to Road



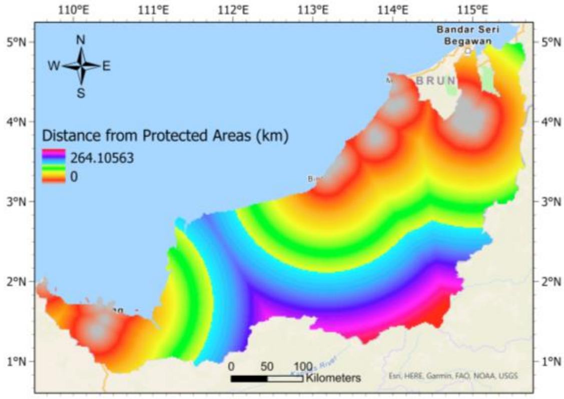
(g) Proximity to Residential Areas



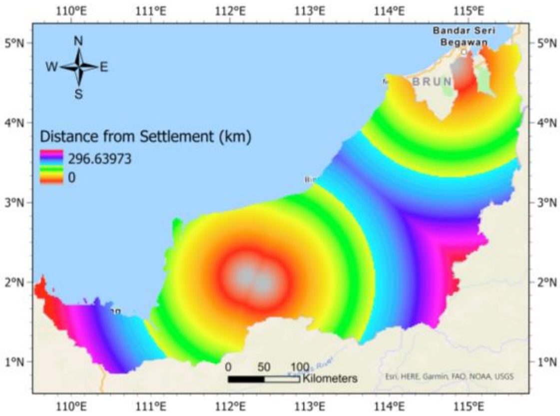
(h) Proximity to Urban Facilities



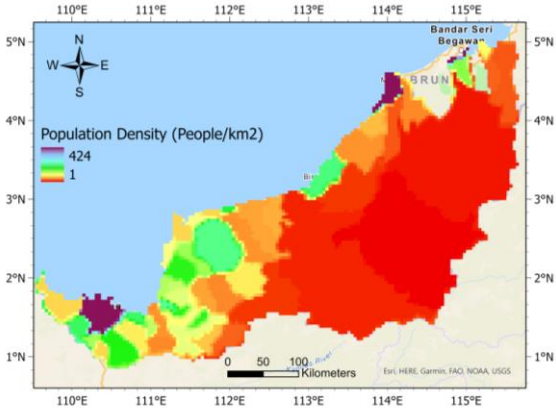
(i) Distance from Water



(j) Distance from Protected Areas



(k) Distance from Settlement



(l) Population Density

Figure 3.6 (a) to (l): Mapping of Influential Criteria (NextGIS, 2023)

The input point layer containing the remaining SES from the first filtration process, along with the 12 criteria raster layers in either "tif" or "vrt" format as highlighted in Figures 3.6 (a) to (l), are employed to extract numerical raster values using the pseudocode algorithm depicted in Figure B-2 (Appendix). Once all the remaining SES have acquired their respective raster values based on the influential criteria, the subsequent step is to develop criteria constraints in preparation for the second phase of SES filtration.

The second filtration process relies on defined criteria constraints, providing a basis for preserving SES that meet acceptable criteria values amidst the vast array of filtered SES in the pursuit of optimal SES. It is essential to acknowledge that there are no universally perfect values when establishing criteria constraints. Therefore, these constraints are derived from reputable high-impact journals, offering valuable guidelines. Efficient solar energy generation necessitates high solar radiation levels ($\geq 1200\ kWh\ m^{-2}\ year^{-1}$). To ensure optimal solar panel operation, a minimum temperature of 15 ℃ is set for locations in relatively warm climates, with an upper limit of 28 ℃ to avoid detrimental effects on panel performance and lifespan due to excessive heat. Slopes beyond 25 ° are considered unacceptable as they significantly impact solar shading and pose challenges in construction and maintenance. Although high elevation is desirable for receiving more sunlight, locations above 2200 m are excluded due to accessibility concerns, while areas below sea level (elevation $< 0\ m$) are also disregarded. For safety measures and to minimize power losses, installation costs, and difficulties, there is a mandated minimum distance from power transmission lines, roads, residential, and urban areas. Additionally, a minimum distance of 0.1 km from water bodies is set to prevent environmental impact, with an upper limit of 20 km to ensure proximity to potential water sources as a cooling agent for solar panels. A minimum distance of 0.1 km from protected areas is established to minimize environmental impact and adhere to conservation regulations. Similarly, a minimum distance of 0.1 km from settlements is imposed to address safety concerns and reduce potential disruptions to local communities. A population density greater than zero is required to ensure that the SES is in areas with a residential presence, reflecting the demand for power (Deveci et al., 2021; Tercan et al., 2021). Table 3.3 demonstrates the criteria constraints, and the second filtration process is partitioned into 19 rounds. During this process, points that fall outside the interval defined by the criteria constraints are eliminated using the pseudocode algorithm depicted in Figure B-3 (Appendix). Following the completion of the second filtration process, the further filtered SES are then passed on to the second layer for execution using fuzzy TOPSIS, where their rankings are determined.

Table 3.3: Criteria Constraints (Deveci et al., 2021; Tercan et al., 2021)

Criteria	Round	Criteria Constraint
Solar Radiation ($kWh\ m^{-2}\ year^{-1}$)	1	$C_1 \geq 1200$
Temperature (℃)	2	$C_2 \geq 15$
	3	$C_2 < 28$
Slope (°)	4	$C_3 < 25$
Elevation (m)	5	$C_4 < 2200$
	6	$C_4 \geq 0$
Proximity to Power Transmission Lines (km)	7	$C_5 \geq 0.01$
	8	$C_5 < 50$
Proximity to Roads (km)	9	$C_6 \geq 0.1$
	10	$C_6 < 50$
Proximity to Residential Areas (km)	11	$C_7 \geq 0.3$
	12	$C_7 < 45$
Proximity to Urban Facilities (km)	13	$C_8 \geq 0.3$
	14	$C_8 < 45$
Distance from Water (km)	15	$C_9 \geq 0.1$
	16	$C_9 < 20$
Distance from Protected Areas (km)	17	$C_{10} \geq 0.1$
Distance from Settlement (km)	18	$C_{11} \geq 0.1$
Population Density $\left(\frac{People}{km^2}\right)$	19	$C_{12} > 0$

3.3.2 Proposed Fuzzy TOPSIS Algorithm

The second layer develops fuzzy TOPSIS algorithm to identify and rank the top 100 optimal SES from the outputs of the first layer. Fuzzy logic operations are utilized due to their proven effectiveness across various problem domains, including decision-making analysis, pattern recognition, control systems, robotic automation, artificial intelligence, and even medical applications. The integration of fuzzy to the MCDM method is a strategic choice driven by its effectiveness in managing vagueness and imprecision. Unlike conventional MCDM methods, which are limited to handling crisp values (0 and 1), this approach is particularly crucial given that traditional methods tend to be highly sensitive to alterations in input data. Even minor adjustments to the input data can impact the results. Therefore, fuzzy operations offer enhanced adaptability by incorporating a degree of membership. This feature empowers decision-makers to assess the extent to which each alternative or criterion aligns with a specific category, providing a more realistic depiction of ambiguity within decision-making scenarios. Various fuzzy membership functions are available, including triangular, Gaussian, trapezoidal, sigmoidal, and bell-shaped. The triangular membership function is selected for its notable advantages, including straightforward interpretation, simplicity, and intuitive representation, making it well-suited for the proposed model. Moreover, TOPSIS is preferred over other MCDM methods for integration with fuzzy algorithms due to its adeptness in handling multiple criteria by assessing both ideal and anti-ideal solutions. This makes it particularly suitable for identifying SES, where influential criteria encompass both benefit and cost attributes. Additionally, TOPSIS implementation is characterized by its simplicity and transparency, ensuring that the reliability and interpretability of results are not compromised. Therefore, a fuzzy TOPSIS algorithm is proposed, which not only identifies promising courses of action but also considers and accommodates the inherent complexity and uncertainty associated with decision-making processes related to SES identification. The implementation of the fuzzy TOPSIS algorithm entails seven steps, denoted as (i), (ii), (iii), (iv), (v), (vi), and (vii).

(i) Define Inputs

Five elements are designated as inputs for the fuzzy TOPSIS algorithm.

(a) Alternatives: These are the remaining alternatives following the second filtration process. They are structured in a matrix format $(m \times n)$, with m denotes the alternatives and n represents the corresponding raster values for criteria.

(b) Criteria: Designated as C_1 to C_{12}, they integrate fuzzy linguistic parameters within predefined intervals, as demonstrated in Table 3.4.

Table 3.4: Criteria Parameters (Deveci et al., 2021; Tercan et al., 2021)

Very Low	**Low**	**Medium**	**High**	**Very High**
$1200 < C_1 \leq 1400$	$1400 < C_1 \leq 1600$	$1600 < C_1 \leq 1700$	$1700 < C_1 \leq 1800$	$C_1 > 1800$
$27 < C_2 \leq 28$	$26 < C_2 \leq 27$	$25 < C_2 \leq 26$	$24 < C_2 \leq 25$	$15 < C_2 \leq 24$
$15 < C_3 \leq 25$	$10 < C_3 \leq 15$	$5 < C_3 \leq 10$	$2 < C_3 \leq 5$	$C_3 \leq 2$
$0 < C_4 \leq 200$	$200 < C_4 \leq 450$	$450 < C_4 \leq 750$	$750 < C_4 \leq 1200$	$1200 < C_4 \leq 2200$
$20 < C_5 \leq 50$	$15 < C_5 \leq 20$	$10 < C_5 \leq 15$	$5 < C_5 \leq 10$	$0.01 < C_5 \leq 5$
$30 < C_6 \leq 50$	$20 < C_6 \leq 30$	$10 < C_6 \leq 20$	$5 < C_6 \leq 10$	$0.1 < C_6 \leq 5$
$30 < C_7 \leq 45$	$20 < C_7 \leq 30$	$15 < C_7 \leq 20$	$10 < C_7 \leq 15$	$0.3 < C_7 \leq 10$
$30 < C_8 \leq 45$	$20 < C_8 \leq 30$	$15 < C_8 \leq 20$	$10 < C_8 \leq 15$	$0.3 < C_8 \leq 10$
$16 < C_9 \leq 20$	$12 < C_9 \leq 16$	$8 < C_9 \leq 12$	$4 < C_9 \leq 8$	$0.1 < C_9 \leq 4$
$0.1 < C_{10} \leq 1$	$1 < C_{10} \leq 2$	$2 < C_{10} \leq 3$	$3 < C_{10} \leq 4$	$C_{10} > 4$
$30 < C_{11}$	$20 < C_{11} \leq 30$	$10 < C_{11} \leq 20$	$5 < C_{11} \leq 10$	$0.1 < C_{11} \leq 5$
$0 < C_{12} \leq 100$	$100 < C_{12} \leq 200$	$200 < C_{12} \leq 300$	$300 < C_{12} \leq 400$	$C_{12} > 400$

(c) Criteria Attributes: They are categorized as either benefit or cost. Benefit criteria signify a preference for higher values, while cost criteria indicate a preference for lower values. Table 3.5 presents all 12 criteria attributes along with their respective IDs.

Table 3.5: Criteria Attributes (Deveci et al., 2021)

ID	**Criteria Attribute**
C_1	Benefit
C_2	Cost
C_3	Cost
C_4	Benefit
C_5	Cost
C_6	Cost
C_7	Cost
C_8	Cost
C_9	Cost
C_{10}	Benefit
C_{11}	Cost
C_{12}	Benefit

(d) Fuzzy Membership Functions: Triangular membership functions are constructed, encompassing Very Low (VL), Low (L), Medium (M), High (H), and Very High (VH), as depicted in Figure 3.7.

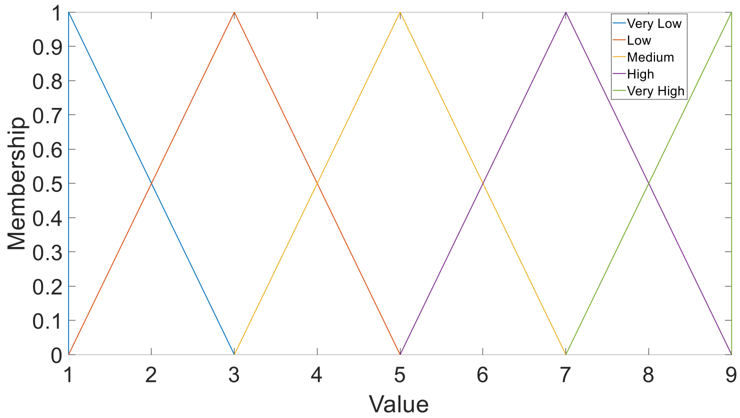


Figure 3.7: Triangular Fuzzy Membership Functions (Fu & Tzeng, 2016)

(e) Fuzzy Criteria Weights: They are divided into lower, median, and upper sections as highlighted in Table 3.6.

Table 3.6: Fuzzy Criteria Weight (Deveci et al., 2021)

ID	Fuzzy Weight		
	Lower	Median	Upper
C_1	8.43	12.12	17.65
C_2	6.87	10.85	16.06
C_3	6.57	9.88	14.85
C_4	5.56	10.48	15.61
C_5	7.02	10.42	15.53
C_6	5.81	9.58	14.47
C_7	5.76	9.21	14.02
C_8	5.76	9.21	14.02
C_9	5.76	9.21	14.02
C_{10}	7.27	11.03	16.29
C_{11}	5.76	9.21	14.02
C_{12}	6.21	9.76	14.70

The alternatives that remain after the second filtration process are organized into an input decision matrix, which is stored in an excel file named filename.xlsx. All these inputs are then integrated into the fuzzy TOPSIS algorithm as illustrated in Figure B-4 (Appendix) and subsequently executed within MATLAB.

(ii) Decision Matrix with Fuzzy Triangular Membership Values

All values within the decision matrix are initially transformed into fuzzy linguistic values to create matrix B, comprising VL, L, M, H, and VH. Subsequently, each fuzzy linguistic parameter is assigned its corresponding numerical values (1, 3, 5, 7, and 9), and these values are stored within matrix B_num. Following this stage, the numerical values (1, 3, 5, 7, and 9) are further mapped into fuzzy triangular membership values, specifically [(1, 1, 3), (1, 3, 5), (3, 5, 7), (5, 7, 9), and (7, 9, 9)], respectively. These matrices, containing the fuzzy triangular membership values, are then stored in B_cell. The conversion of decision matrix values into fuzzy triangular membership values is outlined in Figure B-5 (Appendix).

(iii) Normalize Fuzzy Decision Matrix

The normalization of a fuzzy decision matrix commonly employs four main methods: max, max-min, sum, and vector normalization. Among these, the "max method" is preferred for its simplicity and effectiveness when integrated with fuzzy algorithms, as formulated in Equations 3.1 to 3.4. However, it may exhibit sensitivity issues in the presence of outliers, particularly when the difference between the minimum and maximum values in the decision matrix is significant. To mitigate this, fuzzy logic is introduced to categorize decision matrix values into predefined sets of {Very Low (L) = [1, 1, 3]}, {Low (L) = [1, 3, 5]}, {Medium = [3, 5, 7]}, {High = [5, 7, 9]}, and {Very High = [7, 9, 9]}. This classification constrains the range of values within the decision matrix, rendering the "max method" highly suitable for normalization in this context. Let B_cell_{ij} denotes the fuzzy decision matrix and C_cell_{ij} denotes the normalized fuzzy decision matrix, where $i = 1,2, \ldots, m$ (number of alternatives) and $j = 1,2, \ldots, n$ (number of criteria). The parameters a, b, and c represent elements within the cell array of the fuzzy decision matrix B_cell_{ij} and C_cell_{ij}. Furthermore, $C\_cell_{ij}^{*}$ denotes the normalized fuzzy decision matrix for benefit criteria, while $C\_cell_{ij}^{-}$ represents the normalized fuzzy decision matrix for cost criteria. Pseudocode outlining the algorithm for executing this process is provided in Figure B-6 (Appendix).

$$B\_cell_{ij_c}^{*} = max_i\{B\_cell_{ij_c}\} \quad \text{Equation 3.1}$$

$$C\_cell_{ij}^{*} = \begin{bmatrix} \left(\frac{B\_cell_{i1_a}}{B\_cell_{i1_c}^{*}}, \quad \frac{B\_cell_{i1_b}}{B\_cell_{i1_c}^{*}}, \quad \frac{B\_cell_{i1_c}}{B\_cell_{i1_c}^{*}}\right), \\ \ldots, \\ \left(\frac{B\_cell_{in_a}}{B\_cell_{in_c}^{*}}, \quad \frac{B\_cell_{in_b}}{B\_cell_{in_c}^{*}}, \quad \frac{B\_cell_{in_c}}{B\_cell_{in_c}^{*}}\right) \end{bmatrix} \quad \text{Equation 3.2}$$

$$B\_cell_{ij_a}^{-} = min_i\{B\_cell_{ij_a}\} \quad \text{Equation 3.3}$$

$$C\_cell_{ij}^{-} = \begin{bmatrix} \left(\frac{B\_cell_{i1_a}}{B\_cell_{i1_a}^{-}}, \quad \frac{B\_cell_{i1_b}}{B\_cell_{i1_a}^{-}}, \quad \frac{B\_cell_{i1_c}}{B\_cell_{i1_a}^{-}}\right), \\ \ldots, \\ \left(\frac{B\_cell_{in_a}}{B\_cell_{in_a}^{-}}, \quad \frac{B\_cell_{in_b}}{B\_cell_{in_a}^{-}}, \quad \frac{B\_cell_{in_c}}{B\_cell_{in_a}^{-}}\right) \end{bmatrix} \quad \text{Equation 3.4}$$

(iv) Normalize Fuzzy Criteria Weight

Let FCW_{ij} denote the fuzzy criteria weights, representing the intensity of importance for each criterion, where $i = 1,2, \ldots, m$ (number of alternatives) and $j = 1,2, \ldots, n$ (number of criteria). The fuzzy criteria weight is computed within D_cell_{ij}. The parameters a, b, and c represent the elements in cell arrays in both C_cell_{ij} and FCW_{ij}. Figure B-7 (Appendix) outlines the algorithmic procedure for computing the normalized fuzzy criteria weight.

$$D_{cell_{ij}} = [(C\_cell_{i1_a}.FCW_{i1_a},\ C\_cell_{i1_b}.FCW_{i1_b}, C\_cell_{i1_c}.FCW_{i1_c}),$$

Equation 3.5

$$\ldots, (C\_cell_{in_a}.FCW_{in_a},\ C\_cell_{in_b}.FCW_{in_b}, C\_cell_{in_c}.FCW_{in_c})]$$

(v) Fuzzy Positive Ideal Solution (FPIS) and Fuzzy Negative Ideal Solution (FNIS)

Equations 3.6 to 3.15 are formulated to calculate the FPIS and FNIS, which are then used to compute the Euclidean distance for each SES.

Let $FPIS_j = \{D_{cell_1^*}, \ldots, D_{cell_n^*}\}$ for n criteria,

If $size\left[max_i\left\{D_{cell_{ij_c}}\right\}\right] \leq 1$, select $D_{cell_{ij}^*} = max_i\{D\_cell_{ij_c}\}$ Equation 3.6

If $size\left[max_i\left\{D_{cell_{ij_c}}\right\}\right] > 1$, select $D_{cell_{ij}^*} = max_i\{D\_cell_{ij_{bc}}\}$ Equation 3.7

If $size\ [max_i\left\{D_{cell_{ij_{bc}}}\right\}] > 1$, select $D_{cell_{ij}^*} = max_i\{D\_cell_{ij_{abc}}\}$ Equation 3.8

$max_i\left\{D_{cell_{ij_{bc}}}\right\} = max_i\{D\_cell_{ij_c}\}, max_i\{D\_cell_{ij_b}\}$ Equation 3.9

$max_i\left\{D_{cell_{ij_{abc}}}\right\} = max_i\{D\_cell_{ij_c}\}, max_i\{D\_cell_{ij_b}\}, max_i\{D\_cell_{ij_a}\}$ Equation 3.10

Let $FNIS_j = \{D\_cell_{\bar{1}}, \ldots, D\_cell_{\bar{n}}\}$ for n criteria,

If $size\ [min_i\{D\_cell_{ij_a}\}] \leq 1$, select $D\_cell_{\bar{ij}} = min_i\{D\_cell_{ij_a}\}$ Equation 3.11

If $size\ [min_i\{D\_cell_{ij_a}\}] > 1$, select $D\_cell_{\bar{ij}} = min_i\{D\_cell_{ij_{ab}}\}$ Equation 3.12

If $size\ [min_i\{D\_cell_{ij_{ab}}\}] > 1$, select $D\_cell_{\bar{ij}} = min_i\{D\_cell_{ij_{abc}}\}$ Equation 3.13

$min_i\left\{D_{cell_{ij_{ab}}}\right\} = min_i\{D\_cell_{ij_a}\}, min_i\{D\_cell_{ij_b}\}$ Equation 3.14

$min_i\left\{D_{cell_{ij_{abc}}}\right\} = min_i\{D\_cell_{ij_a}\}, min_i\{D\_cell_{ij_b}\}, min_i\{D\_cell_{ij_c}\}$ Equation 3.15

(vi) Determine Euclidean Distance for FPIS and FNIS

The D_FPIS_{ij} and D_FNIS_{ij} are the Euclidean distances for benefit and cost criteria respectively, where i denotes the SES index and j represents the criteria index ranging from 1 to n (the total number of criteria). The summation of these Euclidean distances is denoted as $SumDis\_FPIS_i$ and $SumDis\_FNIS_i$, as expressed in Equations 3.16 to 3.19. The process of generating these Euclidean distances is depicted in Figure B-9 (Appendix).

Summation of Euclidean distance for D_FPIS_{ij},

$$D\_FPIS_{ij} = \left\{ \begin{matrix} \sqrt{\frac{1}{3}\left[\left(D\_cell_{i1_a} - FPIS_{1_a}\right)^2 + \left(D\_cell_{i1_b} - FPIS_{1_b}\right)^2 + \left(D\_cell_{i1_c} - FPIS_{1_c}\right)^2\right]}, \\ \dots, \\ \sqrt{\frac{1}{3}\left[\left(D\_cell_{in_a} - FPIS_{n_a}\right)^2 + \left(D\_cell_{in_b} - FPIS_{n_b}\right)^2 + \left(D\_cell_{in_c} - FPIS_{n_c}\right)^2\right]} \end{matrix} \right\} \quad \text{Equation 3.16}$$

$$SumDis\_FPIS_i = \left\{ \sum_{j}^{n} \{D\_FPIS_{1j}\}, \dots, \sum_{j}^{n} \{D\_FPIS_{mj}\} \right\} \quad \text{Equation 3.17}$$

Summation of Euclidean distance for D_FNIS_{ij},

$$D\_FNIS_{ij} = \left\{ \begin{matrix} \sqrt{\frac{1}{3}\left[\left(D\_cell_{i1_a} - FNIS_{1_a}\right)^2 + \left(D\_cell_{i1_b} - FNIS_{1_b}\right)^2 + \left(D\_cell_{i1_c} - FNIS_{1_c}\right)^2\right]}, \\ \dots, \\ \sqrt{\frac{1}{3}\left[\left(D\_cell_{in_a} - FNIS_{n_a}\right)^2 + \left(D\_cell_{in_b} - FNIS_{n_b}\right)^2 + \left(D\_cell_{in_c} - FNIS_{n_c}\right)^2\right]} \end{matrix} \right\} \quad \text{Equation 3.18}$$

$$SumDis\_FNIS_i = \left\{ \sum_{j}^{n} \{D\_FNIS_{1j}\}, \dots, \sum_{j}^{n} \{D\_FNIS_{mj}\} \right\} \quad \text{Equation 3.19}$$

(vii) Calculate Closeness Coefficient for SES Ranking

The Closeness Coefficient, CC_i is formulated according to Equation 3.20. Figure B-10 (Appendix) depicts the pseudocode that facilitates the generation of CC_i. These closeness coefficients are utilized to assign rankings for SES. Rankings are determined based on the value of CC_i, with higher values indicating preferable choices among the filtered SES.

$$CC_{ij} = \left\{ \begin{matrix} \dfrac{D_{sum_{1j}^{-}}}{D_{sum_{1j}^{-}} + D_{sum_{1j}^{*}}}, \\ \dfrac{D_{sum_{2j}^{-}}}{D_{sum_{2j}^{-}} + D_{sum_{2j}^{*}}}, \\ \cdots, \\ \dfrac{D_{sum_{mj}^{-}}}{D_{sum_{mj}^{-}} + D_{sum_{mj}^{*}}} \end{matrix} \right\} \qquad \text{Equation 3.20}$$

3.3.3 Validation Method for Green Energy Locations Identification

The validation process began with the reclassification of 12 criteria raster layers, followed by weighted sum analyses carried out using ArcGIS Pro to validate the results of optimal SES derived from the proposed methodology. The reclassification of criteria raster maps is necessary due to the presence of dissimilar units in each layer. Specifically, each individual raster map underwent reclassification within the reclassify window, utilizing the "Equal interval function" method to establish classes. The preference for the "Equal interval function" method over alternative approaches was chosen to uphold consistent class widths, thereby ensuring uniform intervals across varying frequencies to mitigate potential biases in result generation. The maximum frequency ($f = 32$) within the "Equal interval function" method was chosen to enhance result sensitivity. The input parameters for the reclassify tool concerning the benefit criteria, C_b and cost criteria, C_c are detailed in Tables 3.7 and 3.8, respectively. In these tables, the "Start and End" columns indicate the regular intervals for each class, with v_0 representing the minimum value of raster data and v_{32} representing the maximum values of raster data. The “new” column indicates a scale ranging from 1 (indicating the least suitable) to 32 (indicating the most suitable).

Table 3.7: Benefit Criteria Input for Reclassify Tool

Criteria	**Equal interval ($\boldsymbol{f = 32}$)**		
	Start	**End**	**New**
C_b	v_0	v_1	1
	v_1	v_2	2
	v_2	v_3	3
	$\cdots$	$\cdots$	$\cdots$
	v_{31}	v_{32}	32

Table 3.8: Cost Criteria Input for Reclassify Tool

Criteria	**Equal interval ($\boldsymbol{f = 32}$)**		
	Start	**End**	**New**
C_c	v_0	v_1	32
	v_1	v_2	31
	v_2	v_3	30
	$\cdots$	$\cdots$	$\cdots$
	v_{31}	v_{32}	1

As a result, all criteria raster maps are standardized to utilize a uniform scale, ranging from 1 to 32. This standardization allows the utilization of weighted sum analysis. Following this, the fuzzy criteria weights outlined in Table 3.6 are applied as the weights for creating the solar suitability map. However, since the weighted sum analysis tool exclusively accepts crisp weights, the weighted average method, known for its simplicity and efficiency, is employed to convert the symmetrical fuzzy weights into crisp values (Mustapha et al., 2023).

$$Crisp\ Weight = \frac{\sum_{i=1}^{n} \mu_{\bar{c_i}}\,(x_i).\,(x_i)}{\sum_{i=1}^{n} \mu_{\bar{c_i}}\,(x_i)} \qquad \text{Equation 3.21}$$

Within the interface of the weighted sum tool, two input fields outlined in Table 3.9 necessitate completion: (a) The raster layer generated via the reclassify tool and (b) The crisp weight derived from Equation 3.21. Here, n represents the count of elements in the set, $\mu_{\bar{c_i}}(x_i)$ denotes the degree of fuzzy weight, which can be lower, median, or upper and all set to 1/3, with (x_i) representing the fuzzy weight.

Table 3.9: Criteria Weights for Weighted Sum Analysis Tool (Deveci et al., 2021)

Raster Layer	Crisp Weight
Solar Radiation	12.7333
Temperature	11.2600
Slope	10.4333
Elevation	10.5500
Proximity to Power Transmission Lines	10.9900
Proximity to Roads	9.9533
Proximity to Residential Areas	9.6633
Proximity to Urban Facilities	9.6633
Distance from Water	9.6633
Distance from Protected Areas	11.5300
Distance from Settlement	9.6633
Population Density	10.2233

The solar suitability map is created using the weighted sum tool, with the maximum weight derived from the map serving as a reference value representing the most optimal SES location for solar energy utilization. This reference point acted as a benchmark for evaluating the suitability of other locations relative to the most favorable site. To perform this evaluation, the raster values of the identified SES locations are extracted from the solar suitability raster map. These extracted values are then systematically compared to the maximum value of the solar suitability map using Equation 3.22.

$$r = \frac{W_{SES} - W_{min}}{W_{max} - W_{min}} \times 100\ \% \qquad \text{Equation 3.22}$$

Equation 3.22 provides a relative percentage, r (%), quantifying the proximity of the selected SES locations to the optimal site. By incorporating this metric, the methodology facilitated a structured comparison of the results generated by the GIS-based fuzzy TOPSIS and filtration algorithms with the most suitable location for solar energy deployment. Moreover, Equation 3.22 ensures data consistency by standardizing the weight values assigned to each SES location. This consistency is crucial for maintaining the robustness and reproducibility of the model, ensuring that SES locations are evaluated using a uniform and reliable metric. This approach validates the methodology by ensuring the selected SES locations aligned with the most suitable conditions for solar energy deployment.

3.4 Proposed GIS-driven Fuzzy TSP-BIP Algorithm

Figure 3.8 illustrates the flowchart for the integration process of identified sites.

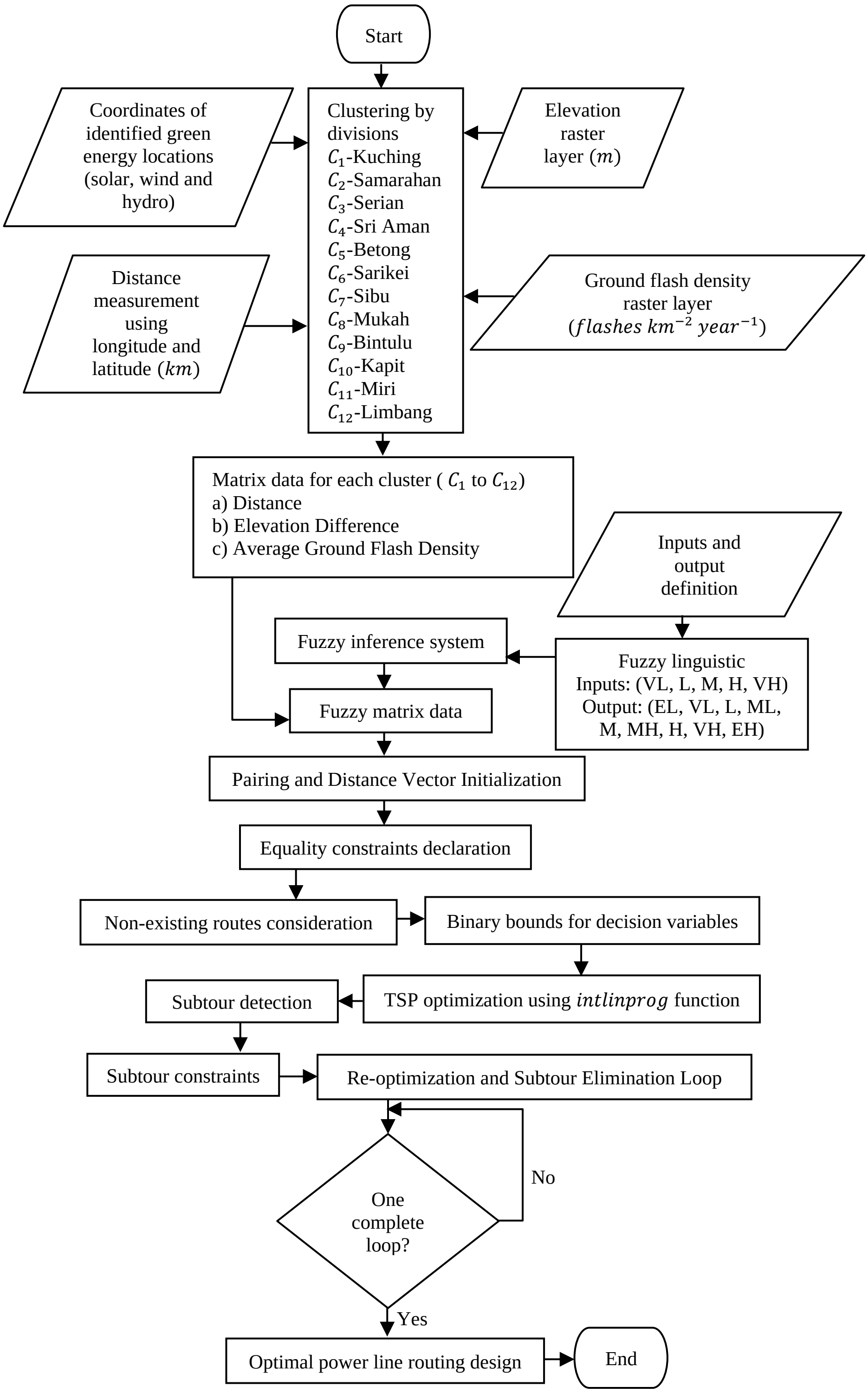


Figure 3.8: Flowchart of Proposed GIS-driven Fuzzy TSP-BIP Algorithm

3.4.1 GIS Input Data

The proposed method takes into account the identified green energy locations, including solar, wind, and hydro, along with three key parameters: distance, elevation difference, and average ground flash density. The objective is to minimize overall values while striking an optimal balance among these variables. Initially, the green energy locations are divided into clusters to facilitate better management. The chosen network topology for integration is the ring topology, renowned for its resilience against power disruptions. This topology provides multiple pathways for power delivery, ensuring continuity even if one section of the network fails.

Distance emerges as a critical parameter in the integration of green energy locations. It is imperative to minimize the total distance (d) of the ring network to reduce electrical power line costs and mitigate power loss over extended distances. Consequently, a distance matrix is compiled, capturing all possible route permutations among green energy locations within each cluster. The integration process prioritizes achieving minimal total distances within the ring system. Figure 3.9 depicts the potential routes of the typical configuration of green energy locations, with GE_i representing individual green energy locations.

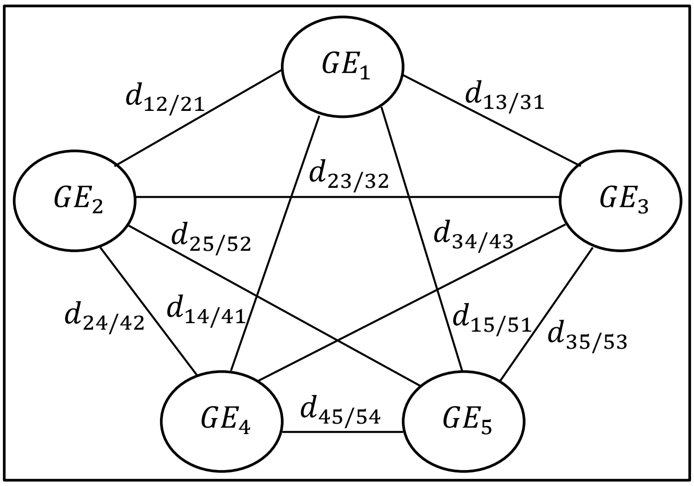


Figure 3.9: Possible Routes for Typical Green Energy Locations Configuration

All identified green energy locations are organized into 12 clusters, and the coordinates of each cluster are extracted into ArcGIS Pro in the form of a shapefile. Figure C-1 (Appendix) outlines the algorithm in pseudocode utilized in the ArcGIS Pro Python window to generate the distance matrix data for each cluster.

The elevation difference (Δe) signifies the steepness between two green energy locations. Typically, a high steepness between them increases installation difficulty and subsequent maintenance needs, leading to escalated costs such as labor expenses and extended power line costs. Therefore, the integration of green energy locations must prioritize minimizing the total elevation difference. As each cluster comprises numerous green energy locations in a ring topology, the elevation difference between each pair of green energy locations is determined using the ArcGIS Pro tool to generate the elevation difference matrix data. Figure 3.10 illustrates the approach to consider the measurement of elevation difference for each pair.

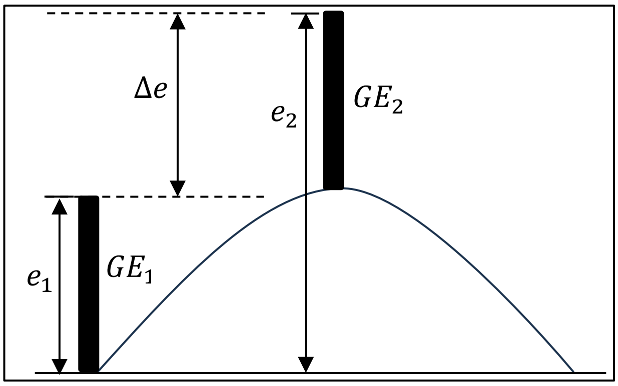


Figure 3.10: Measurement of Elevation Difference

To obtain the elevation difference for each cluster, the elevation raster map for Sarawak is retrieved from DIVA-GIS (DIVA-GIS, 2023), as illustrated in Figure 3.11. Subsequently, green energy locations within each cluster acquire their respective elevation values by extracting values from the elevation raster map. A pairwise matrix is then generated to determine the elevation difference for each pair. The complete algorithm in pseudocode to obtain the elevation difference matrix data for each cluster is presented in Figure C-2 (Appendix).

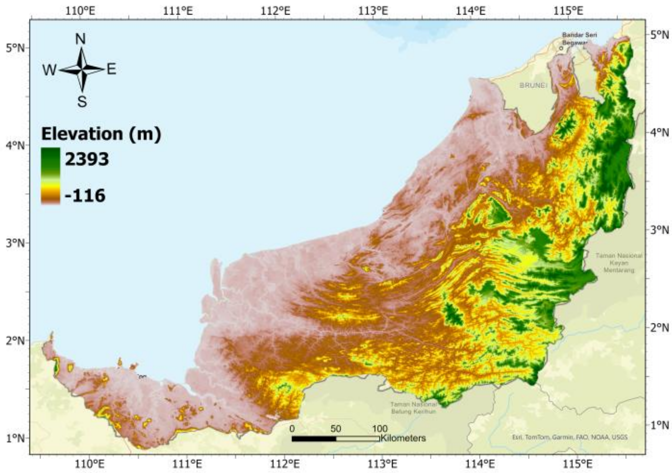


Figure 3.11: Elevation Raster Map (DIVA-GIS, 2023)

Additionally, an innovative methodology is employed to consider the severity of lightning. GFD data source serves as a crucial parameter in calculating the Average Ground Flash Density ($\overline{GFD}$) along the power line connecting two green energy locations. The occurrence of lightning striking the power line escalates the risks of power line failure, blackout, and potential system damage. Therefore, the integration of green energy locations must strive to minimize the total average ground flash density. Figure 3.12 elucidates the approach to determining the $\overline{GFD}$ by employing a significant number of regular intervals, specifically 100, to enhance the sensitivity of input data.

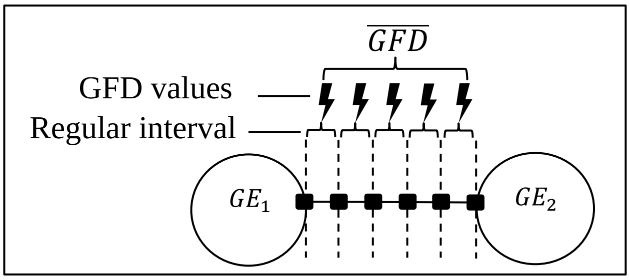


Figure 3.12: Measurement of Average Ground Flash Density

The GFD raster map is acquired from EARTHDATA (EARTHDATA, 2024). Similarly, within each cluster, green energy locations are tasked with extracting their respective GFD values from the ground flash density raster map, as depicted in Figure 3.13. Subsequently, a pairwise matrix is generated to ascertain the average GFD for each pair. The comprehensive algorithm in pseudocode for obtaining the average matrix data for each cluster is elucidated in Figure C-3 (Appendix).

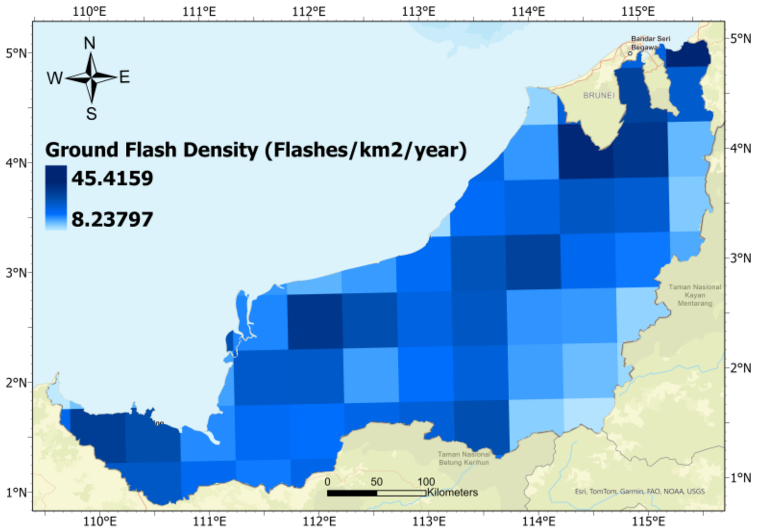


Figure 3.13: GFD Raster Map (EARTHDATA, 2024)

With the aggregation of all GIS input data, the 12 clusters are equipped to generate their respective distance matrix data, elevation matrix data, and average ground flash density matrix data. The ensuing critical phase involves the transmission of these matrices to the fuzzy TSP-BIP algorithm for the integration of green energy locations. This pivotal phase utilizes the fuzzy TSP-BIP algorithm to optimize power line routing design, minimizing distance, elevation differences, and average GFD to enhance the efficient integration of green energy locations.

3.4.2 Proposed Fuzzy TSP-BIP Algorithm

As the matrix data, including distance, elevation difference and average ground flash density are generated, it is essential to determine the minimal values of these three key parameters. These parameters play a critical role in identifying optimal green energy locations. However, traditional TSP solutions are limited to optimizing a single objective function. Therefore, incorporating the TSP model with fuzzy logic operations enables the handling of multi-objective functions, specifically in discerning the best trade-offs among these parameters to derive optimal solutions. The singular objective functions for each parameter are represented by Equations 3.23 to 3.25.

$$f(d) = min \sum_{i=1}^{n} \sum_{j=1, j \neq 1}^{n} \left(d_{ij}.x_{ij}\right) \qquad \text{Equation 3.23}$$

$$f(\Delta e) = \ min \sum_{i=1}^{n} \sum_{j=1, j \neq 1}^{n} \left(\Delta e_{ij}.x_{ij}\right) \qquad \text{Equation 3.24}$$

$$f(\overline{GFD}) = \ min \sum_{i=1}^{n} \sum_{j=1, j \neq 1}^{n} \left(\overline{GFD}_{ij}.x_{ij}\right) \qquad \text{Equation 3.25}$$

Where n = Number of Green Energy Locations (GEs)

d_{ij} = Distance between GE_i and GE_j

Δe_{ij} = Elevation difference between GE_i and GE_j

$\overline{GFD}_{ij}$ = Average GFD between GE_i and GE_j

x_{ij} = Binary decision variables equal to 0 or 1

$$x_{ij} = \begin{cases} 0, & i = j \\ 1, & otherwise \end{cases} \qquad \text{Equation 3.26}$$

Prior to applying fuzzy logic operations, these singular objective functions in Equation 3.23 to 3.25 are integrated to formulate the core multi-objective function, as depicted in Equation 3.27.

$$f(x) = min \begin{cases} \sum_{i=1}^{n} \sum_{j=1, j\neq 1}^{n} \left(d_{ij} . x_{ij}\right) \\ \sum_{i=1}^{n} \sum_{j=1, j\neq 1}^{n} \left(\Delta e_{ij} . x_{ij}\right) \\ \sum_{i=1}^{n} \sum_{j=1, j\neq 1}^{n} \left(\overline{GFD}_{ij} . x_{ij}\right) \end{cases} \qquad \text{Equation 3.27}$$

Once the multi-objective function is established, the focus shifts towards configuring fuzzy logic operations, involving: (i) Defining fuzzy inputs and outputs, (ii) Constructing fuzzy membership functions, and (iii) Developing fuzzy rules tailored to address challenges in multi-objective optimization. The fuzzy inputs encompass distance, elevation difference, and average GFD, which are carefully considered within the framework of fuzzy logic operations. This process culminates in the creation of a fuzzy matrix that comprehensively incorporates and integrates influences from all input parameters. Subsequent stages in the fuzzy logic operations play a pivotal role in determining optimal solutions and trade-offs within the multi-objective optimization landscape for integrating green energy locations.

(i) Definition of Fuzzy Inputs and Output

In the model, three inputs (distance, elevation difference, and average ground flash density) are meticulously defined and integrated to constitute a comprehensive input set. Moreover, a singular fuzzy output containing the fuzzy value is incorporated. The interplay among these inputs and the resulting output is visually represented in Figure 3.14. This graphical depiction offers a broad illustration of the fluctuations in distance, elevation difference, and average ground flash density, which collectively influence and contribute to the overall fuzzy output via the Fuzzy Inference System (FIS).

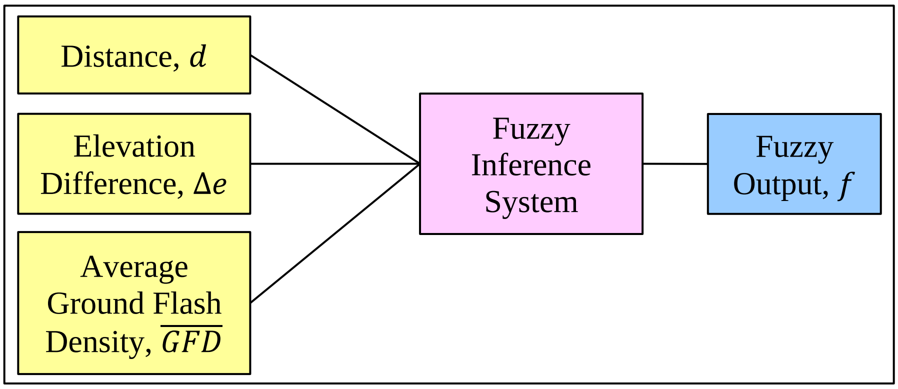


Figure 3.14: Correlation of Inputs and Output

(ii) Creation of Fuzzy Membership Functions

Triangular membership functions are utilized in fuzzy logic to represent linguistic variables and membership degrees for inputs and outputs due to their simplicity and effectiveness. The triangular membership function, denoted as $\mu(x)$, is expressed as Equation 3.28.

$$\mu(x) = \begin{cases} 0, & x \leq a \\ \dfrac{x-a}{b-a}, & a \leq x \leq b \\ \dfrac{c-x}{c-b}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \qquad \text{Equation 3.28}$$

For each input, five regular membership functions are employed, with each featuring five linguistic variables: Very Low (VL), Low (L), Medium (M), High (H), and Very High (VH). The range of these membership functions is determined by the minimum and maximum values of their respective matrix data. A total of 9 membership functions are utilized for the fuzzy output, encompassing Extremely Low (EL), Very Low (VL), Low (L), Medium Low (ML), Medium (M), Medium High (MH), High (H), Very High (VH), and Extremely High (EH). The range of the membership function for the fuzzy output is set from 0.1 to 0.9. A greater number of fuzzy linguistics are employed for the fuzzy output to enhance data sensitivity. Figures 3.15 (a) to (c) depict the fuzzy membership functions for the inputs, while Figure 3.15 (d) illustrates the fuzzy membership function for the output.

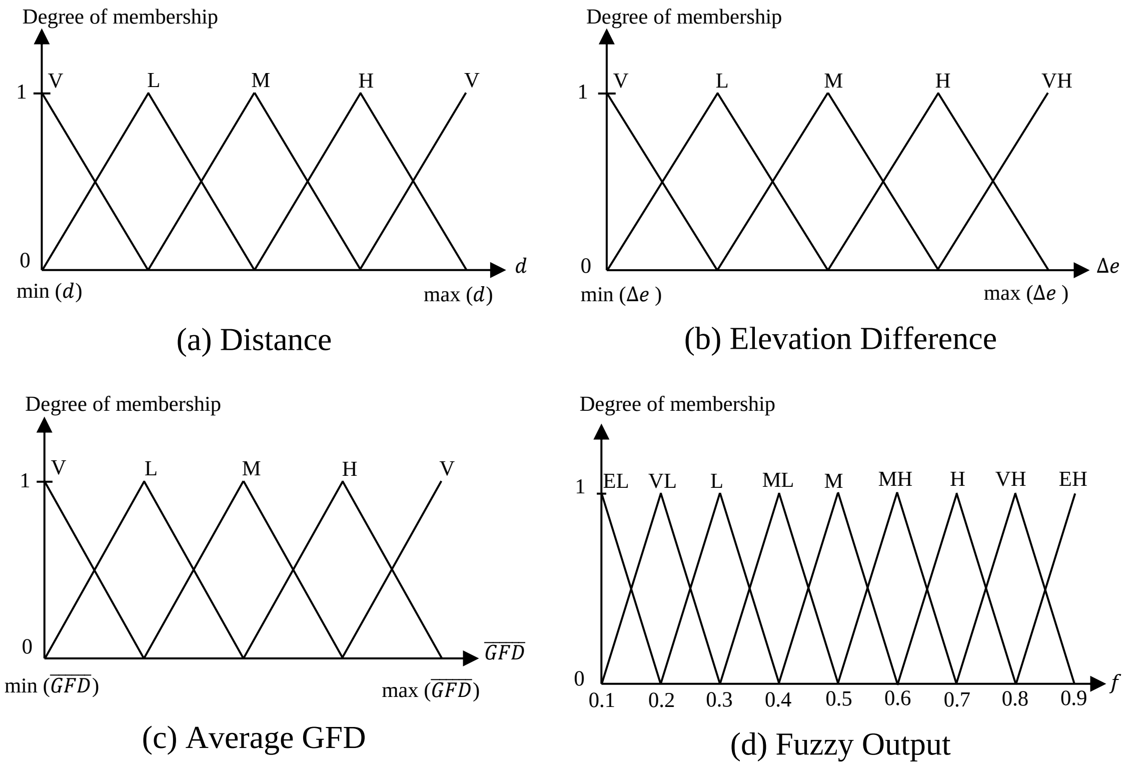


Figure 3.15 (a) to (d): Fuzzy Membership Functions for Inputs and Output

(iii) Fuzzy Rules Setting in FIS

Upon constructing the membership functions for both inputs and output, fuzzy rules in the FIS are established. The total number of fuzzy rules, denoted as R, can be determined using the formula ρ^n, where ρ represents the number of linguistic variables in each input and n denotes the number of inputs. Given that there are five linguistic variables (VL, L, M, H, VH) and three inputs, the formula yields $\rho^n = 5^3$, resulting in a total of 125 fuzzy rules. Table C-1 outlines the fuzzy rules in the FIS.

With all input matrix data, membership functions, and the FIS equipped with fuzzy rules prepared, the production of fuzzy matrix data based on fuzzy output data ensues. Figure C-4 (Appendix) portrays the algorithm in pseudocode for generating the fuzzy matrix data for each cluster. The three inputs previously generated using the ArcGIS Pro tool, along with the FIS fuzzy rule system, are employed to generate fuzzy matrix data, f_m for TSP optimization. It is noteworthy that each cluster (1 to 12) must execute this algorithm individually to produce 12 fuzzy matrix datasets corresponding to the 12 clusters.

BIP emerges as a superior method for optimizing the TSP problem due to its exact nature compared to other approximate algorithms such as GA, NN, TS, GWO, ACO, and more. While dynamic programming is the preferred choice for obtaining reliable TSP results, its efficiency diminishes as the number of green energy locations exceeds 15, resulting in exponential increases in computational time. BIP formulations find widespread application in optimization problems, including TSP, where the objective is to identify the optimal solution using binary decision variables. The implementation of BIP using MATLAB entails four main steps: (i), (ii), (iii) and (iv).

(i) Create Pairs and Distance Vector

The initiation of the TSP-BIP algorithm involves scrutinizing pairs of green energy locations extracted from a fuzzy matrix dataset. In a fuzzy matrix dataset sized $n \times n$, each pair is defined as per Equation 3.29. Following this, Equation 3.30 delineates the process of obtaining a distance vector containing the distances between corresponding pairs of green energy locations. Figure C-5 (Appendix) provides visual representation of the 12-input matrix dataset generated earlier, facilitating the creation of pairs. Subsequently, the distance square matrix is converted into a column distance vector.

$$GE_{p_{ij}} = \{(i,j)|i \in \{1,2,3,\dots,n\}, j \in \{1,2,3,\dots,n\}\}$$ Equation 3.29

$$d_{v_{ij}} = \left\{(f_{m_i})|i \in \{1,2,3,\dots,n\}, j \in \{1,2,3,\dots,n\}\right\}$$ Equation 3.30

Where GE_p = Pairs of green energy locations

n = Number of GEs

d_v = Distance vector

f_m = Fuzzy matrix data

(ii) Equality Constraints and Binary Bounds

Each pair of existing routes should have precisely one input and one output. The responsibility of the $spalloc$ function lies in allocating space for a sparse matrix.

$$A_{eq} = [A_{eq} ; spalloc(2n, length(GE_p), 2n(2n - 1))]$$ Equation 3.31

$$B_{eq} = [B_{eq} ; ones(2n, 1)]$$ Equation 3.32

Where A_{eq} = Matrix embodying the equality constraints

B_{eq} = Column vector encapsulating the right-hand side of equality constraints

$length(GE_p)$ = Total Count of GE_p

In practical terms, the A_{eq} constraint plays a pivotal role in guaranteeing that every GE is precisely traversed once, adhering to the specifications outlined in Figure 3.16.

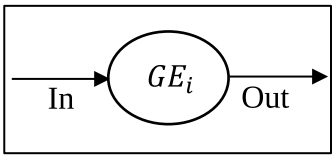


Figure 3.16: In-Out for Each Pair

To eliminate all non-existing routes, represented by R_{non} in the fuzzy matrix dataset, it is imperative to enforce the constraint delineated in Equation 3.33. Specifically for these non-existing routes, the matrices A_{eq} and B_{eq} are formulated as per Equations 3.34 and 3.35, respectively.

$$R_{non} = GE_{p_{ij}}, (i = j), d_{v_{ij}} = 0 \quad \text{Equation 3.33}$$

$$A_{eq}(2n, :) \ = \ R_{non} \quad \text{Equation 3.34}$$

$$B_{eq} \ = \ [B_{eq} \ ; \ 0] \quad \text{Equation 3.35}$$

Moreover, the integer constraint, denoted as $intcon$, along with the lower-bound (l_b) and upper-bound (u_b), are configured in binary representations, employing values of 0 and 1. Figure C-6 (Appendix) provides a depiction of the algorithms utilized for generating equality constraints and defining binary bounds.

$$intcon \ = 1: length(\ d_v) \quad \text{Equation 3.36}$$

$$l_b \ = \ zero\ (length(d_v), 1) \quad \text{Equation 3.37}$$

$$u_b \ = \ ones\ (length(d_v), 1) \quad \text{Equation 3.38}$$

(iii) Optimization with Subtour Detection and Constraints

Upon configuring all parameters, the optimization process commences utilizing the $intlinprog$ function from the MATLAB optimization toolbox. The determination of minimum value within fuzzy matrix data is achieved by integrating f_m into Equation 3.39.

$$\min f_m = \begin{cases} x\ (intcon) \in I \\ A_{ineq}.x \leq B_{ineq} \\ A_{eq}.x = B_{eq} \\ l_b \leq x \leq u_b \end{cases} \qquad \text{Equation 3.40}$$

Where x = Decision variable

$intcon$ = Integer constraint

I = Integer number

A_{ineq} = Inequality matrices

A_{eq} = Equality matrices

B_{ineq} = Inequality vector

B_{eq} = Equality vector

l_b = Lower-bound

u_b = Upper-bound

Equation 3.40 is formulated for decision variable, x_{ij}, which stipulates assigning a value of 0 to decision variables if their magnitude is less than 0.001. Furthermore, Equation 3.41 extracts the relevant pairs, r_p, comprising the indices of non-zero elements from the processed decision variables. Subsequently, Equation 3.42 delineates the representation of subpairs, s_p, which correspond to GEs associated with the non-zero decision variables.

$$x_{ij} = \begin{cases} 0, & x_{ij} < 0.001 \\ x_{ij}, & otherwise \end{cases} \qquad \text{Equation 3.40}$$

$$r_p = \{i | x_{ij} \neq 0\} \qquad \text{Equation 3.41}$$

$$s_p = \{GE_{p_{ij}}[r_p, ;]\} \qquad \text{Equation 3.42}$$

To incorporate constraints for eliminating subtours, a sparse matrix A is initialized as empty, along with an empty vector B. When multiple subtours are detected, a column vector of zeros, with a length equivalent to the number of subtours, s_n, is appended to B. Additionally, a sparse matrix A is augmented with a number of rows equal to the total number of subtours t_n, and several columns equal to the length of the distance vector. An estimated count of non-zero elements equal to n.

$$A = [A; spalloc(t_n, length(d_r), n)] \quad \text{Equation 3.43}$$

$$B = [B; zeros(s_n, 1)] \quad \text{Equation 3.44}$$

To address each subtour, the index for the subsequent inequality constraint is computed. This involves retrieving the nodes within the current subtour and generating all possible pairs of nodes contained within it. Subsequently, for each pair of nodes within the subtour, a binary variable is defined to signify the existence of the current edge within the subtour. This subtour variable, denoted as s_v, is mathematically expressed in Equation 3.45.

$$s_{v_{ij}} = (GE_p = s_{tID}(s_{tp}(jj, 1))) \wedge (GE_p = s_{tID}(s_{tp}(jj, 2))) \quad \text{Equation 3.45}$$

Where s_{tID} = Subtour ID

s_{tp} = Subtour pair

jj = Variable index from 1 to size of s_{tp}

Following this, inequality matrix, A_{ineq} is incorporating these binary values, whereby elements are set to 1 where the pair exists within the subtour. Concomitantly, the right-hand side inequality vector, B_{ineq}, is updated by assigning the length of the current subtour “minus 1”, adhering to the standard formulation for subtour elimination constraints.

Figure C-7 (Appendix) delineates the algorithms devised for optimization employing the *intlinprog* function, integrating subtour detection and its associated constraints.

$$A_{ineq}\ \left(intcon, s_{v_{ij}}\right) = 1 \qquad \text{Equation 3.46}$$

$$B_{ineq}\ (intcon) = length\ (s_{tID}) - 1 \qquad \text{Equation 3.47}$$

(iv) Re-optimization and Subtour Elimination Loop

The TSP-BIP optimization process iteratively proceeds to eliminate subtours until only one subtour persists. Figure 3.17 provides a graphical depiction of the TSP operation, ultimately unveiling the optimal power line routing design for green energy locations. Additionally, Figure C-8 (Appendix) delineates the algorithm in pseudocode for the iterative re-optimization and subtour elimination looping process.

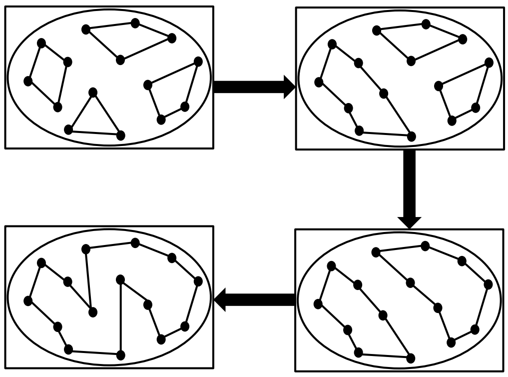


Figure 3.17: Subtour Eliminations for Optimal TSP Configuration

3.4.3 Validation Method for Green Energy Location Integration

During first phase, the validation is made between the optimal results generated by the proposed fuzzy TSP-BIP algorithm and a conventional TSP-BIP algorithm. Two MCDM algorithms (TOPSIS and COPRAS) (Irik Mukhametzyanov, 2024) are employed to evaluate the results, yielding respective rankings. In the subsequent phase, the performance of the proposed fuzzy TSP-BIP algorithm is scrutinized against 7 alternative fuzzy TSP algorithms to determine their performances in terms of optimal solution values and computational efficiency. Fuzzy values generated are integrated into these 7 TSP algorithms and computed utilizing MATLAB. The parameter configurations for each TSP algorithm are delineated in Table 3.10, where adjustments have been made to optimize the probability of obtaining optimal results. Notably, fuzzy TSP-NN does not necessitate specific parameter values due to its inherent coding nature.

Table 3.10: Setting Values for Fuzzy TSP Algorithms

Source	**Algorithm**	**Setting Value**
(Yarpiz / Mostapha Heris, 2024a)	TSP-ACO	Number of iterations = 5,000
(hossein, 2024)	TSP-GA	Population size = 5,000
		Generation = 5,000
(Seyedali Mirjalili, 2024)	TSP-GWO	Number of iterations = 5,000
		Number of search agents = 50
		Lower bound = −5
		Upper bound = 5
(ajevtic, 2024)	TSP-NN	N/A
(Aravind Seshadri, 2024)	TSP-SA	Number of iterations = 5,000
		Initial temperature = 1
		Cooling factor = 0.99
(S. Muhammad Hossein Mousavi, 2024)	TSP-TLBO	Number of iterations = 5,000
		Population size = 5,000
(Yarpiz / Mostapha Heris, 2024b)	TSP-TS	Number of iterations = 5,000

3.5 Proposed IIoT-based System for IGESs

The IIoT-based real-time monitoring and control strategies methodology is structured into four primary sections. The framework depicted in Figure 3.18 illustrates the establishment of IIoT technology in real-time monitoring, control and automation.

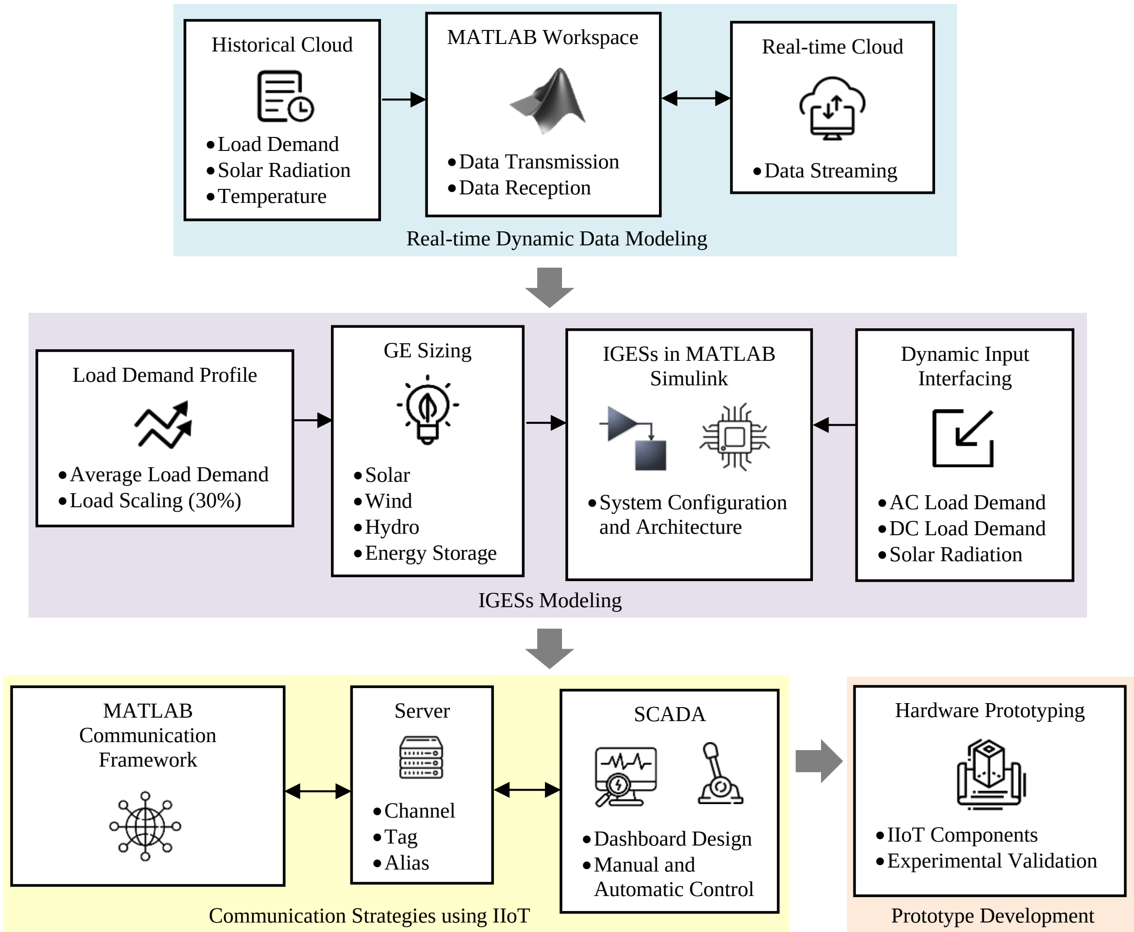
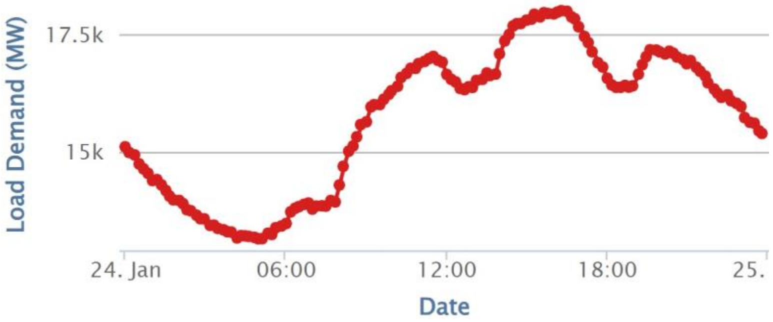


Figure 3.18: Block Diagram of Proposed IIoT-based System for IGESs

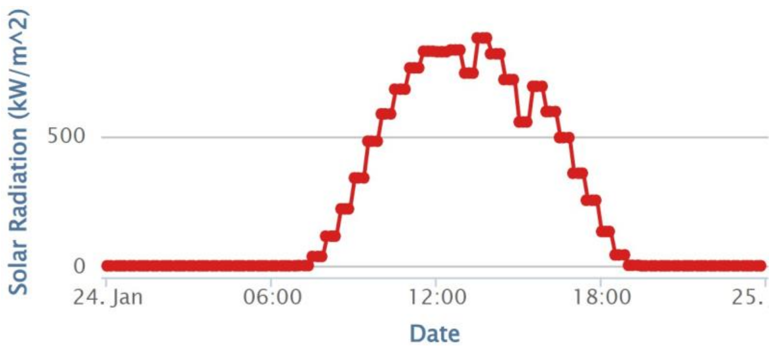
The initial section focuses on modeling dynamic input data IGESs. Following this, the second section illustrates the modeling of IGESs while considering load demand. This involves determining the appropriate generation sizing and integrating these dynamic inputs into the IGESs framework. Moving forward, the third section introduces a communication framework tailored for IGESs modeling, incorporating servers, and SCADA systems. Lastly, the fourth section is dedicated to validating the effectiveness of monitoring and control application strategies through experimental prototyping.

3.5.1 Real-Time Dynamic Data Modeling

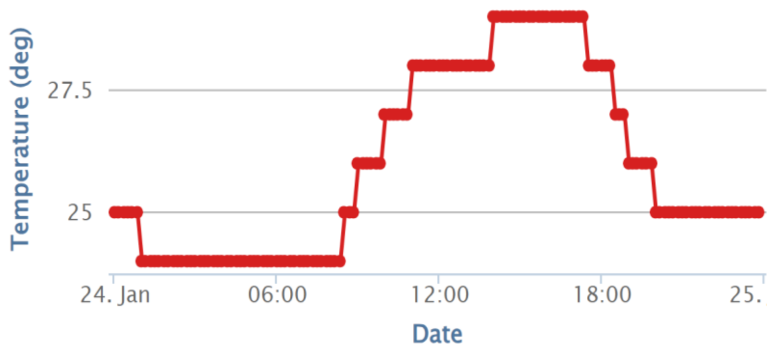
To commence the modeling of dynamic input data, the process entails acquiring 24-hour historical data pertinent to input variables. Specifically, load demand data (Grid System Operator, 2024), along with solar radiation and temperature data (Solcast, 2024) are obtained in CSV format. These CSV files are subsequently imported into the ThingSpeak channel, with the time zone configured to GMT + 08:00 (Kuala Lumpur). Within the ThingSpeak channel, three distinct fields are designated to store the 24-hour historical data, as depicted in Figures 3.19 (a) to (c).



(a) Load Demand Data



(b) Solar Radiation Data



(c) Temperature Data

Figure 3.19: Historical Data (Grid System Operator, 2024; Solcast, 2024)

Following the upload of the 24-hour historical load demand, solar radiation, and temperature data to the ThingSpeak cloud, the MATLAB workspace serves as the platform for streaming historical data based on the current time. The historical load demand data readings are refreshed at 10-minute intervals, while updates for solar radiation and temperature occur at 30-minute intervals. To retrieve the respective historical data based on the current time, t_m is defined as the lower bound of historical time (e.g., 00:00), while t_n represents the upper bound of historical time (e.g., 00:10). The t_c refers to the current time (e.g., 00:08). In this context, the data value corresponding to t_m is selected, as expressed by Equation 3.48.

$$For\ t_m \leq t_c < t_n, select\ t_m \qquad \text{Equation 3.48}$$

Figure D-1 (Appendix) presents the algorithm implemented in pseudocode for extracting and uploading historical data to ThingSpeak, facilitating dynamic modeling. The extraction of historical data necessitates the utilization of an Application Programming Interface (API) key, serving as the authentication token. Essential parameters such as the channel ID, read API key, and fields must be specified according to the user's ThingSpeak channel settings to extract the historical data. The fields are restricted to values 1, 2, and 3, corresponding to the three considered data types. Moreover, the start and end times are provided in the timestamp format “YYYY-MM-DD HH:MM:SS” for accurate recognition of their associated data. Following extraction, the retrieved data is uploaded or written to another channel to serve as current real-time data. Consequently, the new channel ID and write API key must also be specified to model the current real-time data on the ThingSpeak cloud. The purpose of uploading real-time data to ThingSpeak is to seamlessly integrate these data into the IGESs. Additionally, given that the historical demand data exhibits an average of 15,695 MW, and each AC and DC load is constrained within the range of 1 MW to 2 MW, the historical load demand data is multiplied by 100 to represent the output value in W, as stipulated in Equation 3.49.

$$L_r = L_h \times 100 \qquad \text{Equation 3.49}$$

Where L_h = Historical load demand in MW

L_r = Real-time load demand in W

3.5.2 IGESs Modeling

The modeling of IGESs initiates with the determination of their total generation capacity. This capacity can be computed employing Equation 3.50.

$$P_G = P_L + \Delta_l \qquad \text{Equation 3.50}$$

Where P_G = Total generation power

P_L = Total load power

Δ_l = Power loss

Before determining the generation capacity, the load demand is assessed as emphasized in Equation 3.51. Accordingly, hourly load demand data (Grid System Operator, 2024) are provided in Table 3.11. The daily average load demand, L_{avg} in MW is calculated using Equation 3.51.

$$L_{avg} = \frac{\sum_{h=1}^{24} P_{h-1}}{24} \qquad \text{Equation 3.51}$$

Where P_{h-1} = hourly load demand data ranging from hour 0 to 23.

Table 3.11: Hourly Load Demand Data

Hour	**Load (MW)**	**Hour**	**Load (MW)**
0	15,107	12	16,653
1	14,387	13	16,376
2	13,961	14	17,094
3	13,569	15	17,813
4	13,286	16	17,947
5	13,143	17	17,671
6	13,470	18	16,573
7	13,777	19	16,410
8	14,291	20	17,136
9	15,641	21	16,881
10	16,304	22	16,339
11	16,884	23	15,968
		L_{avg}	15,695 MW

In this model, the size of each load (2 AC loads and 2 DC loads) is determined based on L_{avg} as tabulated in Table 3.11. However, L_{avg} It is exceptionally high (15,695 MW), making it unsuitable for modeling purposes. Therefore, the AC loads and DC loads are rescaled based on L_{avg} using Equation 3.52 , ensuring that their active powers fall within the range of 1 MW to 2 MW for simulation purposes. Additionally, a load demand scaling of 30 % is considered when designing the generation capacity. The L_d represents the capacity of load demand for each AC and DC load. Table 3.12 presents the load demand of each load (2 AC loads and 2 DC loads) for both baseline and scaled 30 % scenarios.

$$L_d = \frac{L_{avg}}{10^4}, 1 \leq \ L_{AC/DC} \leq \ 2 \qquad \text{Equation 3.52}$$

Table 3.12: Load Parameters

Load Demand		**Value**
Baseline	Average (kW)	1,569.5
	Average (kWh/day)	37,668
	Peak (kW)	1,794.7
	Load Factor	0.87
Scaled (30 %)	Average (kW)	470.85
	Average (kWh/day)	11,300.4
	Peak (kW)	538.41
	Load Factor	0.87

The IGESs consider three primary resources: solar, wind, and hydro, along with energy storage. Data pertaining to these green energy resources are acquired from the NASA Prediction of Worldwide Energy Resource (POWER) database. Subsequently, the sizing of generation capacity can be determined and optimized through grid search algorithms within HOMER Pro software. Table 3.13 presents the capacity of the generations.

Table 3.13: Green Energy Parameters

Type	Capacity
Solar	9,614 kW
Wind	10 kW
Hydro	1,000 kW
Energy Storage	32,943 kWh

Figure 3.20 depicts the developed IGESs model for IIoT-based real-time monitoring, control and automation. The topology employed in the system is the ring network, chosen for its superior performance in comparison to radial and parallel systems. The ring system is deemed more suitable due to the intermittent nature of GERs, as it allows loads to be supplied from two directions. In the event that one of the power suppliers fails to deliver power to the loads, the other side's terminal can still transmit power to the load. This integrated green energy system operates at 15 kV for its ring system.

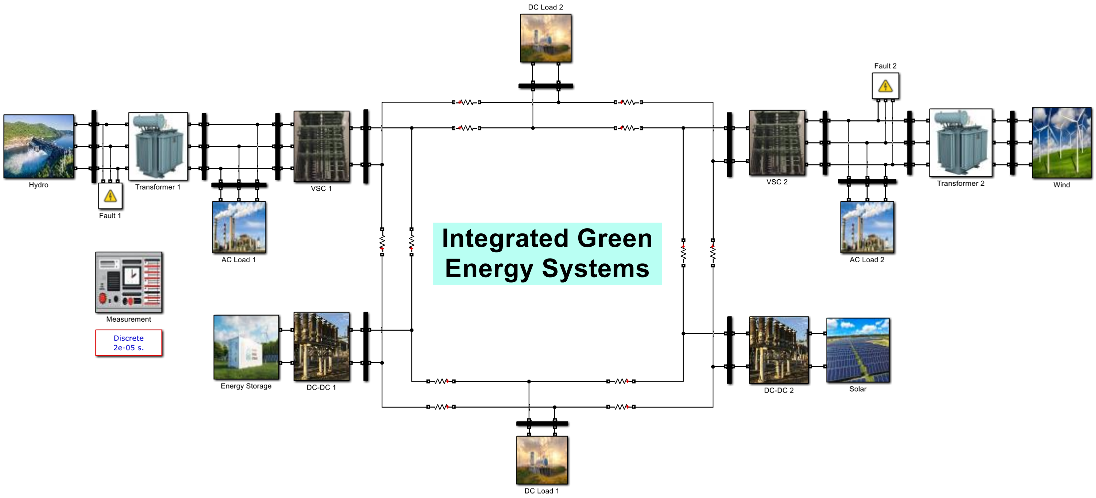


Figure 3.20: Developed IGESs Modeling

The SPR-415E-WHT-D solar panel type is chosen for the solar configuration due to its availability on the market. Each module has a maximum power output of 414.801 W.

Based on the solar PV capacity specified in Table 3.13, an approximate capacity of 10,000 kW is required. The series-connected modules per string are adjusted to achieve a voltage level of 5 kV. The capacity of solar generation is determined using Equation 3.53.

$$C_s = P_m \times S_p \times S_s \qquad \text{Equation 3.53}$$

Where C_s = Solar generation capacity

P_m = PV module power

S_p = Number of parallel strings of PV modules

S_s = Number of series strings of PV modules

The solar PV modules are configured with a nominal voltage of 5 kV, as is the energy storage system. For AC generation, hydro generation operates with a nominal power of 1 MVA at 5 kV, while wind generation is set at 10 kW with 5 kV, based on the capacities specified in Table 3.13. Consequently, both transformers 1 and 2 step up the voltage from 5 $kVAC$ to 11 $kVAC$ from the generation sides. Additionally, both voltage source converters rectify 11 $kVAC$ to 15 $kVAC$. Furthermore, both DC-DC boost converters increase the voltage from 5 $kVAC$ to 15 $kVAC$ from the generation sides to the ring systems. Both AC and DC loads are configured at 1,569.50 kW, as per Table 3.12. Following the establishment of the IGESs model, the dynamic inputs of load demand, solar radiation, and temperature data from the ThingSpeak cloud are integrated to the IGESs model. This integration is facilitated using the "Read Block" of the ThingSpeak input channel to stream the data, which is subsequently routed to the "Goto Block" in MATLAB. Prior to this, the setup of channel ID, read API key, and field is necessary. Figure 3.21 illustrates the process of data transmission from the ThingSpeak cloud to MATLAB Simulink.

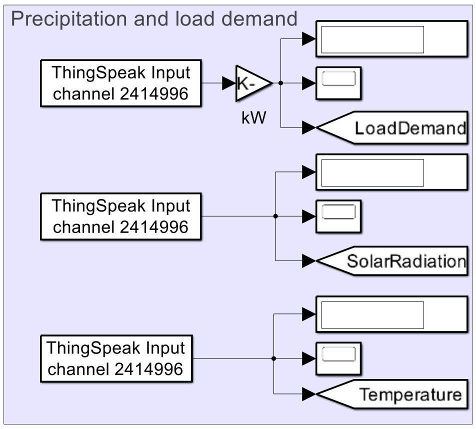


Figure 3.21: Data Transmission from ThingSpeak Cloud to MATLAB Simulink

Within MATLAB Simulink, the dynamic three-phase load block is equipped with an external port to receive active power and retrieve dynamic load demand data for the AC load. The "From Block" is employed to obtain data from its preceding "Goto Block" and transmit it to the dynamic load. Figure 3.22 exemplifies the setup of the dynamic AC load configuration.

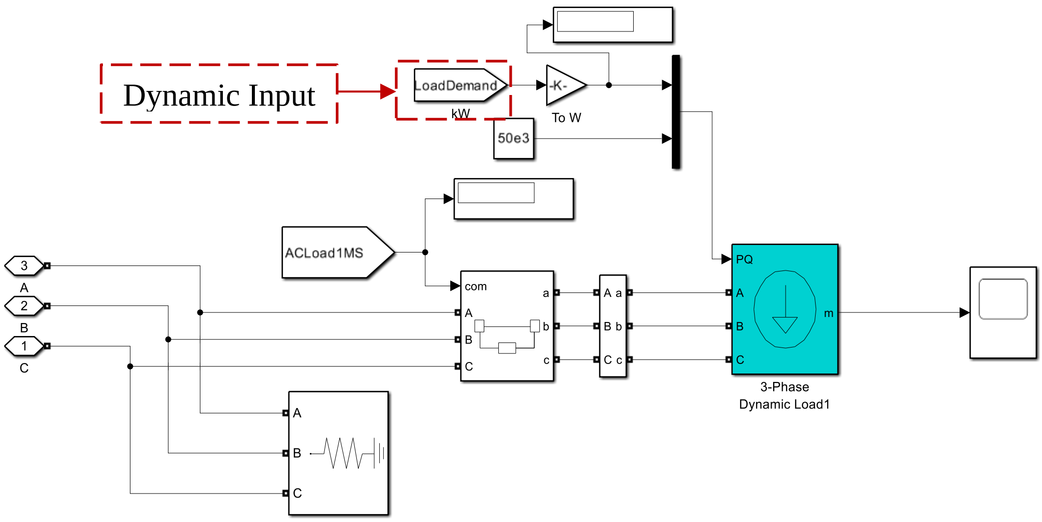


Figure 3.22: Dynamic AC Load Modeling

Figure 3.23 illustrates the configuration of the dynamic DC load. As MATLAB Simulink lacks a predefined DC dynamic load block, the dynamic DC load is devised using Ohm's Law, as outlined in Equation 3.54. Initially, the dynamic DC load modeling is conducted independently, without connecting it to the IGESs model. This approach ensures validation, confirming that the value of the dynamic input load demand from the “From Block” corresponds precisely to the output power scope. Following validation, the dynamic DC load is integrated into the IGESs. Notably, the designed MATLAB function block necessitates two inputs: Voltage, V, and Power, P. These input parameters are directed to the input port of the MATLAB function block, while the resistor, R corresponds to the value from the output port of the MATLAB function block.

$$R = f(V, P) = \frac{V^2}{P} \qquad \text{Equation 3.54}$$

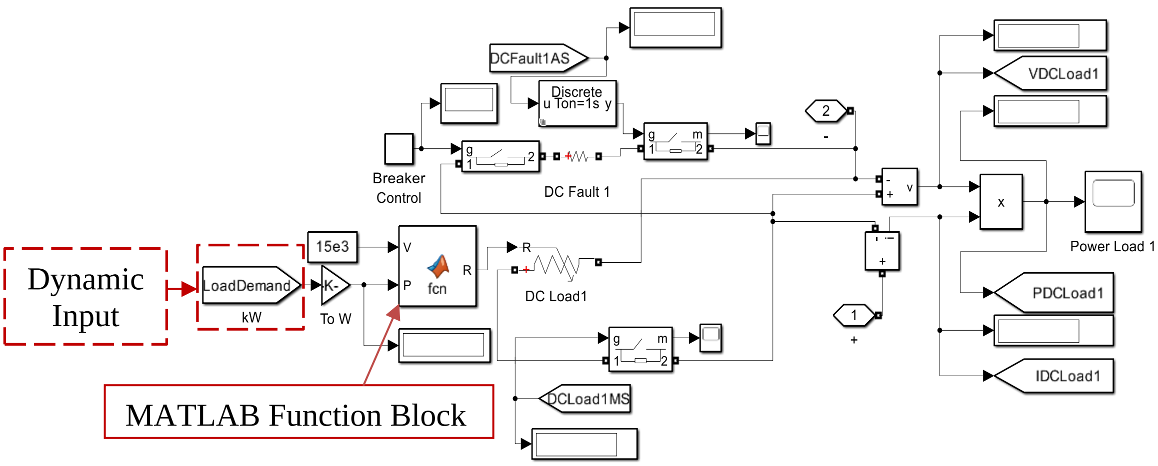


Figure 3.23: Dynamic DC Load Modeling

MATLAB includes the PV panel Simulink block, which features external ports designed to receive dynamic solar radiation and temperature data. The process of feeding solar radiation and temperature data to the PV panel is depicted in Figure 3.24.

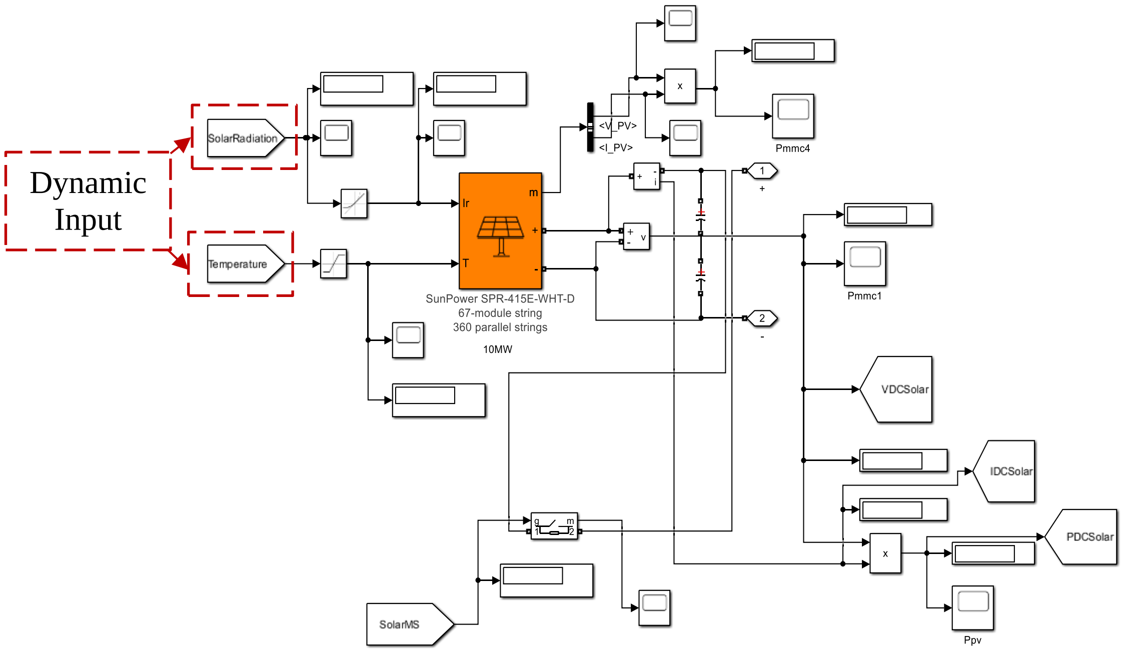


Figure 3.24: Dynamic PV Panel Modeling

3.5.3 Communication Framework of MATLAB Simulink and SCADA

This subsection showcases the detailed communication framework between MATLAB Simulink and SCADA for real-time monitoring and control. Facilitating the data transfer between MATLAB Simulink and Wonderware InTouch SCADA requires the intermediary server cloud KEPServerEX. The data transfer can occur bidirectionally, either from MATLAB to SCADA or vice versa, as illustrated in Figure 3.25.

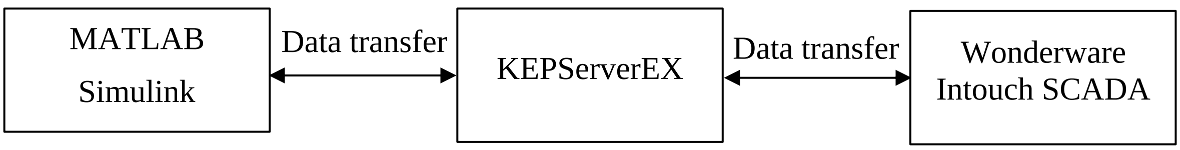


Figure 3.25: Data Transfer between MATLAB Simulink and SCADA

In the initial setup, establishing the communication channel is crucial. The channel type “Simulator” is assigned to “C1” upon adding a channel in KEPServerEX. Subsequently, the device wizard is configured with the name “D1”. Additionally, an alias “C1D1” is declared to facilitate connection from the client computer to the server. Table 3.14 provides an overview of the server settings utilized in KEPServerEX.

Table 3.14: Server Settings

Item	Details
Channel Name	C1
Device Name	D1
Alias Name	C1D1
Driver	Simulator
Model	16 Bit Device
ID Format	Decimal
Mapped to	C1.D1

The primary procedure involves adding tags to the server, where each tag represents a unique parameter for real-time monitoring, control, and automation. Firstly, each tag possesses a distinct name for differentiation purposes. Secondly, the available data type options include string, boolean, char, bytes, short, word, long, float, double, and more, depending on the specific data type being utilized. Furthermore, an address must be assigned to each tag, with formats varying depending on the data type. Lastly, the user selects the client access functionality, thereby allowing the client computer to either read or write data exclusively. Figure 3.26 provides an overview of the characteristics of the server tag.

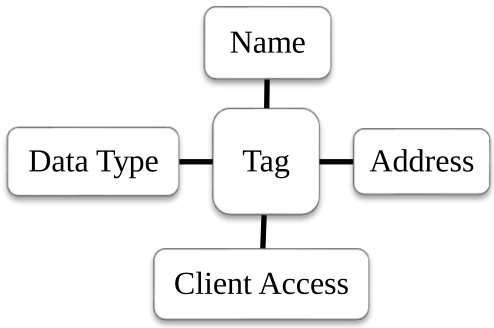


Figure 3.26: Characteristics of Server Tag

After establishing the server configuration, a monitoring framework is devised to collect crucial data from the IGESs model. This framework encompasses the readings of various components including DC solar generation, DC energy storage, DC load 1, DC load 2, AC wind generation, AC hydro generation, AC load 1, and AC load 2, as illustrated in Figure D-2 (Appendix). The Root Mean Square (RMS) values of voltage and current from the monitoring framework are calculated using Equations 3.55 and 3.56, respectively. These "Goto Blocks" containing the reading data within the monitoring framework are now prepared to be integrated into the server for further processing and analysis.

$$V_{rms} = \frac{V_{peak}}{\sqrt{2}} \qquad \text{Equation 3.55}$$

$$I_{rms} = \frac{I_{peak}}{\sqrt{2}} \qquad \text{Equation 3.56}$$

Manual and automatic switches, along with their corresponding signal tags, have been integrated into the sub-models of green energy generations, energy storage, AC and DC loads, and AC and DC faults within the IGESs model. This integration facilitates both manual control and automation functionalities. Manual control allows for switching operations triggered by signals, particularly for electrical power lines maintenance. Conversely, the automatic switch serves as an overcurrent relay, ensuring the safety of the

IGESs by opening the switch or circuit breaker in case of a fault. The integration of manual and automatic switches, along with their respective signal tags, into the sub-models is visually depicted in Figures D-3 (a) to (d) and Figures D-4 (a) to (f). In these representations, “MS” denotes the Manual Switch, while “AS” represents the Automatic Switch. The subsequent step involves utilizing OPC to establish a connection with the server. The channel C1, and its corresponding device D1, store all the necessary tags for communication during the data transfer process within the server cloud. The readings parameters from the “From Block” in Figure D-2 (Appendix) are then transmitted to the OPC Write block with unique tags from T1 to T55. This facilitates data transfer to the SCADA system via the server for real-time monitoring. The data type used for monitoring purposes is “double”, accurately representing the reading value. Furthermore, the OPC Read block is connected to unique tags from TA to TL to enable manual and automation control within the SCADA system. These tags are then linked to the “Goto Block” to receive control signals from the SCADA system via the server. The data type associated with these tags in the OPC Read block is logical, allowing for toggling between 0 and 1 for control purposes. Figure 3.27 illustrates the configuration framework for OPC setup in MATLAB Simulink.

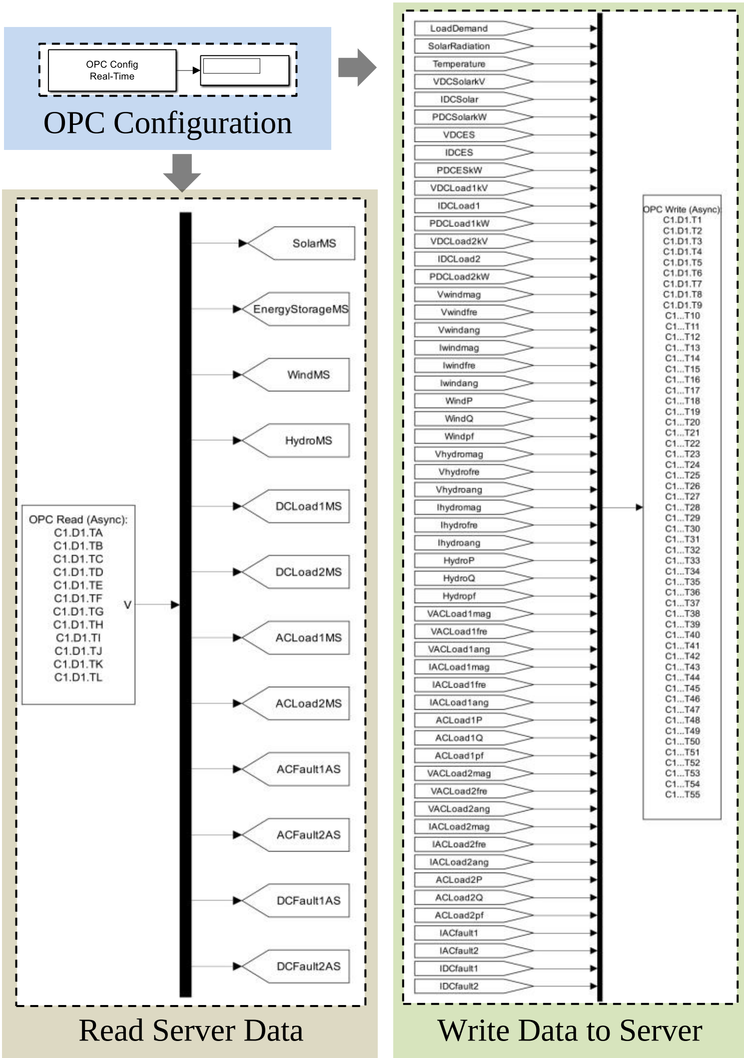


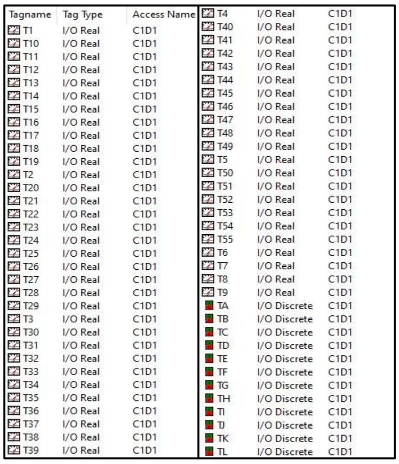
Figure 3.27: OPC Framework in MATLAB Simulink

The SCADA interface is constructed using Wonderware InTouch Maker. In the initial setup, the "Access Names" from the "Special" section are incorporated into the settings as delineated in Table 3.15. To ensure recognition of the channel from the server by the SCADA platform, the topic name is configured to match the alias name (as referenced in Table 3.14). To facilitate communication over the network, the SuiteLink protocol is employed. The application name must be set as "server_runtime" when establishing a connection to KEPServerEX via SuiteLink.

Table 3.15: Access Names Settings

Item	Details
Access Name	C1D1
Application Name	server_runtime
Topic Name	C1D1
Protocol	SuiteLink

Subsequently, the tags (T1 to T55) and (TA to TL) retrieved from the server are specified in the “Tagname Dictionary” within the “Special” section. Within the “Tagname Dictionary”, each tag is allocated a unique tag name. The tag type is designated as “I/O Real” for monitoring purposes, while for controlling purposes, the tag type is specified as “I/O Discrete”, representing discrete data. All these tags have the Access Name “C1D1”. Figure 3.28 illustrates the definition of all the tags utilized in the SCADA system.



Tagname	Tag Type	Access Name
T1	I/O Real	C1D1
T10	I/O Real	C1D1
T11	I/O Real	C1D1
T12	I/O Real	C1D1
T13	I/O Real	C1D1
T14	I/O Real	C1D1
T15	I/O Real	C1D1
T16	I/O Real	C1D1
T17	I/O Real	C1D1
T18	I/O Real	C1D1
T19	I/O Real	C1D1
T2	I/O Real	C1D1
T20	I/O Real	C1D1
T21	I/O Real	C1D1
T22	I/O Real	C1D1
T23	I/O Real	C1D1
T24	I/O Real	C1D1
T25	I/O Real	C1D1
T26	I/O Real	C1D1
T27	I/O Real	C1D1
T28	I/O Real	C1D1
T29	I/O Real	C1D1
T3	I/O Real	C1D1
T30	I/O Real	C1D1
T31	I/O Real	C1D1
T32	I/O Real	C1D1
T33	I/O Real	C1D1
T34	I/O Real	C1D1
T35	I/O Real	C1D1
T36	I/O Real	C1D1
T37	I/O Real	C1D1
T38	I/O Real	C1D1
T39	I/O Real	C1D1
T4	I/O Real	C1D1
T40	I/O Real	C1D1
T41	I/O Real	C1D1
T42	I/O Real	C1D1
T43	I/O Real	C1D1
T44	I/O Real	C1D1
T45	I/O Real	C1D1
T46	I/O Real	C1D1
T47	I/O Real	C1D1
T48	I/O Real	C1D1
T49	I/O Real	C1D1
T5	I/O Real	C1D1
T50	I/O Real	C1D1
T51	I/O Real	C1D1
T52	I/O Real	C1D1
T53	I/O Real	C1D1
T54	I/O Real	C1D1
T55	I/O Real	C1D1
T6	I/O Real	C1D1
T7	I/O Real	C1D1
T8	I/O Real	C1D1
T9	I/O Real	C1D1
TA	I/O Discrete	C1D1
TB	I/O Discrete	C1D1
TC	I/O Discrete	C1D1
TD	I/O Discrete	C1D1
TE	I/O Discrete	C1D1
TF	I/O Discrete	C1D1
TG	I/O Discrete	C1D1
TH	I/O Discrete	C1D1
TI	I/O Discrete	C1D1
TJ	I/O Discrete	C1D1
TK	I/O Discrete	C1D1
TL	I/O Discrete	C1D1

Figure 3.28: Tags Definition

Once all tags have been set up, the system proceeds to design the user interface. The Wonderware InTouch SCADA system provides flexibility, allowing users to create interfaces featuring embedded industrial graphics. The SCADA energy management system dashboard has been developed, comprising three main sections: monitoring, manual control, and automation. For each monitoring screen, the display color, maximum value, minimum value, and tag number have been configured. The tag number precisely follows the OPC configuration. Notably, only tags containing crucial parameters have been added to the SCADA energy management system dashboard panel, as depicted in Figure 3.29.

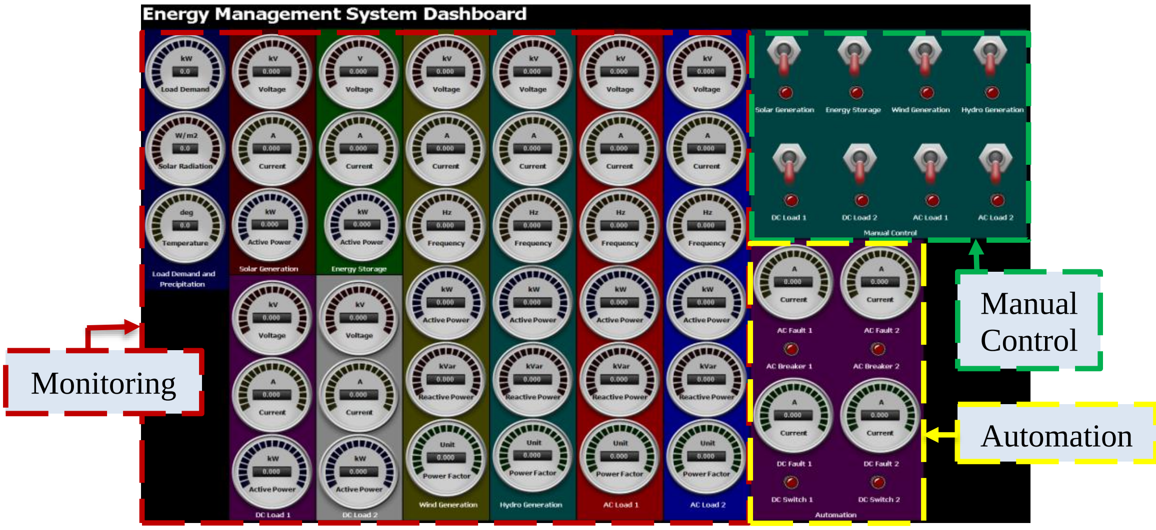


Figure 3.29: SCADA Energy Management System Dashboard

In the manual control segment, specific tags (TA to TH) have been designated to manual switches, each paired with corresponding light indicators. A green light signifies a closed switch, while a red light indicates an open switch. For automation, tags ranging from T52 to T55 have been incorporated into the monitoring screen to denote AC fault 1, AC fault 2, DC fault 1, and DC fault 2, respectively. Moreover, tags TI to TL have been assigned for the light indicators of AC breaker 1, AC breaker 2, DC switch 1, and DC switch 2. In this context, a green-light indicator implies that the breaker is open, while a red-light indicator denotes that the breaker is closed. The Wonderware Intouch SCADA system offers an additional advantage with its capability to utilize window scripts for automation. Preceding and succeeding a fault occurrence, the steady-state current flow behavior has been meticulously analyzed to establish the optimal current limits for the breakers and switches.

The window script acts as an overcurrent relay to activate the breaker, with its parameters determined by Equations 3.57 to 3.60. According to these equations, if the current reading of the tag is 200 A or lower, the breaker remains closed. However, if it exceeds 200 A, the breaker is opened.

$$If\ T52 \leq 200;\ then\ TI = 1;\ else\ TI = 0; \qquad \text{Equation 3.57}$$

$$If\ T53 \leq 200;\ then\ TI = 1;\ else\ TI = 0; \qquad \text{Equation 3.58}$$

$$If\ T54 \leq 200;\ then\ TK = 1;\ else\ TK = 0; \qquad \text{Equation 3.59}$$

$$If\ T55 \leq 200;\ then\ TK = 1;\ else\ TK = 0; \qquad \text{Equation 3.60}$$

3.5.4 Validation using Hardware Prototype

The proposed hardware prototype integrates with the SCADA system is proposed for experimental validation of real-time monitoring, control, and automation. A conceptual representation of the hardware prototype's operational workflow is illustrated in Figure 3.30.

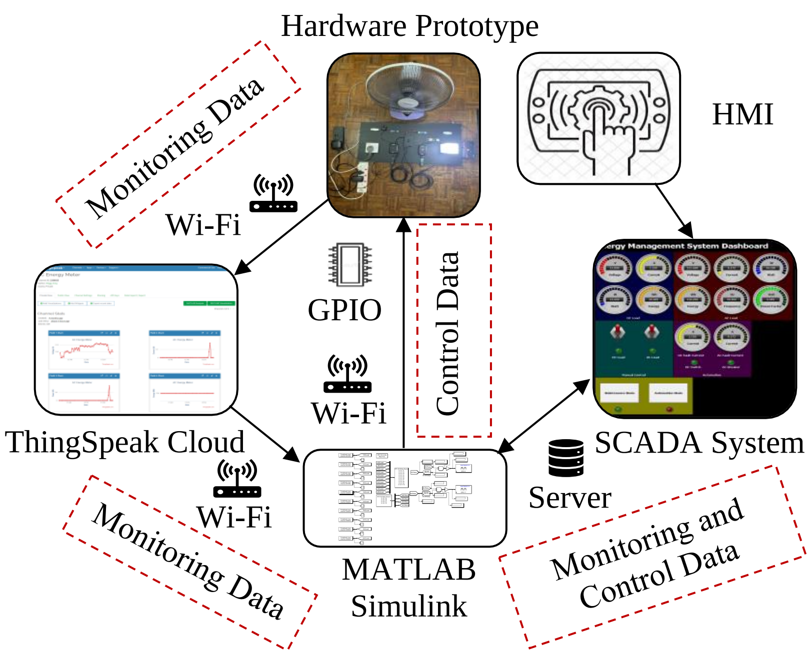


Figure 3.30: Operational Workflow of Hardware Prototype

Figure 3.31 illustrates the development of the hardware prototype in block diagram, which comprises several components: Raspberry Pi 4 model B, a Wi-Fi adapter to facilitate internet connectivity for the Raspberry Pi, a PZEM-016 AC meter, a table fan serving as the AC load (220 V), an AC switch, a 3.3 V AC relay, an AC supply (220 V), a PZEM-017 DC meter, a bulb acting as the DC load (12 V), a DC switch, a 3.3 V DC relay, a DC battery (12 V), and a DC shunt. It is worth noting that the relays utilized are 3.3 V, as the maximum voltage permitted for GPIO pins in the Raspberry Pi is 3.3 V. Subsequently, real-time measurement data for both AC and DC loads can be streamed to the ThingSpeak cloud using Python code executed on the Raspberry Pi.

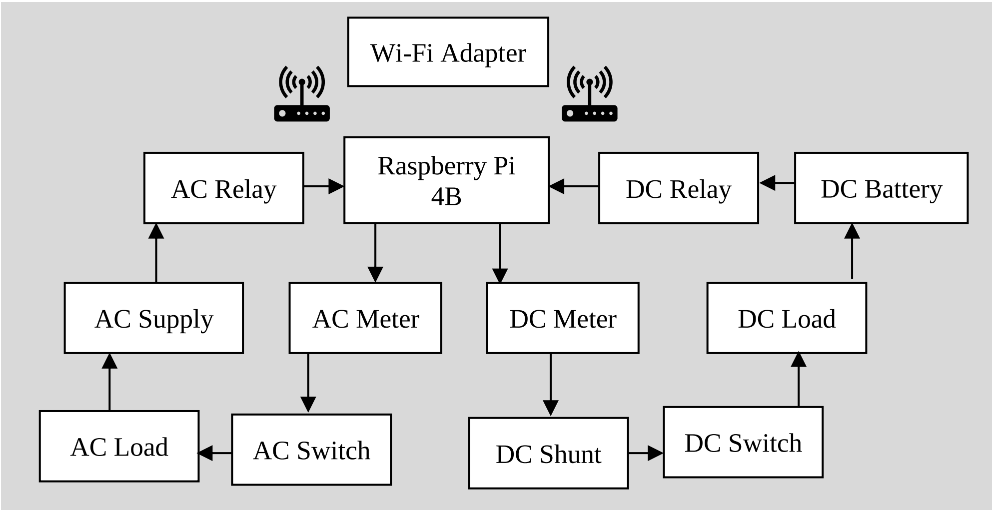


Figure 3.31: Hardware Prototype Development

The communication system in MATLAB Simulink is configured as depicted in Figure D-5 (Appendix). Monitoring data sourced from ThingSpeak is collected and transmitted to the SCADA system through an OPC server. Communication systems for both Manual Switch (MS) and Automatic Switch (AS) for AC and DC loads are devised within MATLAB Simulink. The NOT gate is employed to invert the signal because the relay closes the circuit when it equals 0 and opens when it equals 1. Furthermore, the AND gate is utilized to minimize the number of relays, ensuring that the relay only closes the circuit when both

“MS” and “AS” are equal to 1. A delay block is incorporated into the “AS” to ensure that the relay recloses the circuit after 1 second when the system encounters a fault.

The SCADA energy management system dashboard is designed as illustrated in Figure 3.32. This dashboard presents monitoring data for both AC and DC loads. For manual control, the maintenance mode must be selected to enable the switching of the AC and DC loads. Transitioning to automation mode is achieved by toggling the maintenance mode button. During automation mode, manual switches are disabled, and the relay automatically closes the circuit when the system current falls below the set threshold current. These functionalities are executed using the SCADA system’s condition and application scripts. Tag T2 represents the DC current reading, while tag T6 represents the AC current reading. Boolean tags TA, TB, TC, and TD are assigned for DC manual switch, DC automatic switch, AC manual switch, and AC automatic switch, respectively. Tag TE is set to 1 when maintenance mode is activated and 0 when maintenance mode is deactivated. Additionally, tag TF equals 1 when automation mode is enabled and 0 when automation mode is disabled.

The current threshold is established based on the predetermined normal behavior of current flow, as described in Equations 3.61 to 3.64. The current thresholds for both AC and DC maintenance switches (TA and TB) are set to a high value (20 A) as these switches are manual and should not operate automatically. Conversely, the current thresholds for both AC and DC automatic switches are set to 2 A and 0.2 A, respectively, reflecting the typical steady-state DC current of approximately 1.2 A and AC current of approximately 0.17 A.

$$If\ T2 \leq 20;\ then\ TA = 1;\ else\ TA = 0; \qquad \text{Equation 3.61}$$

$$If\ T6 \leq 20;\ then\ TB = 1;\ else\ TB = 0; \qquad \text{Equation 3.62}$$

$$If\ T2 \leq 2;\ then\ TC = 1;\ else\ TC = 0; \qquad \text{Equation 3.63}$$

$$If\ T6 \leq 0.2;\ then\ TD = 1;\ else\ TD = 0; \qquad \text{Equation 3.64}$$

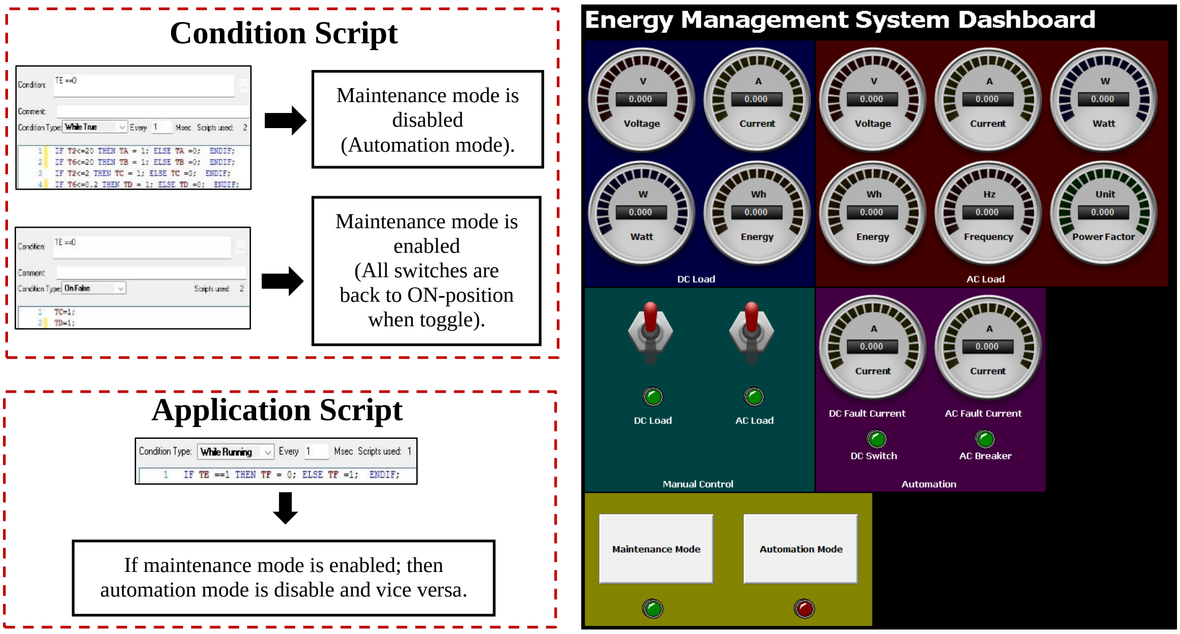


Figure 3.32: SCADA Dashboard Settings and Configurations

3.6 Chapter Summary

This chapter presents the proposed GIS-based fuzzy TOPSIS and filtration algorithms for identifying green energy locations. It further details the development of a GIS-driven fuzzy TSP-BIP algorithm for integrating these locations and the establishment of an IIoT-based system for real-time monitoring, control, and automation of IGESs using MATLAB Simulink, a server, and a SCADA system. All key models, mathematical equations, and formulations are thoroughly presented and discussed. The modeling for identification and integration of green energy locations is compared and validated against state-of-the-art research, while the IIoT-based system is validated through hardware prototypes. These evaluations confirm the reliability and robustness of the proposed methods.

CHAPTER 4

RESULTS AND DISCUSSION

4.1 Introduction

This research findings are categorized into three main sections. The first section demonstrates the results of identifying optimal green energy locations for SES, WES, and HES. The process begins with the first layer of filtration, where optimal HES are obtained, and WES with wind speeds greater than 3 ms^{-1} are considered optimal. All optimal WES and HES are then presented on maps. The second phase of filtration, using criteria constraints, further refines the potential SES. The second layer applies a novel GIS-based fuzzy TOPSIS algorithm model to rank the SES, revealing the top 100 optimal SES coordinates. These optimal SES are then validated using a weighted sum method, and the agreement level of the results is determined and discussed.

Once optimal green energy locations have been identified, they are clustered into 12 divisions and an improved GIS-driven fuzzy TSP-BIP algorithm model is utilized to integrate them by designing optimal power lines routing. The outcomes of this optimal electrical power line routing for each division are presented and compared with those of the conventional TSP-BIP algorithm. These results are then ranked to ascertain the superiority of fuzzy logic operations in TSP algorithms. Additionally, the performance results of TSP-BIP algorithm against other TSP algorithms are evaluated by comparing the fuzzy values of each method and considering computational time. This validation aims to discover the effectiveness of fuzzy TSP-BIP algorithm in comparison to existing TSP algorithms.

The last section presents the successful modeling of real-time dynamic data initially. Subsequently, these real-time dynamic data were integrated into the IGESs model to facilitate comprehensive real-time monitoring, control, and automation using simulation models. The monitoring results from MATLAB Simulink were then compared with those of the SCADA system, with manual control and automation strategies executed to demonstrate the robustness of the proposed model. To validate the simulation model's applicability in real world scenarios, Raspberry Pi and IIoT components were incorporated and integrated with the SCADA systems to achieve real-time monitoring, control, and automation strategies. The monitoring data results were presented, manual control functionality was demonstrated, and automation operations were tested for fault clearance. All these results are presented and discussed accordingly. Consequently, the research findings, from identification through integration to real-time monitoring, control, and automation for IGESs, stand as valuable assets for decision-makers or stakeholders aiming to efficiently harness and manage IGESs.

4.2 Results of Green Energy Locations Identification

This section presents the results derived from three pivotal phases: the first layer, second layer, and ensuing validation procedure. Initially, 19,237 coordinates for SES and WES, along with 155 coordinates for HES, served as inputs. These coordinates underwent a systematic sequential filtration process, which is meticulously detailed and scrutinized for both the first and second filtration phases. Post-filtration, the refined set of options underwent evaluation utilizing the fuzzy TOPSIS algorithm. This algorithm facilitated the establishment of a comprehensive ranking for the top 100 optimal SES. A comparison ensued against the solar suitability map generated using weighted sum methods within ArcGIS Pro. The validation process primarily ensures the reliability of the results, with researchers employing the developed algorithm to conduct analyses aimed at identifying green energy locations.

4.2.1 First Phase of Filtration

The initial filtration process commenced by segregating 11 polygon shapefiles sourced from the NextGIS database into two principal categories: exclusion areas and inclusion areas. Exclusion areas encompassed various features such as protected areas, points of interest, settlements, parking areas, surface features, vegetation, land use, buildings, airports, and bodies of water. Conversely, there was only one inclusion area, denoting an island. Python-based filtration techniques were systematically applied in 11 consecutive iterations to eliminate potential SES and WES associated with either exclusion or inclusion areas from the initial count of 19,237 coordinates. Simultaneously, the same 10 polygon shapefiles, excluding bodies of water, were utilized to filter the 155 potential HES. Table 4.1 and Figures 4.1 illustrate the first phase of filtration for SES and WES, respectively, while Table 4.1 and Figure 4.1 delineate the first phase of filtration for HES.

Table 4.1: Breakdown of First Filtration Process for SES and WES

Round	Area	Type	Removed Locations	Balance
0	Initial State		0	19,237
1	Protected Area	Exclusion	102	19,135
2	Point of interest	Exclusion	22	19,113
3	Settlement	Exclusion	3	19,110
4	Parking	Exclusion	0	19,110
5	Surface	Exclusion	0	19,110
6	Vegetation	Exclusion	383	18,727

Table 4.1 continued

7	Land Use	Exclusion	335	18,392
8	Island	Inclusion	248	18,144
9	Building	Exclusion	15	18,129
10	Airport	Exclusion	5	18,124
11	Water	Exclusion	897	17,227

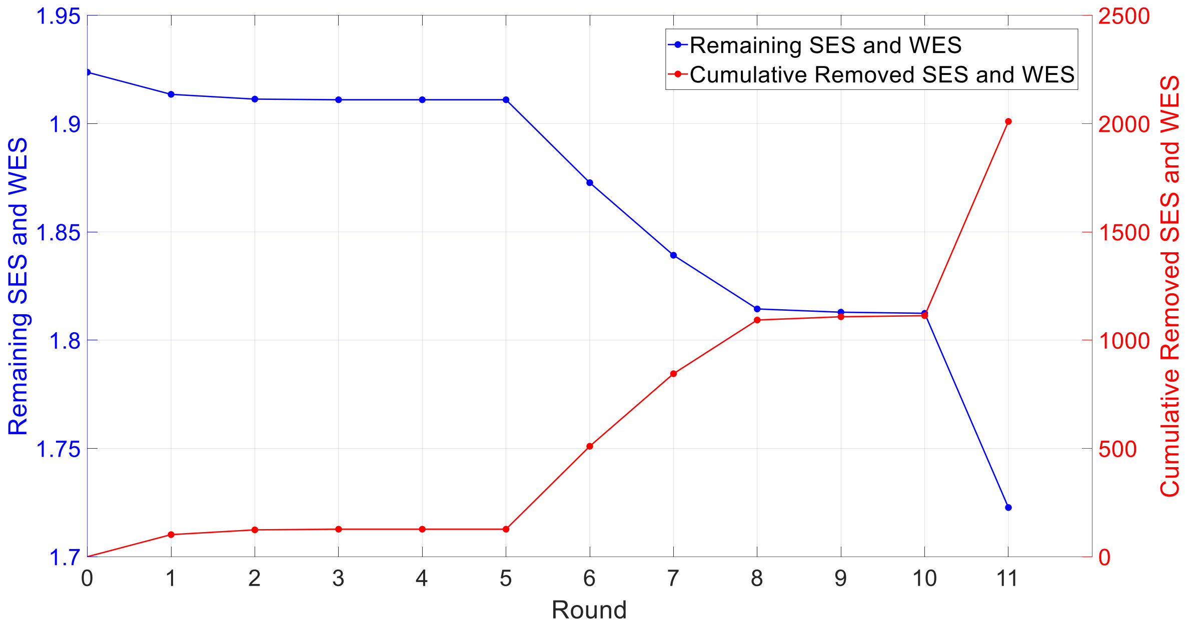


Figure 4.1: Results of Remaining and Cumulative Removed SES and WES

Figure 4.1 indicates that rounds 1 to 5 and rounds 9 to 10 had minimal effect on the available SES and WES, suggesting their limited influence on the dataset. Conversely, rounds 6 to 8 and round 11 had a considerable impact, resulting in substantial exclusions of unsuitable SES and WES. The most notable reduction transpired in round 11, where 897 locations were discarded due to their overlap with bodies of water. Consequently, this initial filtration process effectively streamlined the available SES and WES, yielding a more manageable set of 17,227 coordinates.

Table 4.2: Breakdown of First Filtration Process for HES

Round	Area	Type	Removed Locations	Balance
0	Initial State		0	155
1	Protected Area	Exclusion	3	152
2	Point of interest	Exclusion	0	152
3	Settlement	Exclusion	0	152
4	Parking	Exclusion	0	152
5	Surface	Exclusion	0	152
6	Vegetation	Exclusion	14	138
7	Land Use	Exclusion	0	138
8	Island	Inclusion	0	138
9	Building	Exclusion	0	138
10	Airport	Exclusion	0	138

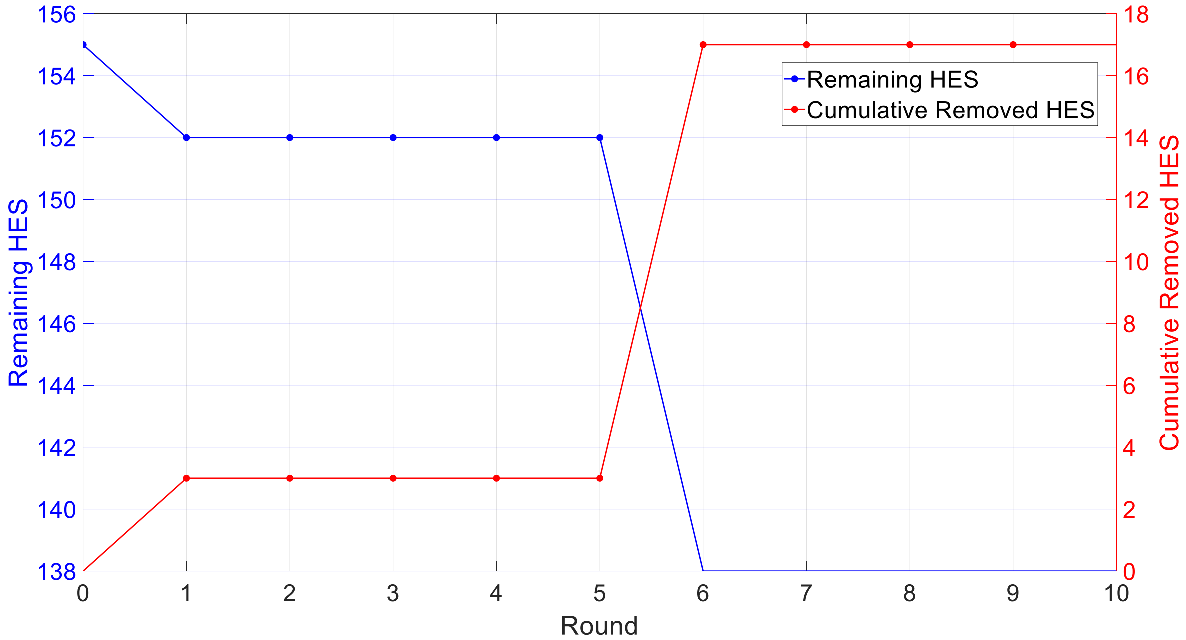


Figure 4.2: Results of Remaining and Cumulative Removed HES

As the potential HES is retrieved from the SEB database, some of the HES might now be unsuitable for hydro generation as the place may have been developed. According to Figure 4.2, where there are only 155 HES in the beginning. There are 3 HES that overlapped with protected areas in round 1, while 14 HES are in vegetation zones in round 6. These HES are removed in the filtration process. It is imperative to note that the water bodies polygon shapefile is excluded from the filtration process for HES as all the HES should be situated on the water areas. After the first phase of the filtration process, the optimal 138 HES remain.

From the first phase of the filtration process, the remaining 17,227 coordinates with wind speeds greater than 3 ms^{-1} are selected as the optimal WES. There are only 23 WES above 3 ms^{-1}. As wind speeds in Sarawak are generally low, preserving some potential WES is valuable for site assessment and further evaluation. Additionally, advancements in technology allow wind turbines to operate at low wind speeds, as low as 2 ms^{-1}, making harnessing low wind speed areas possible. Figure 4.3 demonstrates the remaining 17,227 potential SES and WES. Furthermore, Figure 4.4 showcases the optimal 23 WES, while Figure 4.5 depicts the optimal 138 HES.

According to the findings depicted in Figure 4.3, the elimination process demonstrates rapid operation, with computational times consistently below 10 seconds, even when managing a large-scale of locations (19,237 coordinates). The algorithm offers flexibility by allowing users to incorporate additional raster layers as needed, thus facilitating the development of personalized models.

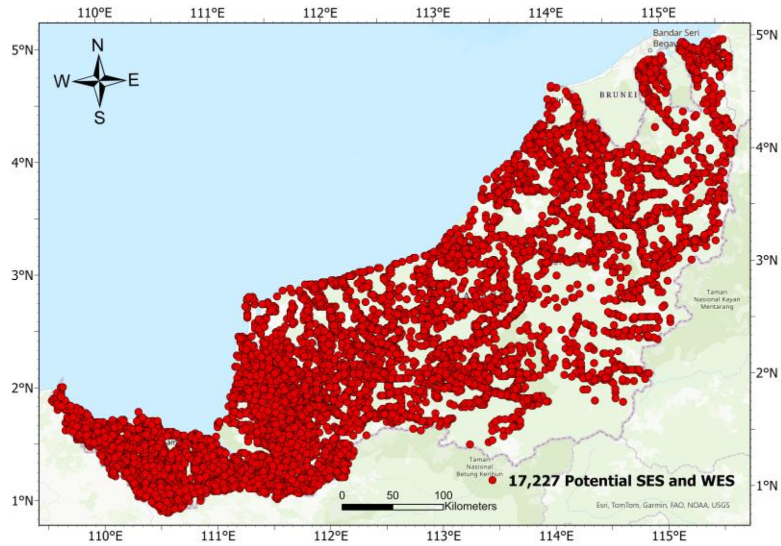


Figure 4.3: Remaining 17,227 Potential SES and WES

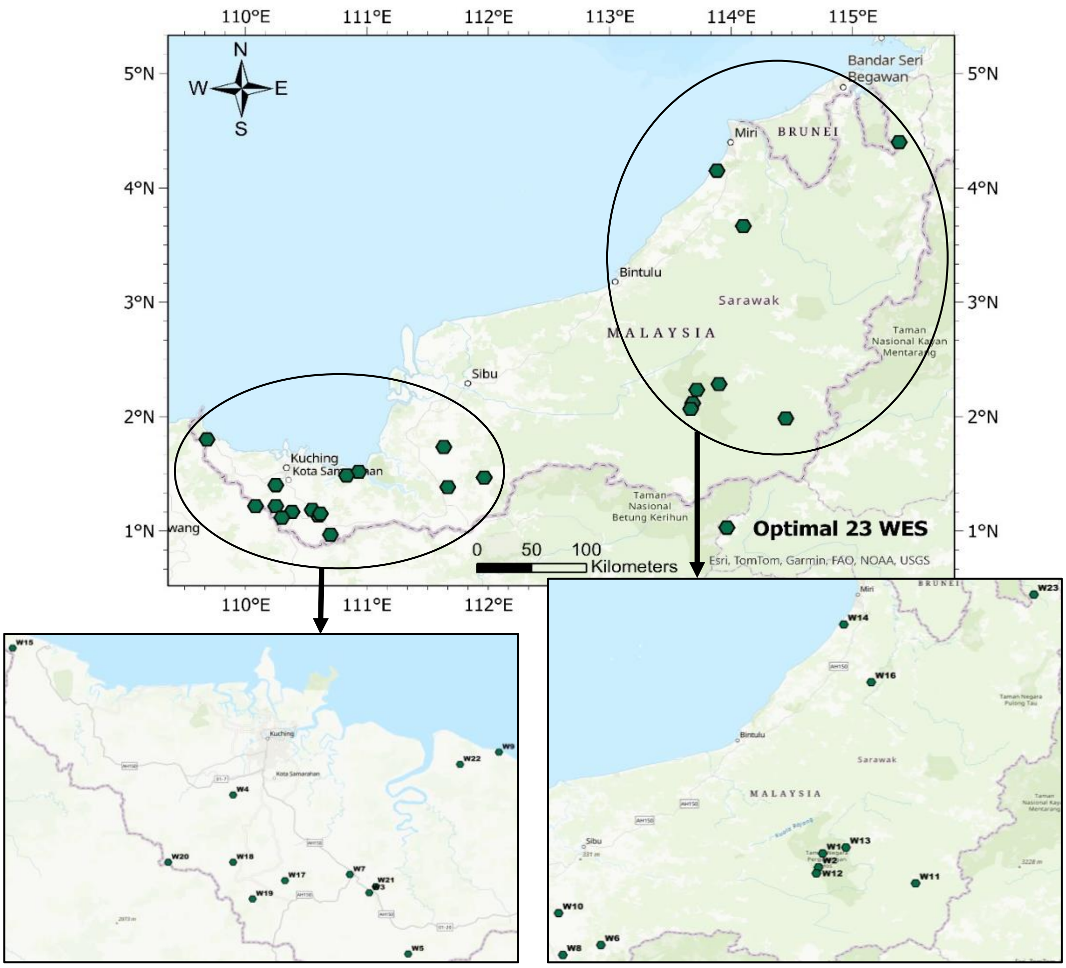


Figure 4.4: Optimal 23 WES

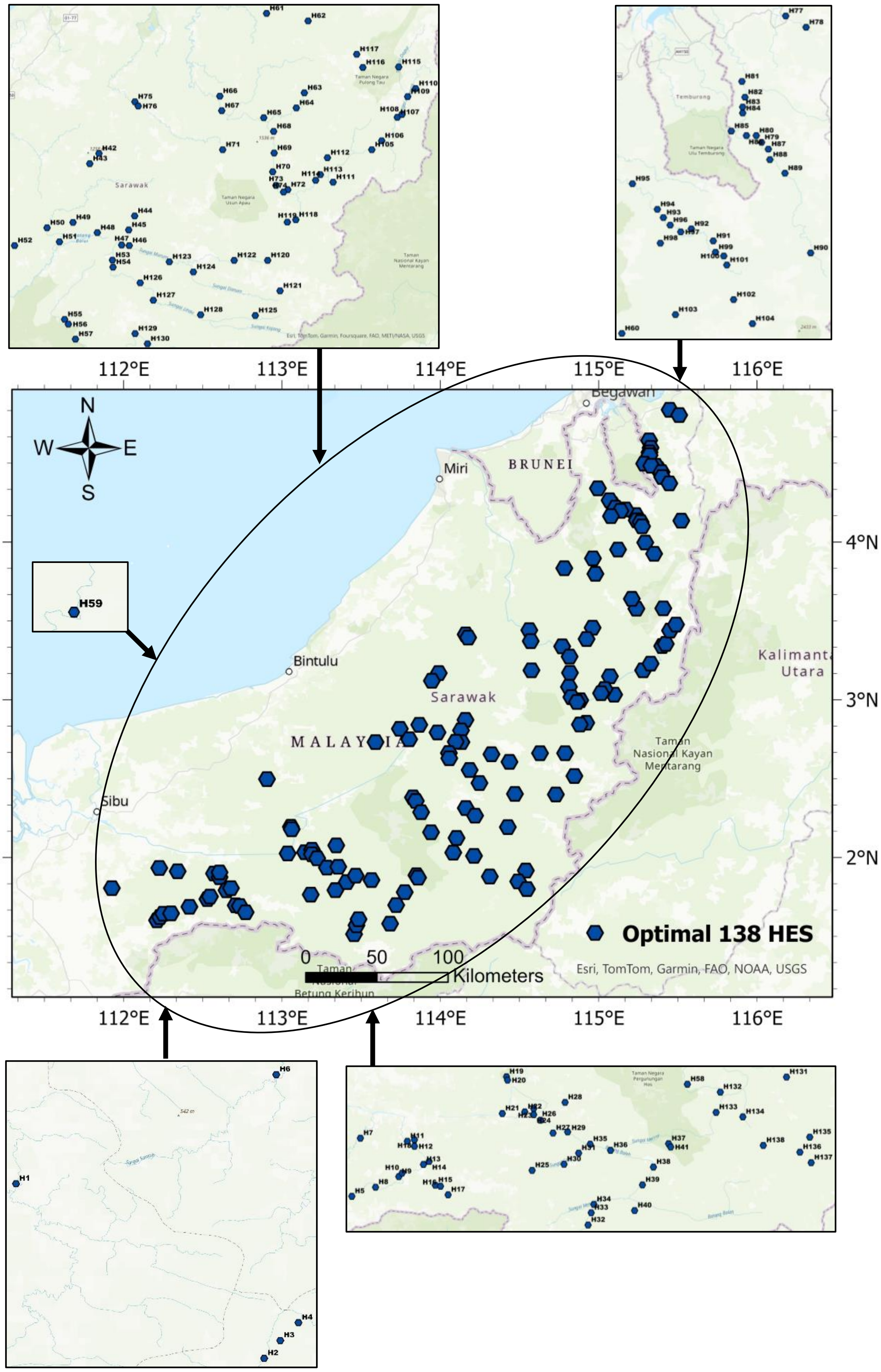


Figure 4.5: Optimal 138 HES

4.2.2 Second Phase of Filtration

The secondary filtration phase further refines the 17,227 coordinates based on predetermined criteria constraints, as outlined in Table 3.3. This stage comprised 19 rounds, each tailored to specific criteria aimed to filter SES. Factors such as solar radiation, temperature, slope, elevation, proximity to electrical power lines, roads, residential areas, urban facilities, distance from water bodies, protected areas, settlements, and population density were considered to exclude locations falling outside the defined constraints. During the initial rounds (1 and 2), where the constraint $C_1 \geq 1200$ and $C_2 \geq 15$ was imposed, no SES were disqualified. However, in round 3, introducing the constraint $C_2 < 28$ resulted in the removal of 47 SES, leaving 17,180 viable options. Subsequent rounds applied constraints such as $C_3 < 25$ in round 4, eliminating 46 SES, and $C_4 < 2200$ in round 5, removing 112 SES. Round 6 with constraint $C_4 \geq 0$, maintained the number of viable SES. Notably, rounds 7 and 8 played a pivotal role, eliminating 1,406 and 2,430 SES, respectively, with constraints $C_5 \geq 0.01$ and $C_5 < 50$. The filtration process continued with constraints like $C_6 \geq 0.1$, $C_7 < 45$, and $C_8 < 45$, resulting in significant reductions. Round 9 witnessed a substantial decrease of 9,088 SES due to the constraint $C_6 \geq 0.1$. Conversely, rounds 10, 17, and 18 had no impact on the number of SES, reflecting less restrictive criteria. In the final round, only 1 SES was excluded when applying the constraint $C_{12} > 0$. The filtration approach showcased meticulousness and effectiveness in systematically narrowing down the initial pool, culminating in a final count of 1,862 filtered SES. The progressive reduction in SES count after each round is detailed in Table 4.3 and Figure 4.6, while Figure 4.7 visually depicts the remaining 1,862 SES, slated for utilization in fuzzy TOPSIS algorithms to identify the optimal 100 SES.

Table 4.3: Breakdown of Second Filtration Process for SES

Round	Criteria Constraint	Removed Locations	Balance
0	Initial State	0	17,227
1	$C_1 \geq 1200$	0	17,227
2	$C_2 \geq 15$	0	17,227
3	$C_2 < 28$	47	17,180
4	$C_3 < 25$	46	17,134
5	$C_4 < 2200$	112	17,022
6	$C_4 \geq 0$	0	17,022
7	$C_5 \geq 0.01$	1406	15,616
8	$C_5 < 50$	2430	13,186
9	$C_6 \geq 0.1$	9088	4,098
10	$C_6 < 50$	0	4,098
11	$C_7 \geq 0.3$	5	4,093
12	$C_7 < 45$	178	3,915
13	$C_8 \geq 0.3$	4	3,911
14	$C_8 < 45$	1214	2,697
15	$C_9 \geq 0.1$	215	2,482
16	$C_9 < 20$	619	1,863
17	$C_{10} \geq 0.1$	0	1,863
18	$C_{11} \geq 0.1$	0	1,863
19	$C_{12} > 0$	1	1,862

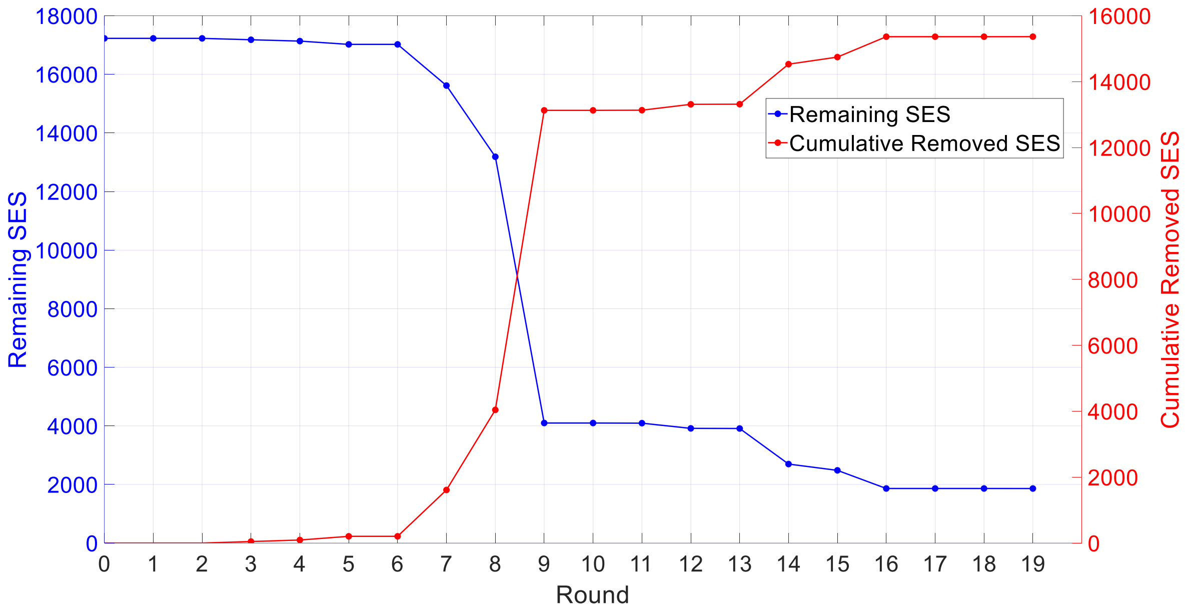


Figure 4.6: Results of Remaining and Cumulative Removed SES

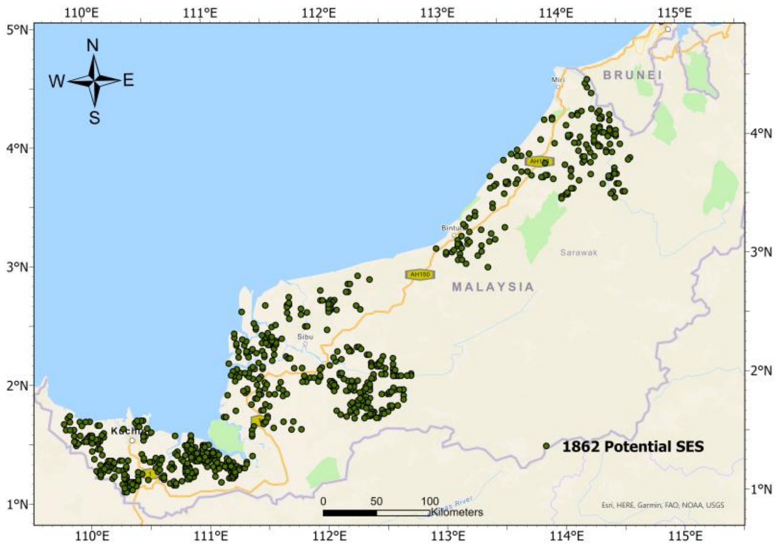


Figure 4.7: Filtered 1,862 potential SES

The filtration algorithms employed in the second phase provide a versatile method, seamlessly integrating with pre-established constraints within the model. Their adaptability to diverse raster maps affords decision-makers the flexibility to precisely define constraints for each criterion. Furthermore, the computational efficiency of the execution is remarkably rapid. Even when managing a substantial dataset of SES, comprising 17,227 entries from the initial filtration process, this second filtration algorithm reduces SES to 1,862, accomplishing the task within a mere 5-second timeframe.

4.2.3 Top 100 Optimal SES

The fuzzy TOPSIS algorithm is proposed to identify the optimal 100 SES among the 1,862 pre-screened alternatives. This proposed algorithm not only gauges the proximity to the positive ideal solution but also takes into account the proximity to the negative ideal solution. Consequently, it yields Closeness Coefficients, CC_i, encompassing all pertinent considerations from the influential criteria raster maps, thereby reflecting the proximity of each SES to the optimal solution. These devised fuzzy TOPSIS algorithms afford a lucid interpretation and boast high transparency in presenting the top 100 optimal SES. Specifically, these CC_i values span from 0 (representing the least desirable SES) to 1 (indicating the most favorable SES), with the selection of the top 100 CC_ivalues representing the optimal SES. Each SES is allocated a distinctive identifier, commencing with S1 for the one exhibiting the highest closeness coefficient and incrementally progressing to S100 for the SES possessing the lowest closeness coefficient within the top 100 ranking. Figure 4.8 visually elucidates the continuum of closeness coefficients ranging from S1 to S100.

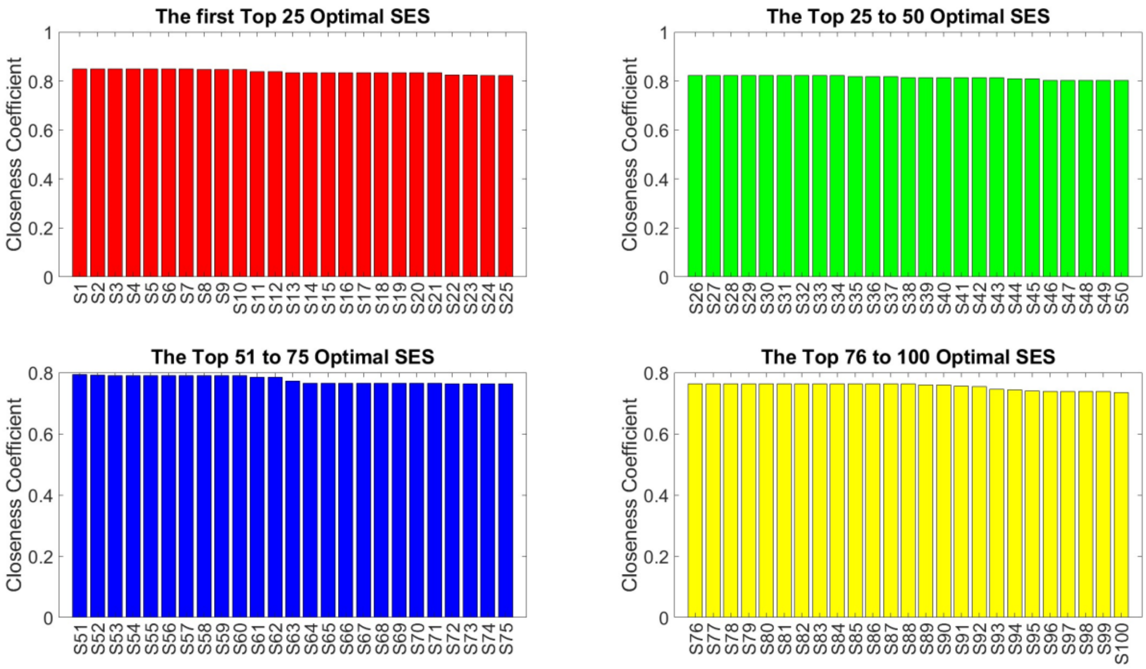


Figure 4.8: Closeness Coefficient of Top 100 Optimal SES

The innermost outcome entails the mapping of the top 100 optimal SES. An intriguing pattern emerges among these top 100 optimal SES, indicating a tendency to form several discernible clusters. However, these clusters present challenges in terms of clear visualization. To address this issue, the map has been zoomed in to improve the clarity in identifying these top 100 optimal SES. It is noticed that these SES fall into three distinct clusters: the first cluster is situated in close proximity to Kuching and Samarahan, the second cluster is predominantly concentrated around Sibu, Kapit, and Mukah, while the last cluster is centered somewhere between Bintulu and Miri. It is noteworthy that, unlike numerous prior studies which primarily focus on areas with solar potential, the findings provided herein offer a more comprehensive and specific delineation of SES throughout the interior of the Sarawak region. Figure 4.9 provides a detailed visual representation of the top 100 optimal SES within the Sarawak region.

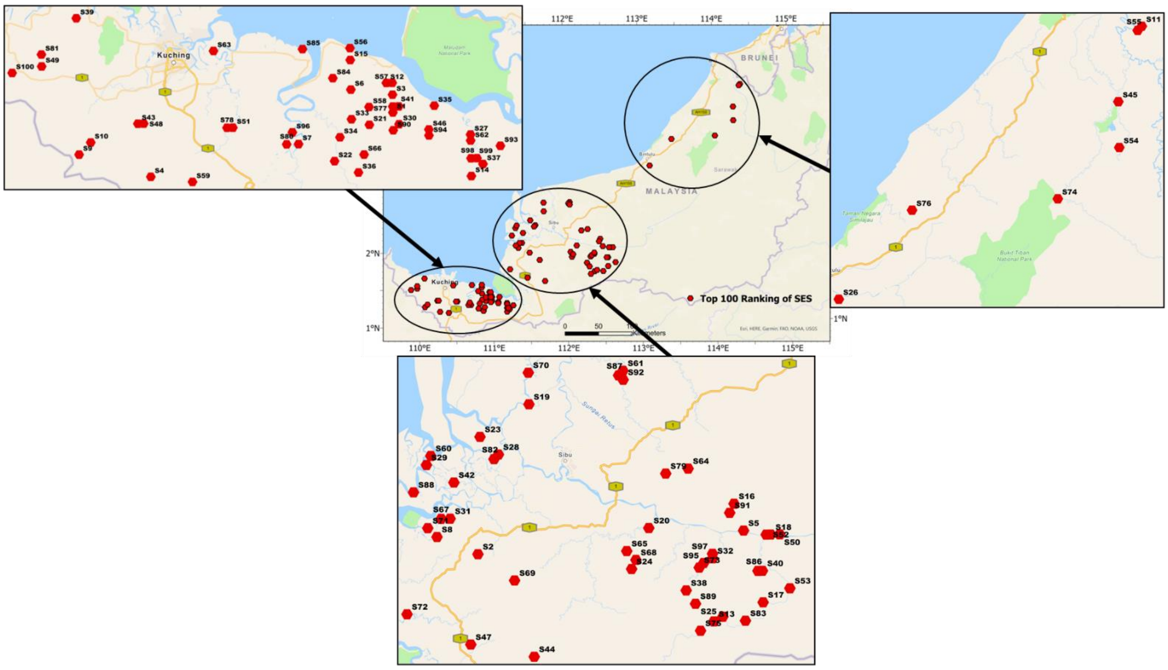


Figure 4.9: Top 100 Optimal SES

4.2.4 Validation Results for Top 100 Optimal SES

The 12 criteria raster maps are initially standardized into a consistent scale spanning from 1 to 32 in order to validate the findings regarding the top 100 optimal SES. This standardization process employs the classify tool to ensure maximum sensitivity in generating each criterion raster map. Subsequently, these maps are amalgamated with their respective weights, determined through the fuzzy weight method employing the weighted average approach. The weighted sum method is then utilized to merge all 12 criteria raster maps (scaled from 1 to 32) along with their associated weights, thereby producing a solar suitability map for validation purposes as depicted in Figure 4.10. The resultant solar suitability map showcases a spectrum of weight values. The location with the lowest recorded weight value (2,074.36) is identified as the least suitable for SES implementation, whereas the site with the highest weight value (2,979.36) is deemed the most suitable. Furthermore, Figure 4.11 combines the top 100 optimal SES derived from proposed method with the solar suitability map, revealing a substantial alignment between most of the top 100 optimal SES and the solar hotspots delineated on the solar suitability map.

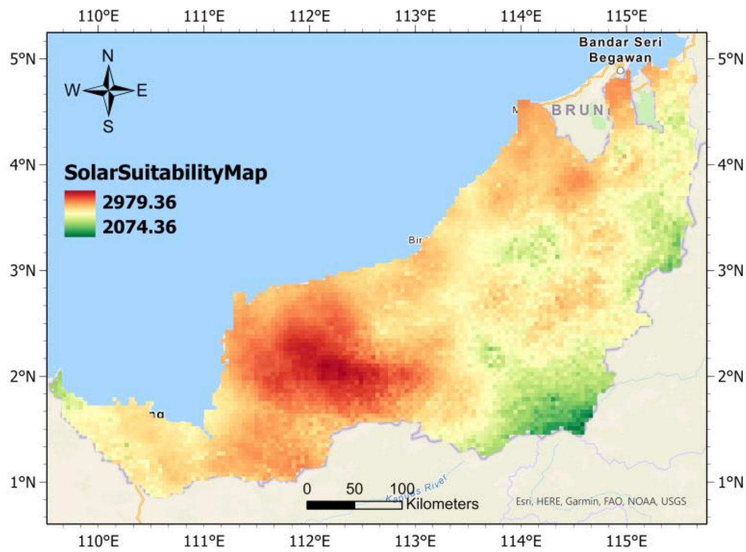


Figure 4.10: Solar Suitability Map

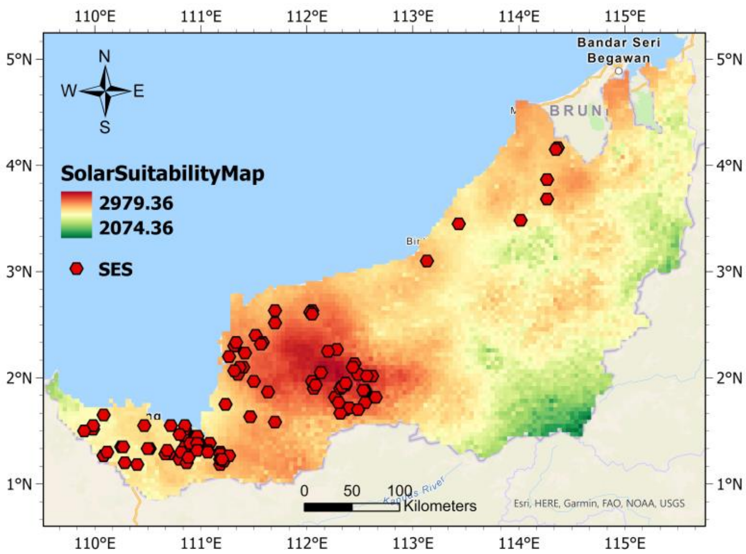


Figure 4.11: Solar Suitability Map with Top 100 Optimal SES

The Python code in Figure B-2 (Appendix) is employed to extract the weights of the top 100 optimal SES from the solar suitability map for a comprehensive comparative analysis. This extraction of weight values is essential for assessing their proximity to the maximum weight value of 2,979.36. The weights generated for each of the top 100 optimal SES are depicted in Figure 4.12.

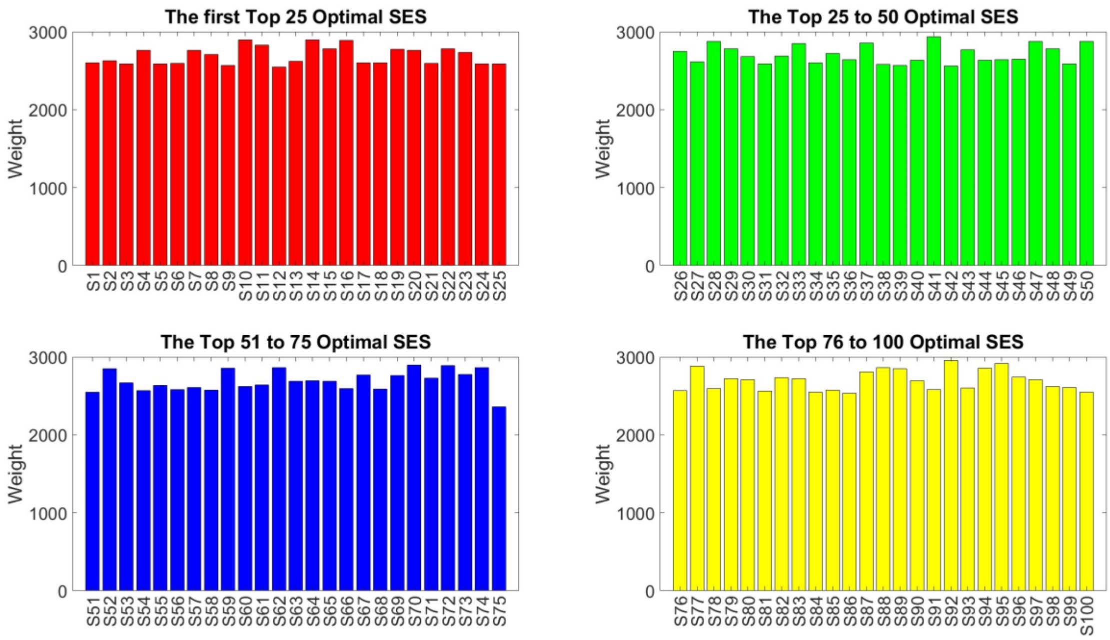


Figure 4.12: Generated Weight for Top 100 Optimal SES

As the weights of the top 100 optimal SES locations are extracted, Equation 3.22 is applied to calculate the relative percentages of each SES in relation to the best and worst values, as illustrated in Figure 4.13.

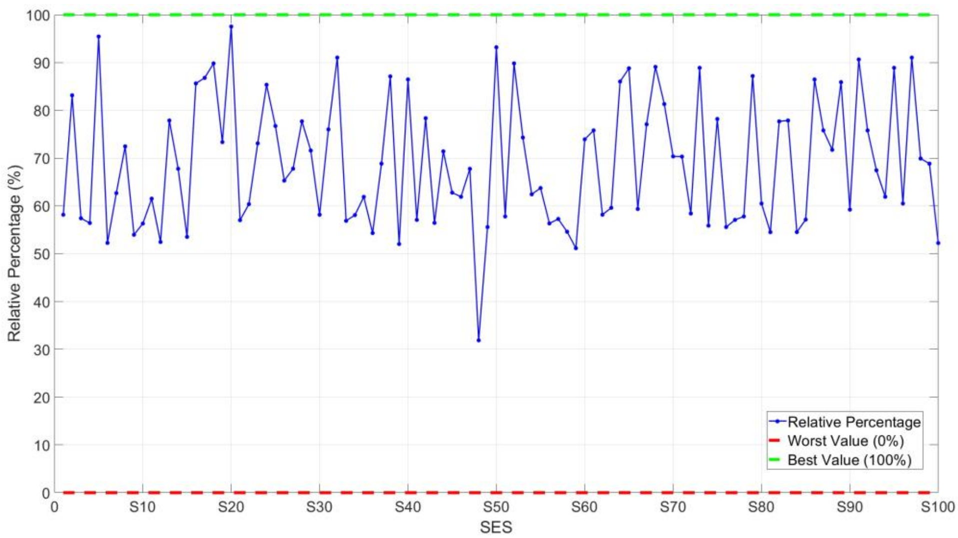


Figure 4.13: Relative Percentage for Top 100 Optimal SES

Figure 4.13 offers a visual representation of the relative percentages, ranging from approximately 31.87 % for S48 to nearly 97.53 % for S20. Notably, only one SES falls below the 50 % threshold. Several SES, including S20, S5, S50, S32, S97, and S91, exhibit notably high sustainability levels, with percentages approaching or exceeding 90 %. A distinct clustering pattern emerges, particularly within the 60 % to 70 % range, suggesting shared influencing factors on their suitability. Various factors contribute to the observed variability in SES percentages compared to the optimal value of 100 %. Primarily, the solar suitability map originates from the weighted sum method, focusing solely on criterion raster layers without incorporating filtration processes. Conversely, the proposed method integrates filtration procedures, accommodating structured data and criteria constraints. Consequently, highly weighted alternatives may be excluded if they fail to meet these constraints during filtration. Additionally, limitations inherent in the weighted sum analysis tool within ArcGIS, lacking support for fuzzy triangular membership weights, may introduce deviations in results. The proposed method leverages fuzzy weights, renowned for their effectiveness in addressing ambiguity in decision-making processes. Hence, based on this analysis, the proposed GIS-based fuzzy TOPSIS and filtration algorithms are validated against a solar suitability map generated via the weighted sum method, affirming their capacity to produce dependable outcomes. Specifically, 99 SES demonstrate suitability percentages surpassing 50 %, with only one SES falling below this threshold. Moreover, the average relative percentage computed for all 100 SES is 69.01 %, indicating a moderately high level of agreement among the top 100 optimal SES concerning their suitability for SES implementation.

4.3 Results of Green Energy Locations Integration

The identified optimal green energy locations of 100 SES, 23 WES, and 138 HES are considered for integration by designing the optimal electrical power lines routing within a ring network for each respective cluster. The validation results including minimum distance, minimum elevation difference, and minimum average GFD were conducted utilizing proposed GIS-driven fuzzy TSP-BIP algorithm against the conventional TSP-BIP algorithm. Furthermore, the outcomes obtained from the proposed methodology underwent comparison with seven alternative fuzzy TSP algorithms, facilitating a comprehensive assessment of its efficiency and performance.

4.3.1 Optimal Electrical Power Lines Routing Design

The green energy locations are grouped into 12 clusters based on geographical divisions: Kuching, Samarahan, Serian, Sri Aman, Betong, Sarikei, Sibu, Mukah, Bintulu, Kapit, Miri, and Limbang. Within each division, green energy locations are further categorized for SES, WES, and HES. Table 4.4 illustrates the clustering of these green energy locations by divisions, presenting their respective IDs from C_1 to C_{12}, and the corresponding number of sites in each cluster.

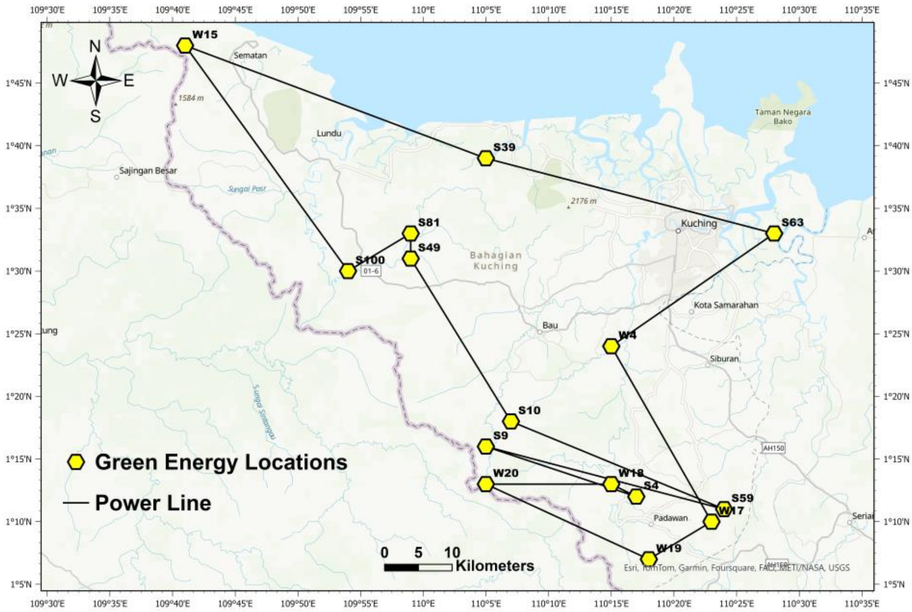
Table 4.4: Clustering Green Energy Locations by Divisions

Cluster	Division	Type	ID	Total
C_1	Kuching	SES	S4, S9, S10, S39, S49, S59, S63, S81, S100	15
		WES	W4, W15, W17, W18, W19, W20	
		HES	N/A	
C_2	Samarahan	SES	S1, S3, S6, S7, S12, S15, S21, S22, S30, S33, S34, S35, S36, S41, S46, S56, S57, S58, S66, S77, S80, S84, S85, S90, S94, S96	28
		WES	W9, W22	
		HES	N/A	
C_3	Serian	SES	S43, S48, S51, S78	8
		WES	W3, W5, W7, W21	
		HES	N/A	
C_4	Sri Aman	SES	S14, S27, S37, S62, S93, S98, S99	9
		WES	W6, W8	
		HES	N/A	
C_5	Betong	SES	S44, S47, S72	4
		WES	W10	
		HES	N/A	
C_6	Sarikei	SES	S2, S8, S31, S67, S69, S71	7
		WES	N/A	
		HES	H1	
C_7	Sibu	SES	S19, S20, S24, S32, S38, S65, S68, S70, S73, S89, S95, S97	14
		WES	N/A	
		HES	H6, H7	
C_8	Mukah	SES	S23, S28, S29, S42, S60, S61, S64, S79, S82, S87, S88, S92	12
		WES	N/A	
		HES	N/A	

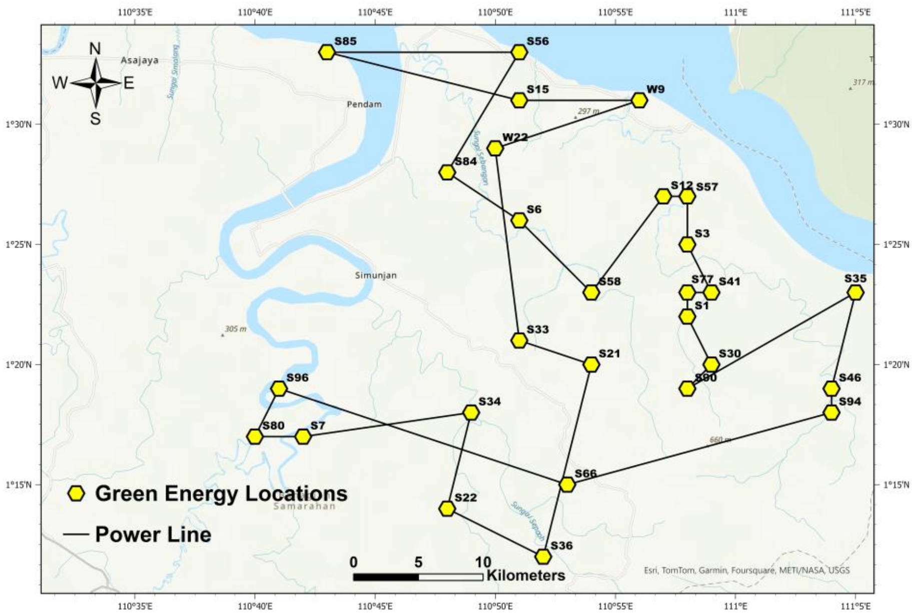
Table 4.4 continued

C_9	Bintulu	SES	S26, S76	4
		WES	N/A	
		HES	H52, H59	
C_{10}	Kapit	SES	S5, S13, S16, S17, S18, S25, S40, S50, S52, S53, S75, S83, S86, S91	92
		WES	W1, W2, W11, W12, W13	
		HES	H2, H3, H4, H5, H8, H9, H10, H11, H12, H13, H14, H15, H16, H17, H18, H19, H20, H21, H22, H23, H24, H25, H26, H27, H28, H29, H30, H31, H32, H33, H34, H35, H36, H37, H38, H39, H40, H41, H42, H43, H44, H45, H46, H47, H48, H49, H50, H51, H53, H54, H55, H56, H57, H58, H120, H121, H122, H123, H124, H125, H126, H127, H128, H129, H130, H131, H132, H133, H134, H135, H136, H137, H138	
C_{11}	Miri	SES	S11, S45, S54, S55, S74	40
		WES	W14, W16	
		HES	H60, H61, H62, H63, H64, H65, H66, H67, H68, H69, H70, H71, H72, H73, H74, H75, H76, H103, H105, H106, H107, H108, H109, H110, H111, H112, H113, H114, H115, H116, H117, H118, H119	
C_{12}	Limbang	SES	N/A	28
		WES	W23	
		HES	H77, H78, H79, H80, H81, H82, H83, H84, H85, H86, H87, H88, H89, H90, H91, H92, H93, H94, H95, H96, H97, H98, H99, H100, H101, H102, H104	

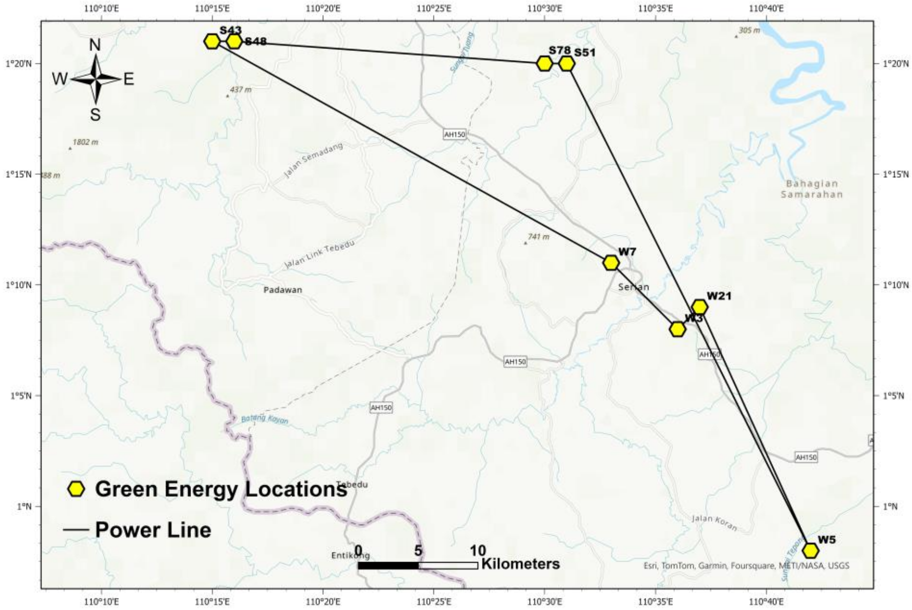
The integration of these clustered green energy locations is accomplished through the establishment of optimal routing for electrical power lines in each cluster, utilizing the proposed GIS-driven fuzzy TSP-BIP algorithm. Figures 4.14 (a) to (l) illustrate the mapping of green energy location integration in each cluster, with a specific focus on the design of optimal electrical power lines routing.



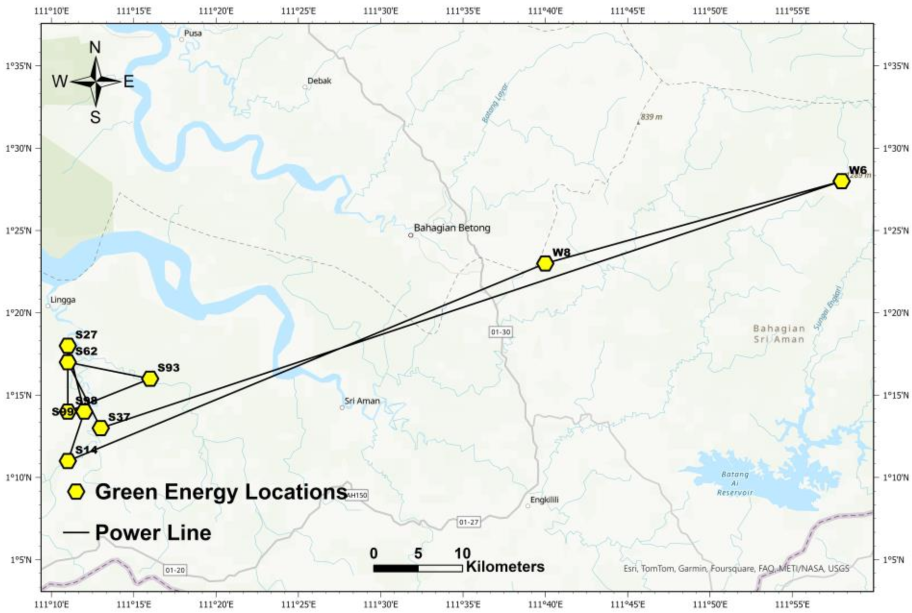
(a) Cluster 1: Kuching



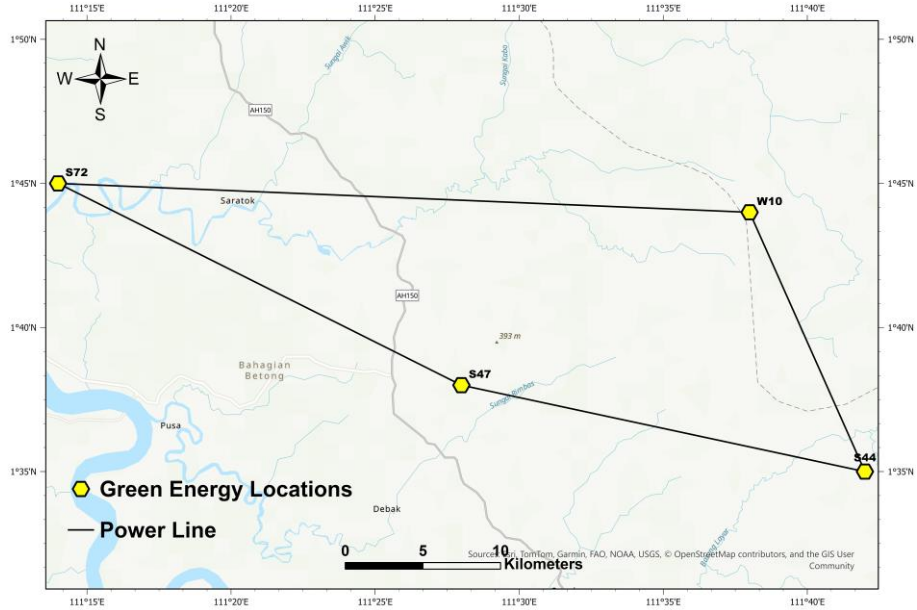
(b) Cluster 2: Samarahan



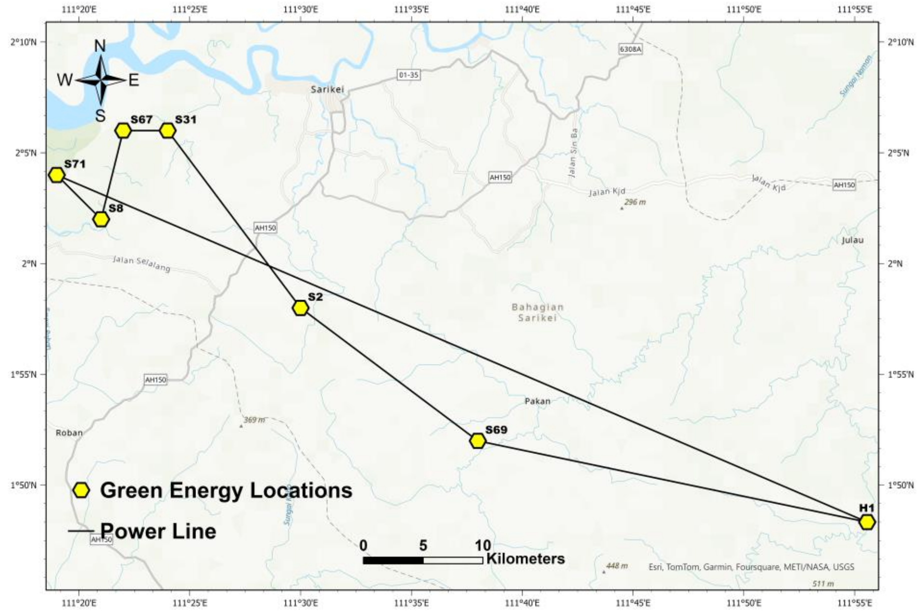
(c) Cluster 3: Serian



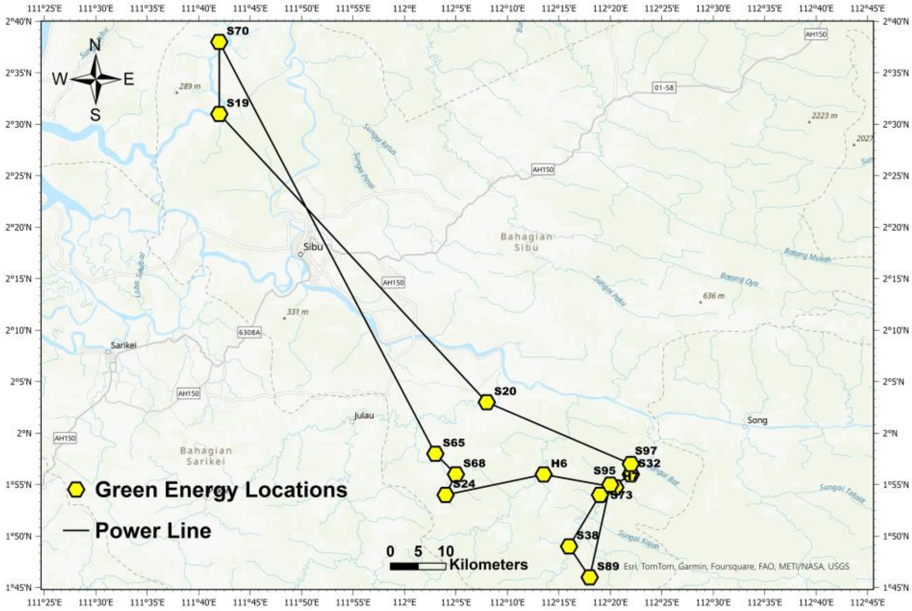
(d) Cluster 4: Sri Aman



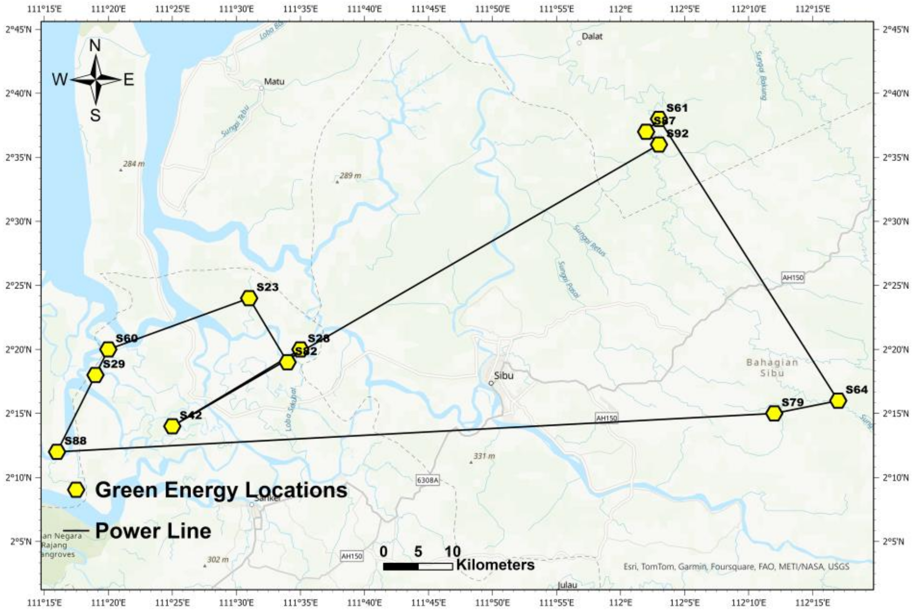
(e) Cluster 5: Betong



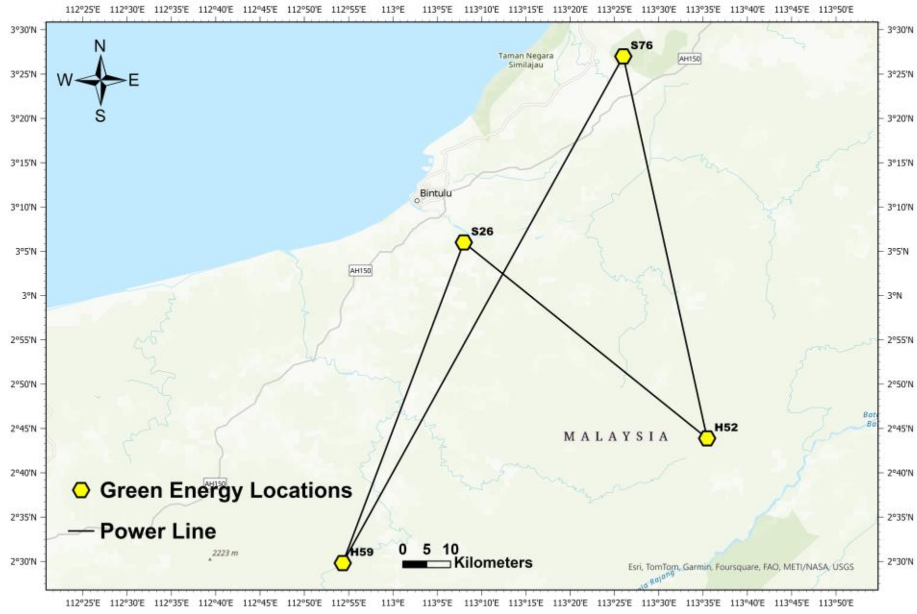
(f) Cluster 6: Sarikei



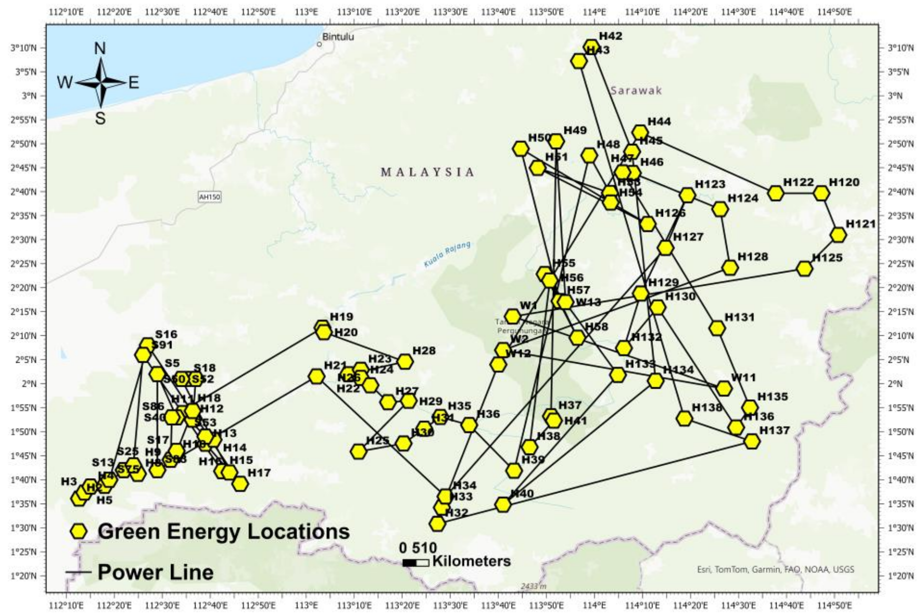
(g) Cluster 7: Sibu



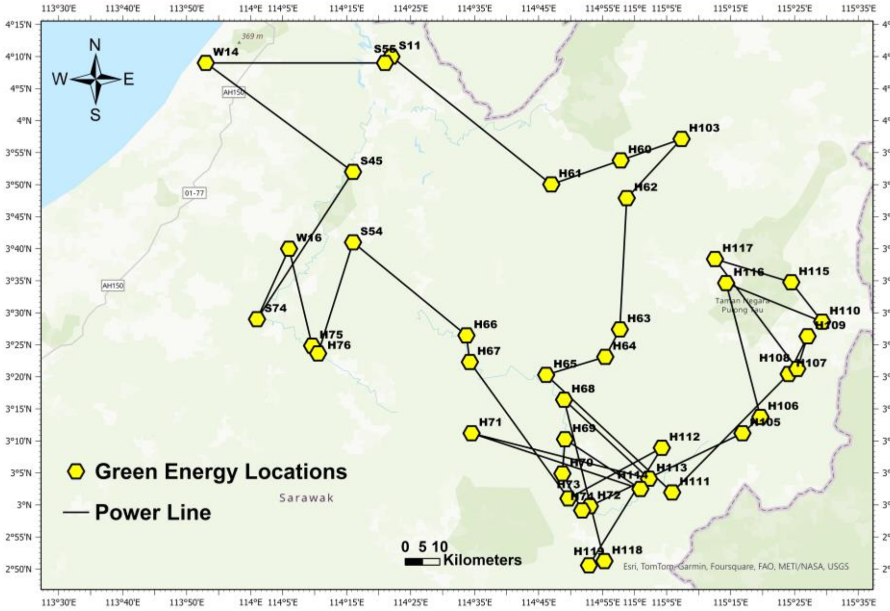
(h) Cluster 8: Mukah



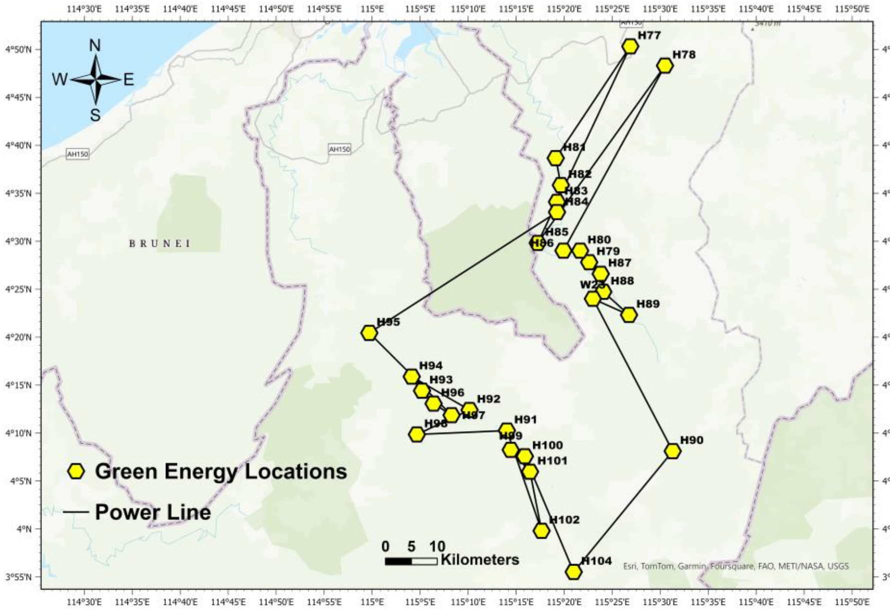
(i) Cluster 9: Bintulu



(j) Cluster 10: Kapit



(k) Cluster 11: Miri



(l) Cluster 12: Limbang

Figure 4.14: Optimal Electrical Power Lines Routing Design for 12 Clusters

The distribution of green energy locations across divisions reveals a varied landscape of green energy development within the region. Kapit emerges prominently with an extensive network of 92 green energy sites, followed closely by Miri with 40 locations. Samarahan and Limbang also exhibit strong emphasis on green energy, each boasting 28 locations. In contrast, Betong and Bintulu demonstrate lower levels of green energy infrastructure, each with only 4 sites. Sarikei, Serian, Sri Aman, Mukah, and Sibu fall within the mid-range, with 7, 8, 9, 12, and 14 green energy locations respectively.

The optimal design for electrical power lines routing in these divisions considers three key factors: minimum total distance, minimum elevation difference, and minimum total average ground flash density. Minimizing the distance between green energy locations is paramount for cost efficiency, reducing construction materials and energy loss during power delivery. Considering elevation differences is crucial, as flat or gently sloping terrain simplifies construction and reduces maintenance costs. Conversely, higher elevations increase installation difficulty and maintenance expenses. Additionally, decreasing the average GFD enhances safety and reliability by mitigating the risk of electrical power lines damage and downtime.

Each cluster adopts a ring topology for its simplicity, low maintenance costs, and reliability. In the event of a fault leading to power interruption in any section, continuity of power supply is ensured through alternative green energy locations feeding the system. Some results display overlapping lines, particularly in Figures 4.14 (j) and (k), attributed to the high number of green energy locations and consideration of multiple objective functions. Values for minimum total distance $min\ (\sum d)$ in km, minimum total elevation difference, $min\ (\sum \Delta e)$ in m and minimum total average ground flash density, $min\ (\sum \overline{GFD})$ in $flashes\ km^{-2}\ year^{-1}$ for each cluster C_i are provided in Table 4.5.

Table 4.5: Generated Results and TSP Configurations for 12 Clusters

C_i	Minimum			TSP Configuration
	$\sum d$	$\sum \Delta e$	$\sum \overline{GFD}$	
C_1	387.9479	3,100	389.4225	S10 → S81 → S100 → W17 → S49 → W15 → S63 → W18 → S39 → W19 → W4 → S59 → S9 → W20 → S4→ S10
C_2	247.4423	598	483.9316	S15 → S7 → S96 → S21 → S58 → S34 → W9 → S94 → S33 → S90 → S80 → S36 → S22 → W22 → S77 → S85 → S84 → S12 → S30 → S1 → S35 → S3 → S6 → S66 → S57 → S41 → S46 → S56→ S15
C_3	146.0911	1286	199.5101	S51 → S48 → W21 → W3 → S78 → W5 → W7 → S43→ S51
C_4	228.9218	2446	159.5003	S27 → W6 → S93 → S37 → S99 → S14 → S62 → W8 → S98→ S27
C_5	118.1112	698	74.5280	S47 → S72 → W10 → S44→ S27
C_6	160.3681	122	162.6671	H1 → S69 → S2 → S31 → S67 → S8 → S71 → H1
C_7	279.9132	480	261.5549	H6 → S70 → S65 → H7 → S73 → S95 → S24 → S20 → S89 → S97 → S19 → S68 → S38 → S32 → H6
C_8	321.0920	210	273.4130	S61 → S64 → S79 → S88 → S82 → S28 → S42 → S92 → S60 → S23 → S29 → S87 → S61

Table 4.5 continued

C_9	339.3953	1088	85.9135	S26 → H52 → S76 → H59 → S26
C_{10}	3,258.4543	5650	1,556.7928	H120 → S91 → H121 → H8 → S50 → S86 → H48 → H5 → S52 → H9 → S16 → S25 → S13 → H133 → H131 → S18 → S17 → H126 → H127 → H129 → H128 → H122 → H130 → S5 → S40 → H51 → H43 → H46 → H49 → H50 → H47 → H134 → H136 → H45 → H135 → S75 → H44 → H25 → H24 → H3 → H22 → H20 → H21 → H58 → H19 → W13 → H23 → W11 → H57 → H56 → H125 → H124 → H4 → H55 → H54 → H42 → H137 → H53 → H132 → H138 → S83 → H123 → S53 → H29 → H16 → H31 → H13 → H14 → H36 → H35 → H12 → H30 → W2 → W1 → W12 → H41 → H39 → H27 → H40 → H11 → H28 → H33 → H38 → H17 → H18 → H32 → H34 → H15 → H10 → H2 → H26 → H37 → H120
C_{11}	1,166.2549	4180	782.2271	H103 → H118 → H117 → H110 → H112 → H111 → H115 → H109 → H69 → H65 → H63 → H114 → H116 → H113 → H67 → H68 → S74 → S54 → H105 → H106 → H107 → H108 → H72 → H71 → H66 → H62 → H61 → H64 → H119 → H70 → H60 → H75 → H74 → S45 → W14 → H76 → W16 → S55 → H73 → S11 → H103
C_{12}	434.0510	2386	680.7809	H77 → H79 → H82 → H81 → H99 → H98 → H97 → H96 → H78 → H80 → H83 → H84 → H85 → H87 → H86 → H89 → W23 → H100 → H88 → H101 → H104 → H92 → H91 → H95 → H93 → H94 → H102 → H90 → H77

4.3.2 Validation Results of Proposed Method against Ordinary TSP-BIP Algorithm

This section discusses the role of fuzzy in the proposed method for enhancing multiple objective functions optimization. The results from the proposed method in Table 4.5 are compared with those achieved through the ordinary TSP-BIP algorithm. The ordinary TSP-BIP algorithm covers three distinct scenarios: optimal in D (minimizing total distance and measuring the configuration values of total elevation difference and total average GFD), optimal in ED (minimizing total elevation difference and measuring the configuration values of total distance and total average GFD), and optimal in AGFD (minimizing total average GFD and measuring the configuration values of total distance and total elevation difference). Tables 4.6 to 4.8 meticulously outline these ordinary TSP-BIP scenarios, facilitating a comprehensive performance evaluation of the proposed method.

Table 4.6: Optimal in D

C_i	$min\left(\sum d\right)$	$min\left(\sum \Delta e\right)$	$min\left(\sum \overline{GFD}\right)$
C_1	289.9556	4,032	392.6813
C_2	192.3793	1,014	490.3684
C_3	142.7217	1,330	200.1546
C_4	197.8871	2,464	164.5501
C_5	118.1112	698	74.5280
C_6	152.4521	122	162.6770
C_7	262.4169	676	264.7188
C_8	269.0974	250	285.6249
C_9	284.8119	1,088	83.6776
C_{10}	1,194.5187	12,722	1,668.9353

Table 4.6 continued

C_{11}	697.8175	5,952	814.1824
C_{12}	285.3759	3,932	701.8238

Table 4.7: Optimal in ED

C_i	$min\left(\sum d\right)$	$min\left(\sum \Delta e\right)$	$min\left(\sum \overline{GFD}\right)$
C_1	620.3646	1,904	407.5034
C_2	563.8661	538	474.2398
C_3	144.3771	1,286	200.1546
C_4	225.0316	2,426	161.6558
C_5	123.7642	698	74.5565
C_6	152.4521	122	162.6770
C_7	326.7989	426	267.7875
C_8	341.1870	210	279.3413
C_9	339.3953	1,088	85.9135
C_{10}	7,520.9489	3,140	1,608.4104
C_{11}	2,363.2775	2,222	811.96021
C_{12}	873.1632	1,768	681.2969

Table 4.8: Optimal in AGFD

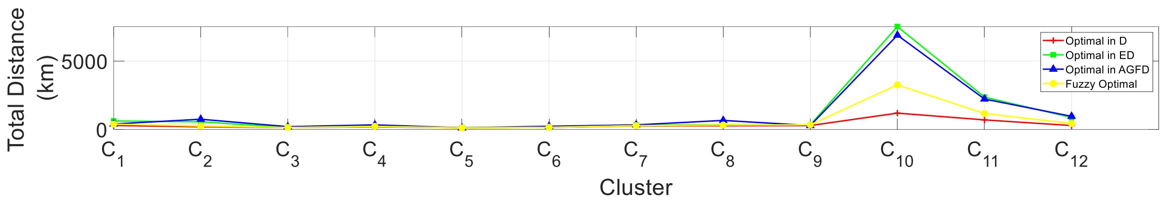
C_i	$min\left(\sum d\right)$	$min\left(\sum \Delta e\right)$	$min\left(\sum \overline{GFD}\right)$
C_1	398.3651	4,300	387.5120
C_2	747.9849	1,092	459.0019

Table 4.8 continued

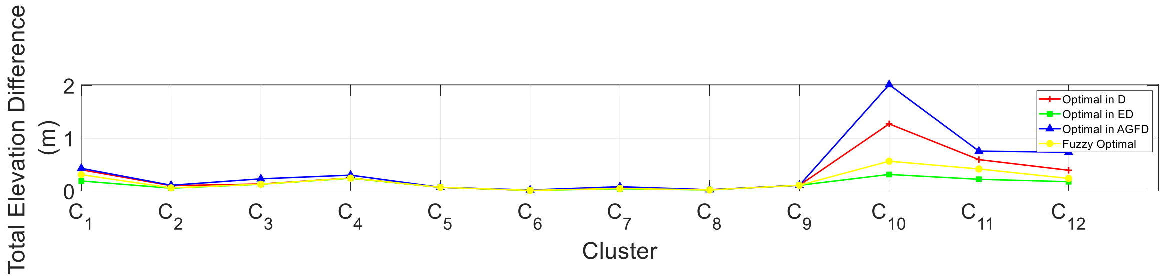
C_3	203.2288	2,312	198.2211
C_4	335.3407	3,008	159.1003
C_5	118.1112	698	74.5280
C_6	232.2735	198	162.6075
C_7	333.8459	818	257.8640
C_8	661.3168	238	262.9734
C_9	284.8119	1,088	83.6776
C_{10}	6,890.9860	20,156	1,457.6814
C_{11}	2,363.2775	2,222	811.96021
C_{12}	873.1632	1,768	681.2969

Tables 4.6 to 4.8 reveal that clusters with optimal values in d tend to exhibit lower values in $min\ (\sum d)$, while clusters optimal in parameter Δe tend to show lower values in $min\ (\sum \Delta e)$, and clusters optimal in parameter $\overline{GFD}$ demonstrate lower values in $min\ (\sum \overline{GFD})$. This reveals the limitation of the ordinary TSP-BIP, which can only address a singular objective function, implying that it optimizes only one parameter at a time. Graphical representations in Figures 4.15 (a) to (c) facilitate a clearer comparison between the fuzzy TSP-BIP algorithm and the ordinary TSP-BIP algorithm. The results produced by the fuzzy TSP-BIP algorithm for $min\ (\sum d)$, $min\ (\sum \Delta e)$ and $min\ (\sum \overline{GFD})$ are collectively termed as “fuzzy optimal”. Clusters 10 and 11 exhibit significant parameter deviations due to their substantial number of green energy locations. Optimizing a single parameter within these clusters may lead to a drastic increase in the other two parameters. In contrast, the

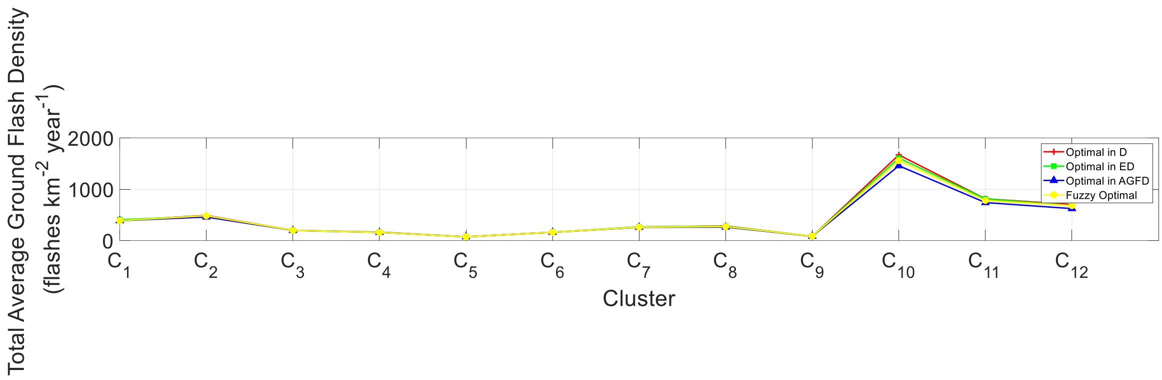
fuzzy optimal consistently maintains low values across all three parameters: distance, elevation difference, and average GFD.



(a) Total Distance



(b) Total Elevation Difference



(c) Total Average Ground Flash Density

Figure 4.15 (a) to (c): Fuzzy Optimal against Optimal in D, ED and AGFD

MCDM methods are employed to determine the best sets among fuzzy optimal, optimal in D, ED, and AGFD through ranking. Two MCDM methods named TOPSIS and COPRAS (Irik Mukhametzyanov, 2024), are utilized to rank the sets from best (1) to worst (4) across all 12 clusters. Decision matrices are derived from Table 4.5 (Fuzzy Optimal) and Tables 4.6 to 4.8 (Optimal in D, ED, and AGFD). In this comparative analysis, clusters 1 to 12 serve as alternatives, and the three criteria are $min\,(\sum d)$, $min\,(\sum \Delta e)$ and

$min\,(\sum \overline{GFD})$, each assigned equal weights of $\frac{1}{3}$ to avoid bias. All three criteria are considered cost criteria, as lower values are preferred. The computations are performed using MATLAB workspace. Figures 4.16 and 4.17 present the score values for each set in the decision matrix using TOPSIS and COPRAS, respectively, while Table 4.9 outlines the resulting ranking from these comparisons.

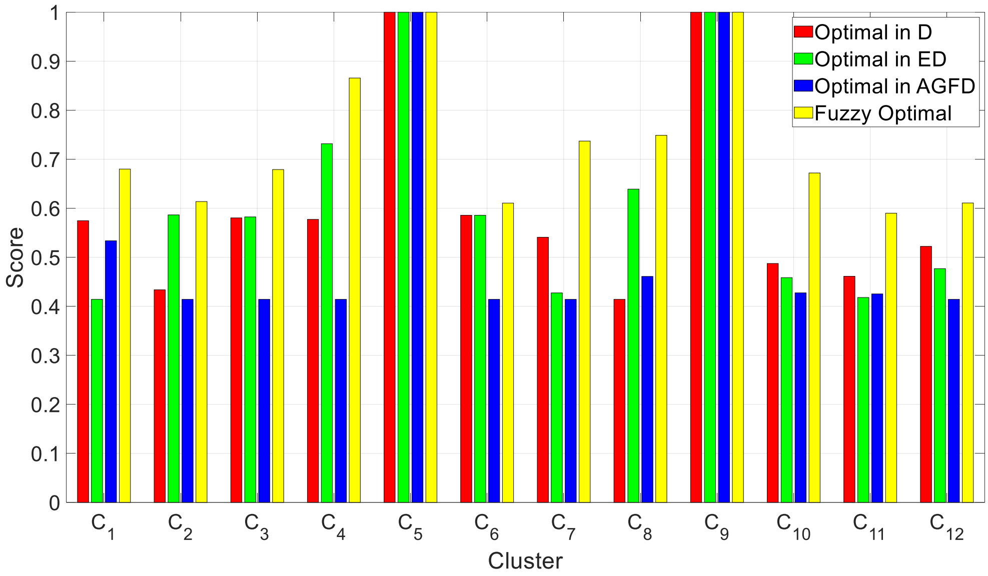


Figure 4.16: Generated Score in TOPSIS

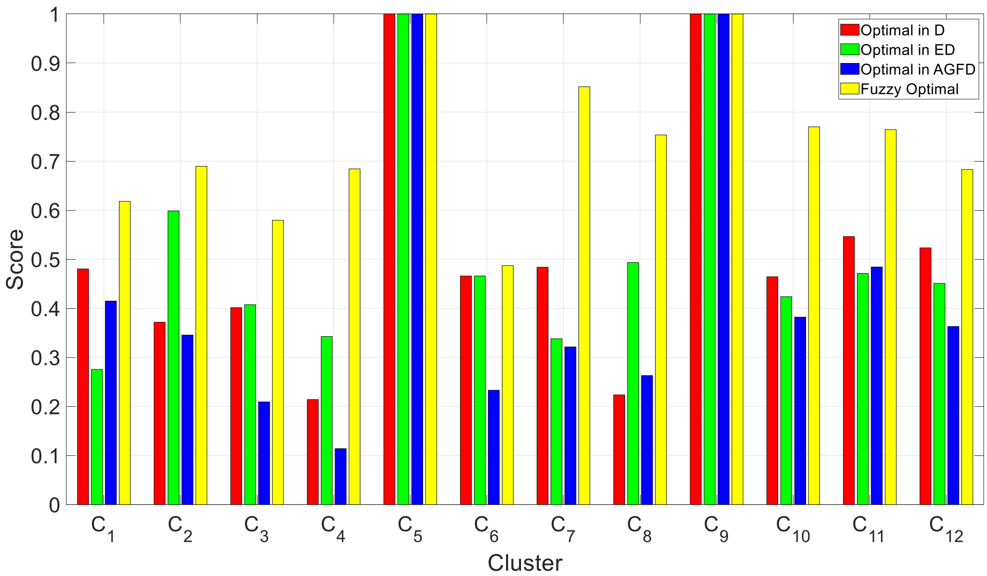


Figure 4.17: Generated Score in COPRAS

Table 4.9: Results Comparison of Fuzzy TSP-BIP against Ordinary TSP-BIP

C_i	Optimization		Rank	
			TOPSIS	**COPRAS**
C_1	TSP-BIP	Optimal in D	2	2
		Optimal in ED	4	4
		Optimal in AGFD	3	3
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_2	TSP-BIP	Optimal in D	3	3
		Optimal in ED	2	2
		Optimal in AGFD	4	4
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_3	TSP-BIP	Optimal in D	3	3
		Optimal in ED	2	2
		Optimal in AGFD	4	4
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_4	TSP-BIP	Optimal in D	3	3
		Optimal in ED	2	2
		Optimal in AGFD	4	4
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_5	TSP-BIP	Optimal in D	1	1
		Optimal in ED	1	1
		Optimal in AGFD	1	1
	Fuzzy TSP-BIP	Fuzzy optimal	1	1

Table 4.9 continued

C_6	TSP-BIP	Optimal in D	2	2
		Optimal in ED	2	2
		Optimal in AGFD	4	4
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_7	TSP-BIP	Optimal in D	2	2
		Optimal in ED	3	3
		Optimal in AGFD	4	4
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_8	TSP-BIP	Optimal in D	4	4
		Optimal in ED	2	2
		Optimal in AGFD	3	3
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_9	TSP-BIP	Optimal in D	1	1
		Optimal in ED	1	1
		Optimal in AGFD	1	1
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_{10}	TSP-BIP	Optimal in D	2	2
		Optimal in ED	3	3
		Optimal in AGFD	4	4
	Fuzzy TSP-BIP	Fuzzy optimal	1	1

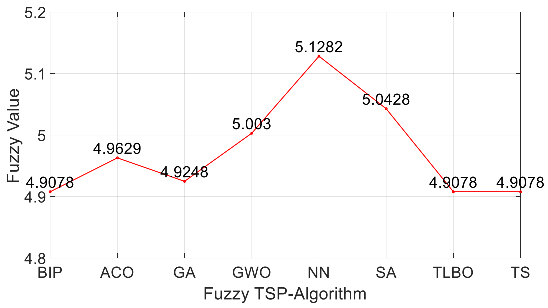
Table 4.9 continued

C_{11}	TSP-BIP	Optimal in D	2	2
		Optimal in ED	4	4
		Optimal in AGFD	3	3
	Fuzzy TSP-BIP	Fuzzy optimal	1	1
C_{12}	TSP-BIP	Optimal in D	2	2
		Optimal in ED	3	3
		Optimal in AGFD	4	4
	Fuzzy TSP-BIP	Fuzzy optimal	1	1

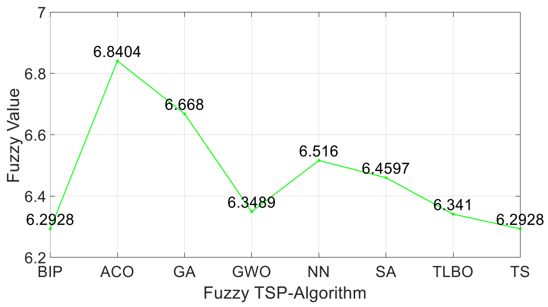
Notably, the fuzzy TSP-BIP consistently outperforms the ordinary TSP-BIP (optimal in D, ED, and AGFD) across all clusters, as evidenced by the higher scores or rankings depicted in Figures 4.16 to 4.17 and Table 4.9. Employing both TOPSIS and COPRAS for results validation consistently places the cluster utilizing the fuzzy TSP-BIP algorithm at the top of the score or ranking. This indicates that integrating fuzzy logic operations into the TSP-BIP solver enhances its effectiveness in improving result performance. By efficiently integrating GEs, fuzzy TSP-BIP enhances the TSP-BIP algorithm's capability to optimize electrical power lines design routing while considering multi-objective functions of $min\ (\sum d)$, $min\ (\sum \Delta e)$ and $min\ (\sum \overline{GFD})$. An intriguing observation from Table 4.9 is that clusters C_5 and C_9 share the same rank for each optimization. This is primarily due to the low number of GEs in these clusters (4 for each), resulting in a smaller number of possible routes. Consequently, the optimization of singular objective functions increases the likelihood of reaching the global optimum, strengthening the reliability of the results. Both TOPSIS and COPRAS produce identical rankings for each cluster due to this simplified optimization landscape. Thus, the superiority of the fuzzy TSP-BIP algorithm over the ordinary TSP-BIP algorithm has been demonstrated.

4.3.3 Validation Results of Proposed Method against Fuzzy TSP Algorithms

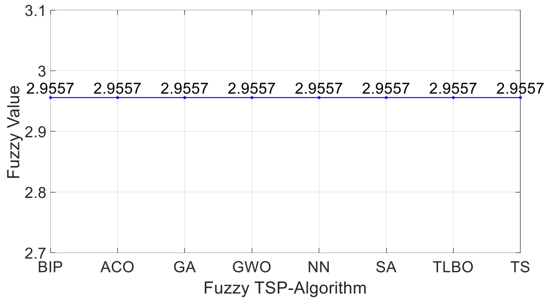
The fuzzy matrix, f_m is generated using the method in Figure C-4 (Appendix). This fuzzy matrix serves as input data for each fuzzy TSP algorithm, aiming to achieve minimal fuzzy values, indicating the lowest $min\,(\sum d)$, $min\,(\sum \Delta e)$ and $min\,(\sum \overline{GFD})$ for each cluster. Therefore, the optimal fuzzy TSP-BIP algorithm is expected to produce the lowest fuzzy values. The obtained fuzzy values for all 12 clusters are presented in Figure 4.18. Figures 4.18 (a) to (l) reveal a consistent trend, wherein the fuzzy TSP-BIP algorithms consistently outperform other TSP-BIP algorithms in generating the lowest fuzzy values. The clusters such as C_3, C_5, C_6, C_9, where all algorithms manage to generate the lowest fuzzy values. This occurrence can be attributed to the rapid convergence of TSP algorithms when dealing with a small number of green energy locations, thereby avoiding being trapped in local optima. However, in cases involving higher counts of green energy locations, particularly in C_{10} with 92 green energy locations, only the fuzzy TSP-BIP algorithm achieves the minimum fuzzy value of 21.8918. These findings consistently highlight the superiority of the fuzzy TSP-BIP algorithm in producing the lowest fuzzy values across all 12 clusters.



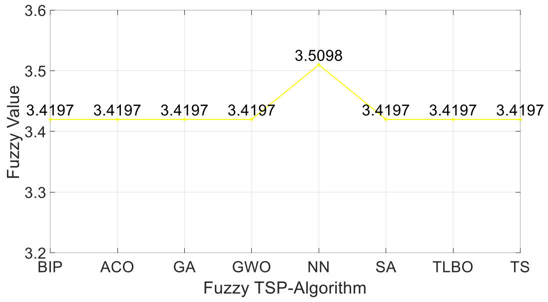
(a) C_1



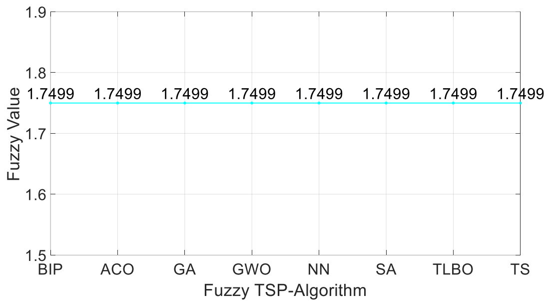
(b) C_2



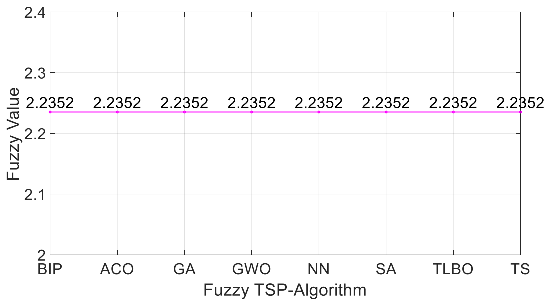
(c) C_3



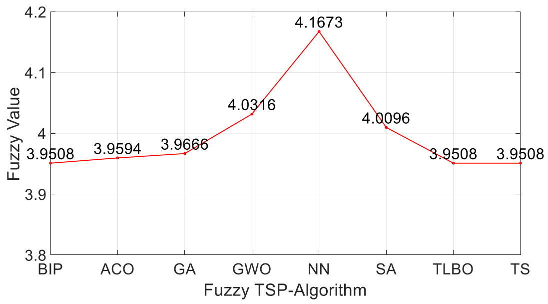
(d) C_4



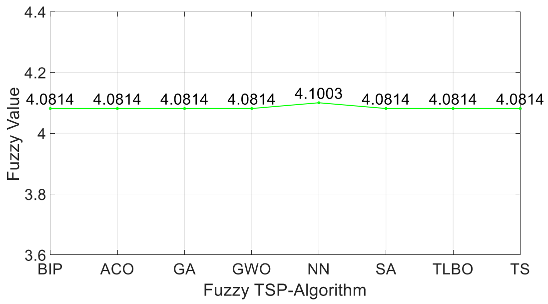
(e) C_5



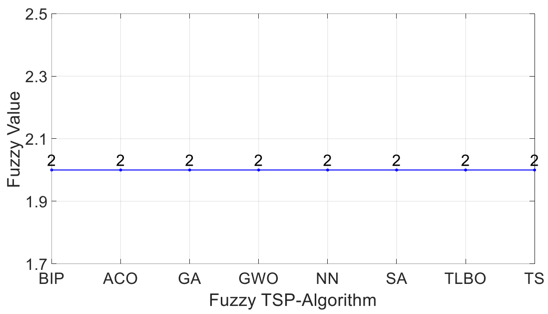
(f) C_6



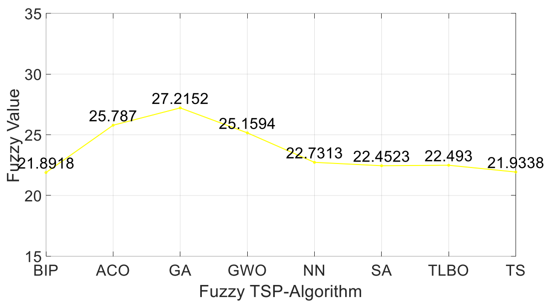
(g) C_7



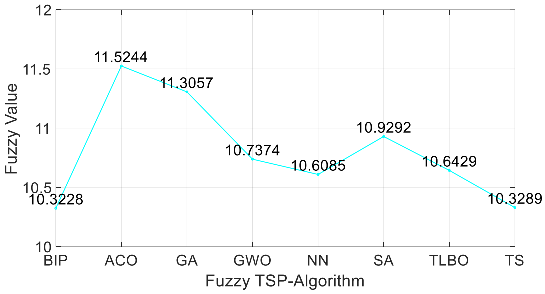
(h) C_8



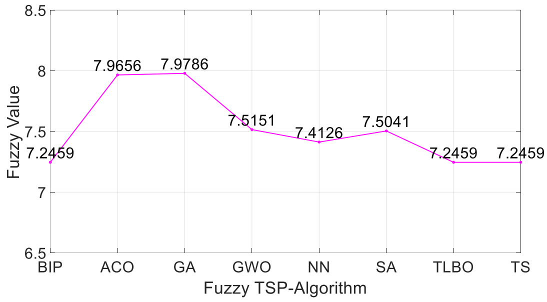
(i) C_9



(j) C_{10}



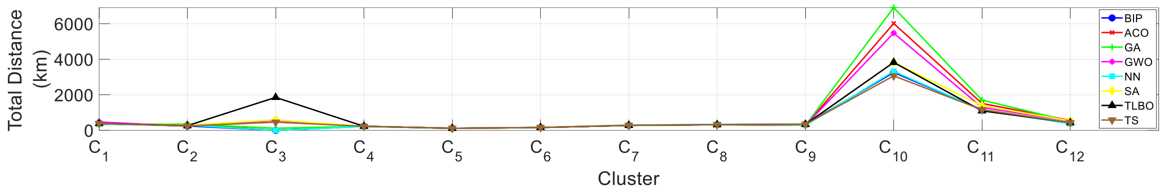
(k) C_{11}



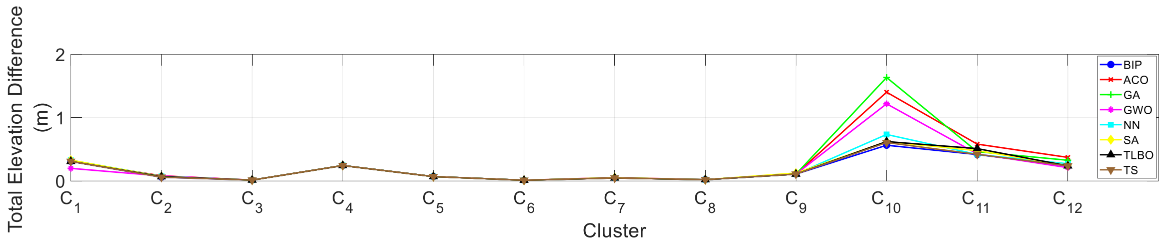
(l) C_{12}

Figure 4.18 (a) to (l): Fuzzy Values in Fuzzy TSP Algorithms

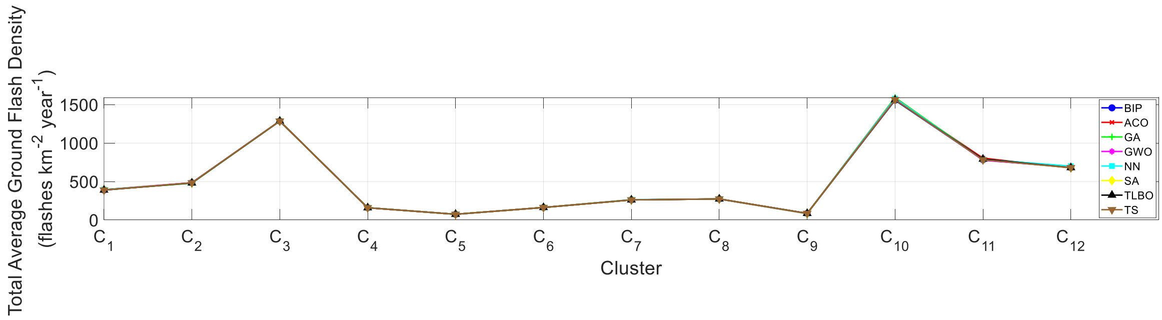
Figures 4.19 (a) to (c) depict the $min\ (\sum d)$, $min\ (\sum \Delta e)$ and $min\ (\sum \overline{GFD})$ for each cluster across all fuzzy TSP algorithms. The fuzzy TSP-GA algorithm yields subpar results, particularly noticeable in cluster 10. Despite employing a population size of 5,000 and running for 5,000 generations, the algorithm experiences significant convergence slowdown due to the higher number of green energy locations (92 locations). Although GA is recognized for its slower convergence, it is valued for its infrequent entrapment in local optima. Moreover, both ACO and GWO exhibit elevated values for total distance and total elevation difference. While ACO excels in addressing smaller-scale TSP problems, its performance sharply declines when confronted with the integration of large-scale green energy locations. On the other hand, GWO, known for its ease of implementation, suffers from slow convergence, reduced accuracy, and a heightened risk of falling into local optima. In contrast, the fuzzy TSP-BIP consistently demonstrates promising results in the design of electrical power lines routing for integrating green energy locations.



(a) $min\ (\sum d)$ of Fuzzy TSP Algorithms



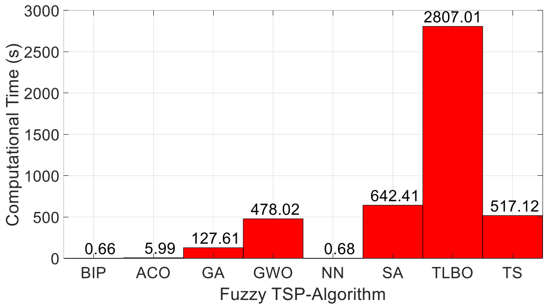
(b) $min\ (\sum \Delta e)$ of Fuzzy TSP Algorithms



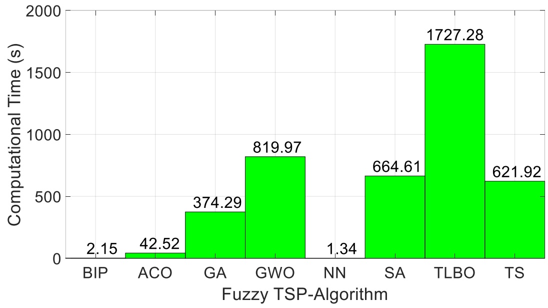
(c) $min\ (\sum \overline{GFD})$ of Fuzzy TSP Algorithms

Figure 4.19 (a) to (c): Parameter Values in Fuzzy TSP Algorithms

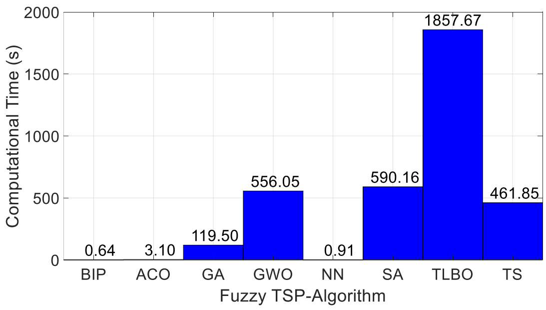
Figures 4.20 (a) to (l) present the computational time for each fuzzy TSP algorithm across all clusters. The computational time is measured using the $tic - toc$ function in MATLAB. The analyses provide comprehensive insights into the computational times for eight fuzzy TSP optimization algorithms applied to the 12 clusters. Remarkably, fuzzy TSP-BIP and fuzzy TSP-NN emerge as top performers, consistently demonstrating impressively low computational times across diverse clusters. In contrast, fuzzy TSP-ACO exhibits moderate to high computational times, reaching 42.52 s for cluster C_2, suggesting a potential trade-off between computational efficiency and solution quality. Fuzzy TSP-GA and fuzzy TSP-GWO show higher computational times, such as 127.61 s and 478.02 s for cluster C_1, indicating a need for further exploration into the tuning of their parameters and adaptability to specific problem domains. Fuzzy TSP-NN and fuzzy TSP-SA display mixed performance, with computational times varying across clusters. Fuzzy TSO-TLBO and fuzzy TSP-TS reveal relatively high computational times, suggesting challenges in achieving convergence. Notably, fuzzy TSP-TLBO and fuzzy TS, particularly in cluster C_{10}, exhibit significantly higher computational times of 3,903.55 s and 1,863.79 s, emphasizing the sensitivity of these algorithms to certain problem instances. This meticulous examination, supported by computational times, provides detailed insights into the performance of fuzzy TSP algorithms. These findings offer valuable guidance for researchers and practitioners aiming to select the most suitable fuzzy optimization approach based on the intricacies of specific problem scenarios. Overall, fuzzy TSP-BIP proves to be the most prominent algorithm for integrating green energy locations in Sarawak, focusing on optimizing electrical power lines routing design. It offers the lowest fuzzy values, high efficiency, and remarkably low computational times compared to other fuzzy TSP algorithms.



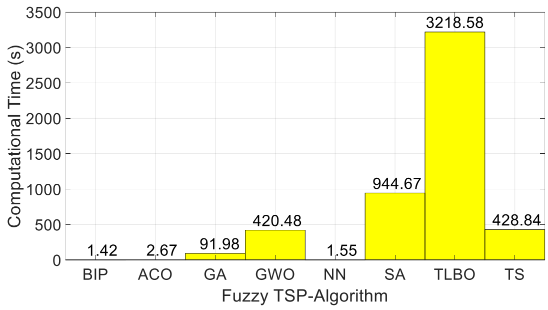
(a) C_1



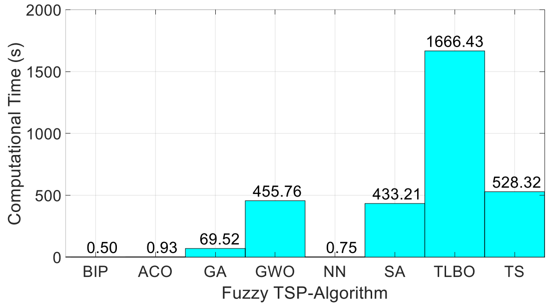
(b) C_2



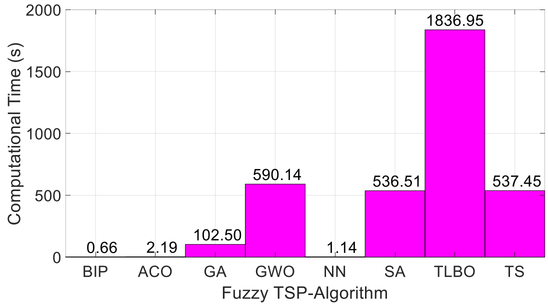
(c) C_3



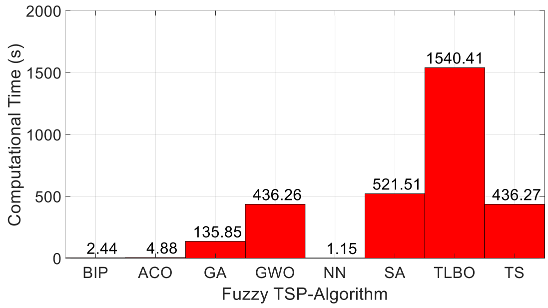
(d) C_4



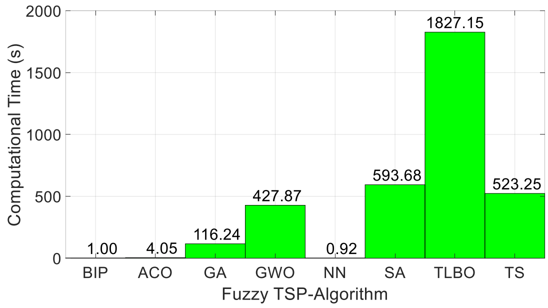
(e) C_5



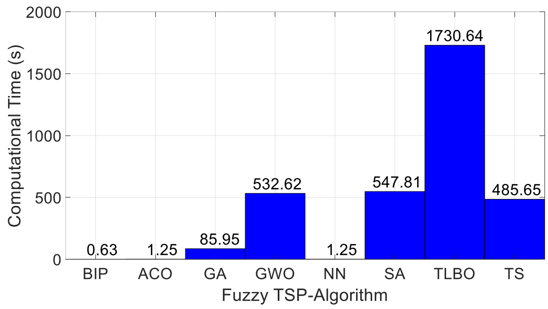
(f) C_6



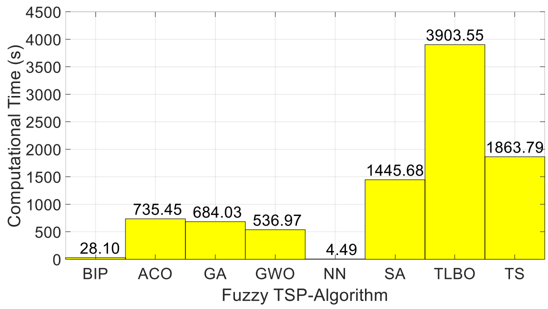
(g) C_7



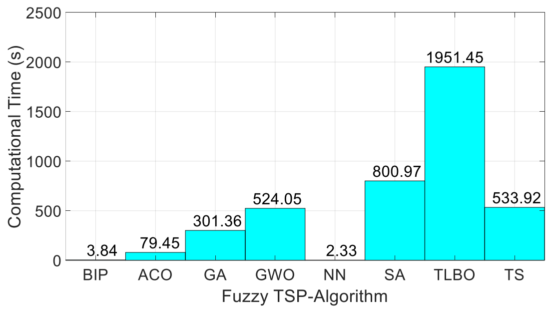
(h) C_8



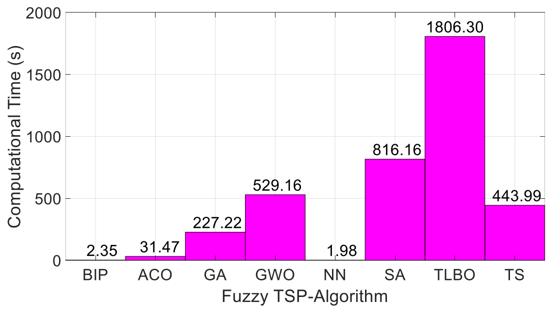
(i) C_9



(j) C_{10}



(k) C_{11}



(l) C_{12}

Figure 4.20 (a) to (l): Computational Time of Fuzzy TSP Algorithms

4.4 Results of Real-Time Monitoring, Control and Automation for IGESs

This section is structured into four distinct subsections. Firstly, it demonstrates the real-time dynamic data transfer from the historical cloud to the real-time cloud via the MATLAB workspace. Following this, critical parameter results concerning green energy generation, energy storage, AC, and DC loads are presented. These findings are showcased through interfaces such as MATLAB Simulink, server systems, and SCADA monitoring dashboards. In the third subsection, outputs from both manual control and automation strategies are highlighted. These strategies have been implemented and executed across various platforms including MATLAB, server systems, and SCADA. Lastly, the fourth section unveils outcomes resulting from the integration of SCADA with experimental prototyping for validation purposes.

4.4.1 Real-Time Dynamic Data Generation

Figure 4.21 illustrates the successful retrieval of load demand, solar radiation, and temperature data from a historical cloud based on the current time. These historical data are then transmitted to a real-time cloud for integration into the model. Within the MATLAB workspace, individual fields (Field 1, Field 2, and Field 3) receive data (load demand, solar radiation, and temperature) from the historical cloud. The data retrieval process operates based on the closest past timestamp within a 10-minute interval. Subsequently, the data received in the MATLAB workspace are transmitted to the real-time cloud. During the streaming of load demand data, it undergoes adjustment using Equation 3.49 within the MATLAB workspace before being sent to the real-time cloud. This adjustment is necessary because the loads are constrained between 1 MW and 2 MW, as outlined in the methodology. The proposed method for the real-time cloud to retrieve dynamic data from the historical cloud based on the current time is valuable as it enables real-time dynamic data to be fed into the testing model or system for performance evaluations and validation.

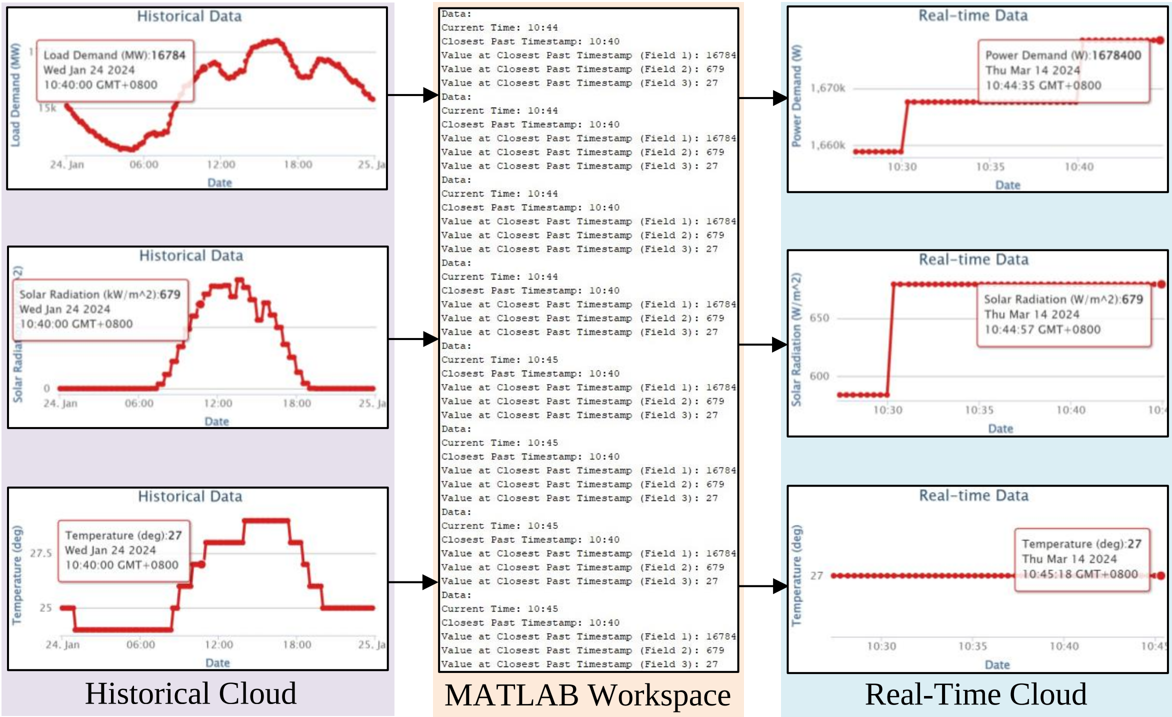
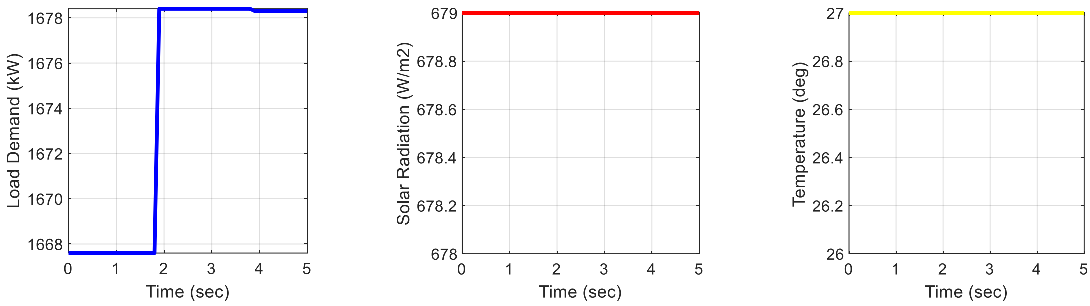


Figure 4.21: Results of Real-Time Dynamic Data

4.4.2 Real-Time Dynamic Monitoring Results

The power parameters data from the IGESs are streamed to SCADA for monitoring purposes. This subsection focuses on validating the successful streaming results from MATLAB Simulink to real-time SCADA via the server. The monitoring data across MATLAB Simulink, the server, and the SCADA dashboard are compared and validated. It is imperative to note that the actual simulation runtime extends to approximately 20 minutes, despite the simulation time being set at 5 s. This prolonged duration is a result of developing a heavily integrated green energy system model. The results data for load demand, solar radiation, and temperature from the IGESs in MATLAB Simulink are illustrated in Figures 4.22 (a) to (c) prior to their transmission to SCADA through a server. While the load demand data are updated every 10 minutes, noticeable variations over time are observed. In contrast, solar radiation and temperature data remain constant throughout the simulation, as they are updated every 30 minutes.



(a) Load Demand (b) Solar Radiation (c) Temperature

Figure 4.22 (a) to (c): Monitoring Data on Load Demand, Solar Radiation and Temperature

Moreover, the monitoring data for power parameters, including voltage, current, and active power, pertaining to solar generation, energy storage, DC load 1, and DC load 2, are depicted in Figures 4.23 (a) to (l). All measurements commence from zero, aligning with the initiation of the simulation. Upon system activation, the load demands of both DC loads (approximately 1.7 MW each) prompt an inrush current occurrence shortly after 0 s. This inrush current can peak twice or even ten times the normal rated current. The abrupt switching of heavy loads induces voltage fluctuations due to the substantial current drawn into the system. To mitigate this, the inrush current can be curtailed by implementing components such as Negative Temperature Coefficient (NTC) thermistors, which reduce resistance as temperature rises. Typically, the very high inrush current is cleared approximately after 0.1 s. Following this, the readings for these power parameters gradually stabilize over 1 s. Concurrently, these data are streamed to the SCADA via the server. The voltage for solar generation is set at 5 kV, while the energy storage voltage remains at 400 V. Additionally, the voltages for DC load 1 and DC load 2 are maintained at 15 kV.

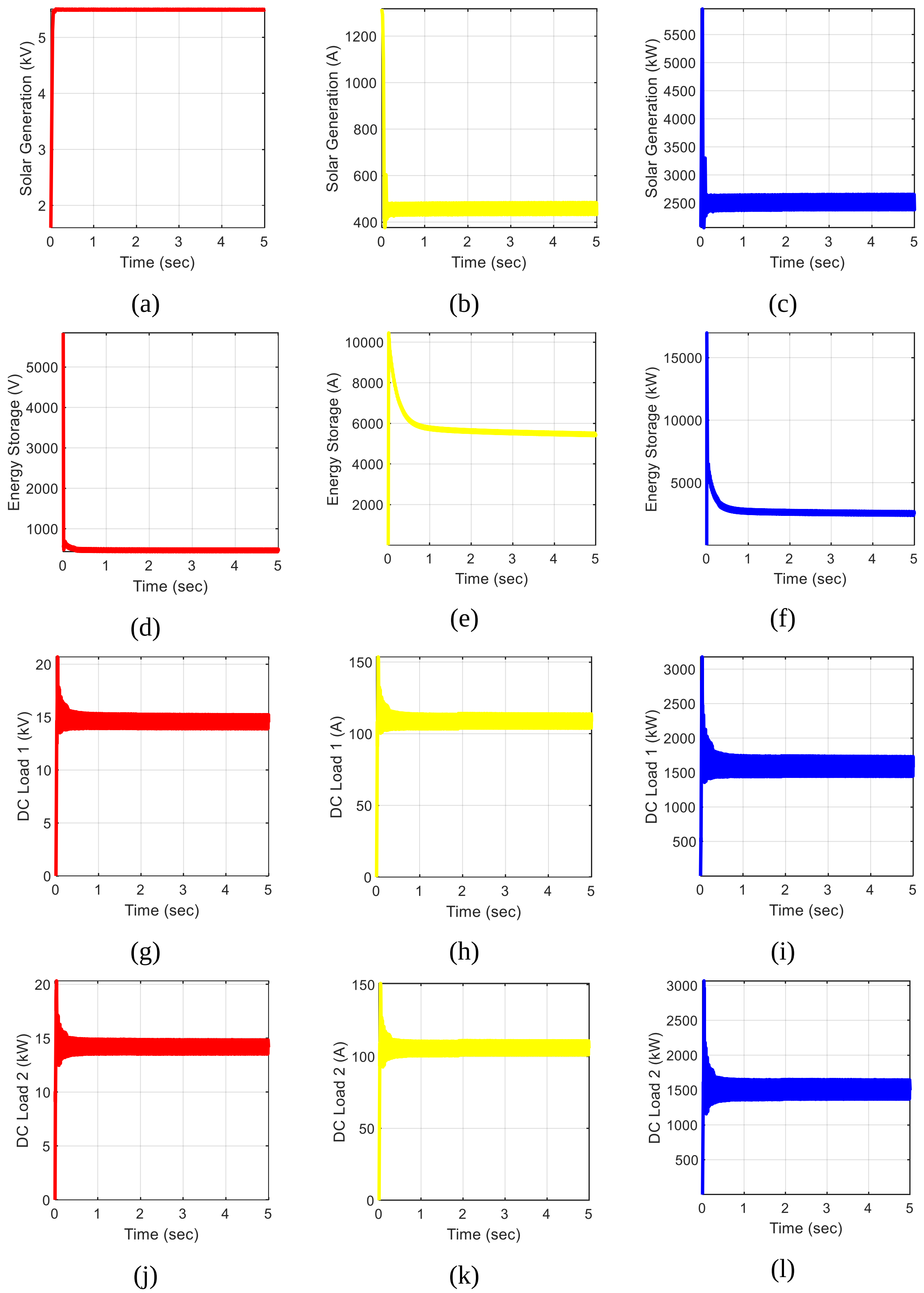
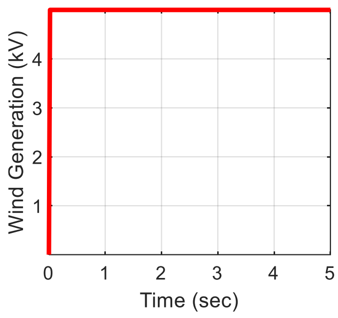
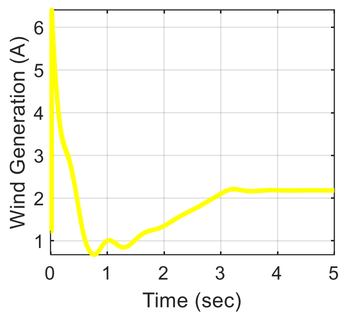


Figure 4.23 (a) to (l): Monitoring Data on Solar Generation, Energy Storage, DC Load 1 and DC Load 2

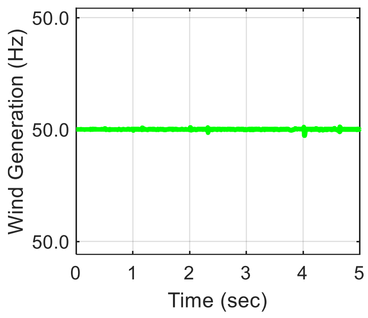
Figures 24 (a) to (x) illustrate the monitoring data of wind generation, hydro generation, AC load 1, and AC load 2, considering six monitoring parameters (voltage, current, frequency, active power, reactive power, and power factor) from AC generations and AC loads. In the case of AC generators, which function as inductive motors converting rotational energy to electrical energy, a load with an inductive power of 250 $kVar$ is connected to each generator. These generators draw reactive power from the inductive load to magnetize the winding for excitation. Once the rotor spins with sufficient magnetic field strength, high rotational speed, and magnetism, it generates electrical power for the AC loads. Therefore, the negative values of the reactive powers shown in Figures 24 (e) and (k) indicate that the generators draw reactive power from the loads. The generators continuously supply the reactive power to maximize the magnetism to their respective windings. Furthermore, wind generator, hydro generator, AC load 1, and AC load 2 experience the same issue of inrush current at the initial period. The hydro generator stabilizes after 1 s as it deals with two heavy AC loads (approximately 1.7 MW each). The current magnitude of the wind generator exhibits an interesting pattern. Please note that the base wind speed set in the model is 9 ms^{-1}, and the nominal power is 10 kW. A skewed limiter block is added to the wind generator sub-model at 0 s; the wind speed is 7 ms^{-1}, and at 2 s, the wind speed slowly starts to increase from 7 $ms^{-1}$to 9 ms^{-1}. Therefore, after 1 s, the current of the wind generator gradually increases and saturates when it reaches approximately 10 kW. The purpose of modeling the wind generator increasing power at a slow rate is to avoid rapid acceleration from low to high speed, which may damage the structure, especially the blade, due to sudden high stress. Hence, wind modeling depicts a more realistic behavior of wind turbine operation. Additionally, the frequency indicated on the y-axis labeling of the wind generation shows multiple occurrences of 50 Hz in Figures 24 (c) and (i), indicating that the frequency fluctuation is extremely low. Both AC loads are in steady state after approximately 0.1 s.



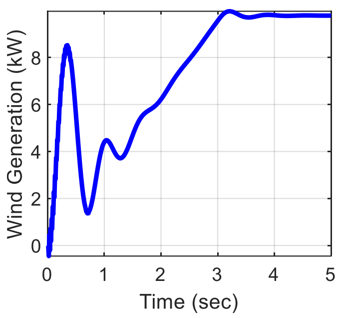
(a)



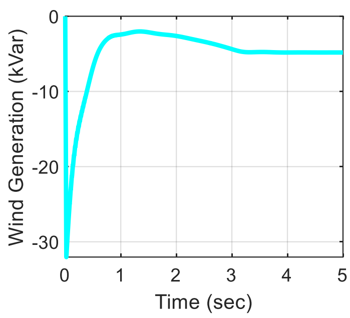
(b)



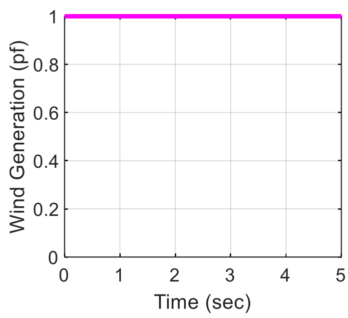
(c)



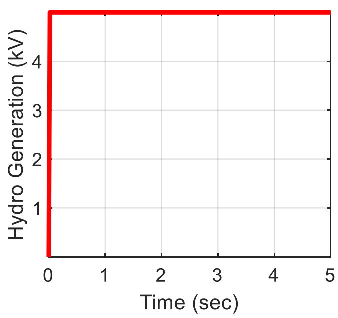
(d)



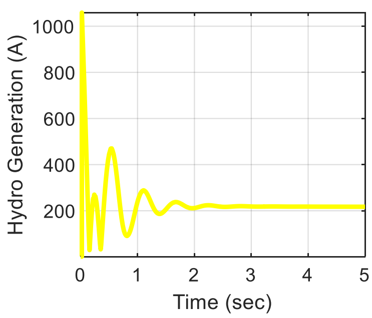
(e)



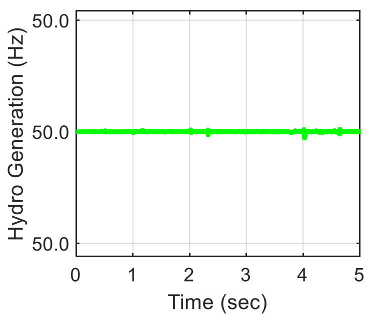
(f)



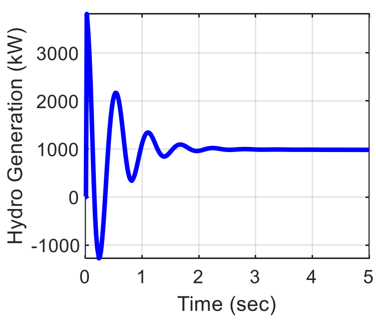
(g)



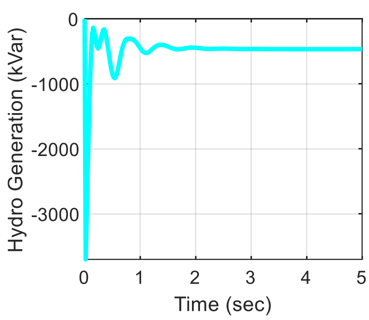
(h)



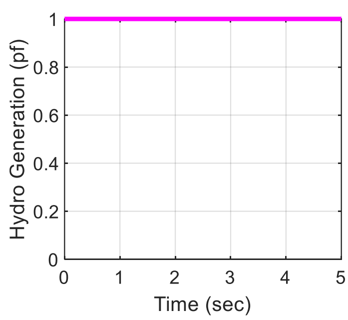
(i)



(j)



(k)



(l)

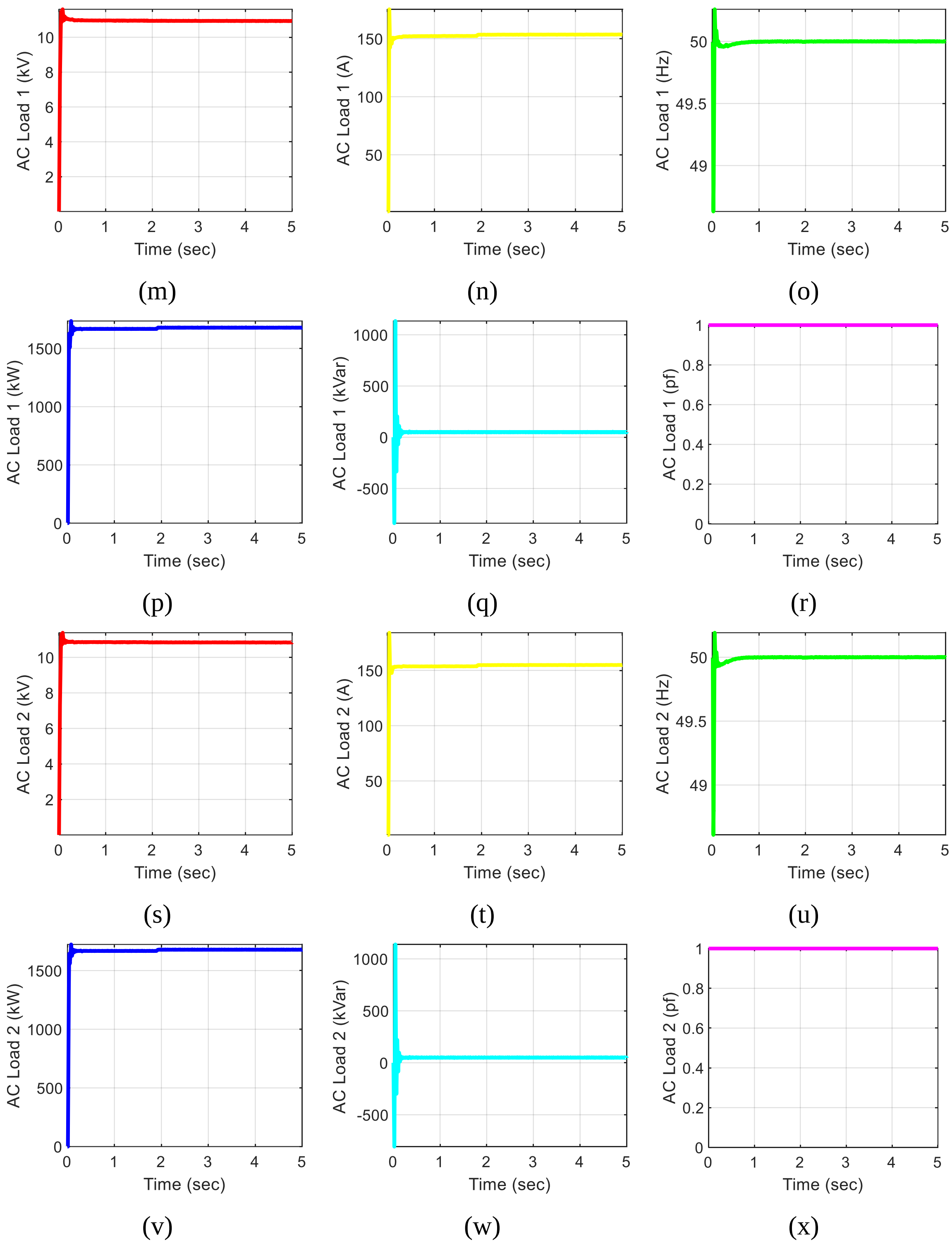


Figure 4.24 (a) to (x): Monitoring Data on Wind Generation, Hydro Generation, AC Load 1 and AC Load 2

The OPC quick client server displays the monitoring data for green energy generation, energy storage, AC loads, and DC loads, as showcased in Figures 4.25 (a) and (b). Each tag corresponds to a specific parameter, as outlined in Figure 3.27. The data type employed is double, representing the actual monitoring values. Timestamps indicate when data is streamed through the server, while the “good” quality signifies that the data retrieval process was successful. The update count denotes the number of data changes during the streaming process, affirming the successful transmission of readings from MATLAB Simulink to the server.

Item ID	Data Type	Value	Timestamp	Quality	Update Count
C1.D1.T1	Double	1678.4	10:40:44.198	Good	3
C1.D1.T2	Double	679	10:31:03.265	Good	2
C1.D1.T3	Double	27	10:16:30.128	Good	1
C1.D1.T4	Double	5.50173	10:43:54.574	Good	869
C1.D1.T5	Double	433.188	10:43:54.574	Good	864
C1.D1.T6	Double	2383.28	10:43:54.590	Good	866
C1.D1.T7	Double	446.881	10:43:54.590	Good	863
C1.D1.T8	Double	5618.51	10:43:54.590	Good	862
C1.D1.T9	Double	2510.8	10:43:54.590	Good	859
C1.D1.T10	Double	13.973	10:43:54.574	Good	879
C1.D1.T11	Double	104.371	10:43:54.574	Good	876
C1.D1.T12	Double	1458.38	10:43:54.574	Good	876
C1.D1.T13	Double	13.6188	10:43:54.574	Good	876
C1.D1.T14	Double	101.726	10:43:54.574	Good	876
C1.D1.T15	Double	1385.38	10:43:54.574	Good	875
C1.D1.T16	Double	5	10:43:54.574	Good	835
C1.D1.T17	Double	50	10:43:53.627	Good	347
C1.D1.T18	Double	-3.97874E-013	10:43:54.574	Good	870
C1.D1.T19	Double	1.74471	10:43:54.574	Good	870
C1.D1.T20	Double	50.0053	10:43:54.574	Good	869
C1.D1.T21	Double	23.2001	10:43:54.574	Good	869
C1.D1.T22	Double	8.01811	10:43:54.574	Good	869
C1.D1.T23	Double	-3.43658	10:43:54.574	Good	869
C1.D1.T24	Double	1	10:16:30.112	Good	1
C1.D1.T25	Double	5	10:43:53.643	Good	831
C1.D1.T26	Double	50	10:43:53.643	Good	351
C1.D1.T27	Double	-3.97874E-013	10:43:54.574	Good	868
C1.D1.T28	Double	216.797	10:43:54.574	Good	868
C1.D1.T29	Double	50.0045	10:43:54.574	Good	867
C1.D1.T30	Double	24.8981	10:43:54.574	Good	866
C1.D1.T31	Double	983.239	10:43:54.574	Good	866
C1.D1.T32	Double	-456.366	10:43:54.574	Good	865
C1.D1.T33	Double	1	10:16:30.128	Good	1
C1.D1.T34	Double	10.9426	10:43:54.574	Good	866
C1.D1.T35	Double	49.9997	10:43:54.574	Good	866
C1.D1.T36	Double	-12.2951	10:43:54.574	Good	866
C1.D1.T37	Double	153.455	10:43:54.574	Good	866
C1.D1.T38	Double	49.9992	10:43:54.574	Good	866
C1.D1.T39	Double	-14.0188	10:43:54.574	Good	866
C1.D1.T40	Double	1678.44	10:43:54.574	Good	865
C1.D1.T41	Double	50.5089	10:43:54.574	Good	864
C1.D1.T42	Double	1	10:16:30.143	Good	1
C1.D1.T43	Double	10.8423	10:43:54.574	Good	861
C1.D1.T44	Double	49.9996	10:43:54.574	Good	861
C1.D1.T45	Double	-14.3122	10:43:54.574	Good	861
C1.D1.T46	Double	154.841	10:43:54.574	Good	861
C1.D1.T47	Double	49.9997	10:43:54.574	Good	860
C1.D1.T48	Double	-16.0155	10:43:54.574	Good	860
C1.D1.T49	Double	1678.09	10:43:54.574	Good	860
C1.D1.T50	Double	49.903	10:43:54.574	Good	860
C1.D1.T51	Double	1	10:16:30.159	Good	1
C1.D1.T52	Double	0.00499651	10:43:54.574	Good	861
C1.D1.T53	Double	0.0108439	10:43:54.574	Good	861
C1.D1.T54	Double	104.371	10:43:54.574	Good	865
C1.D1.T55	Double	101.726	10:43:54.574	Good	865

(a) Monitoring Data at 2.5 *s*

Item ID	Data Type	Value	Timestamp	Quality	Update Count
C1.D1.T1	Double	1678.3	10:50:47.505	Good	6
C1.D1.T2	Double	679	10:31:03.265	Good	3
C1.D1.T3	Double	27	10:00:37.001	Good	2
C1.D1.T4	Double	5.50141	10:56:10.541	Good	2918
C1.D1.T5	Double	433.954	10:56:10.541	Good	2903
C1.D1.T6	Double	2387.36	10:56:10.557	Good	2901
C1.D1.T7	Double	485.67	10:56:10.557	Good	2895
C1.D1.T8	Double	5416.03	10:56:10.557	Good	2893
C1.D1.T9	Double	2630.4	10:56:10.557	Good	2879
C1.D1.T10	Double	14.5223	10:56:10.541	Good	2942
C1.D1.T11	Double	108.468	10:56:10.541	Good	2939
C1.D1.T12	Double	1575.21	10:56:10.541	Good	2938
C1.D1.T13	Double	14.1605	10:56:10.541	Good	2938
C1.D1.T14	Double	105.766	10:56:10.541	Good	2938
C1.D1.T15	Double	1497.7	10:56:10.541	Good	2936
C1.D1.T16	Double	5	10:56:10.541	Good	2773
C1.D1.T17	Double	50	10:56:10.541	Good	1534
C1.D1.T18	Double	3.88983E-014	10:56:10.541	Good	2924
C1.D1.T19	Double	2.18112	10:56:10.541	Good	2923
C1.D1.T20	Double	50	10:56:10.541	Good	2920
C1.D1.T21	Double	26.222	10:56:10.541	Good	2920
C1.D1.T22	Double	9.78328	10:56:10.541	Good	2920
C1.D1.T23	Double	-4.81864	10:56:10.541	Good	2921
C1.D1.T24	Double	1	09:49:41.678	Good	1
C1.D1.T25	Double	5	10:56:10.541	Good	2766
C1.D1.T26	Double	50	10:56:10.541	Good	1530
C1.D1.T27	Double	3.88983E-014	10:56:10.541	Good	2919
C1.D1.T28	Double	217.342	10:56:10.541	Good	2918
C1.D1.T29	Double	50	10:56:10.541	Good	2917
C1.D1.T30	Double	25.2724	10:56:10.541	Good	2912
C1.D1.T31	Double	982.699	10:56:10.541	Good	2911
C1.D1.T32	Double	-463.94	10:56:10.541	Good	2912
C1.D1.T33	Double	1	09:49:41.694	Good	1
C1.D1.T34	Double	10.9379	10:56:10.541	Good	2914
C1.D1.T35	Double	49.9979	10:56:10.541	Good	2916
C1.D1.T36	Double	-12.5758	10:56:10.541	Good	2916
C1.D1.T37	Double	153.536	10:56:10.541	Good	2915
C1.D1.T38	Double	49.9985	10:56:10.541	Good	2913
C1.D1.T39	Double	-14.2587	10:56:10.541	Good	2913
C1.D1.T40	Double	1678.64	10:56:10.541	Good	2907
C1.D1.T41	Double	49.3212	10:56:10.541	Good	2907
C1.D1.T42	Double	1	09:49:41.726	Good	1
C1.D1.T43	Double	10.8395	10:56:10.541	Good	2904
C1.D1.T44	Double	49.9966	10:56:10.541	Good	2904
C1.D1.T45	Double	-14.5228	10:56:10.541	Good	2904
C1.D1.T46	Double	154.885	10:56:10.541	Good	2900
C1.D1.T47	Double	49.9986	10:56:10.541	Good	2899
C1.D1.T48	Double	-16.1526	10:56:10.541	Good	2898
C1.D1.T49	Double	1678.19	10:56:10.541	Good	2898
C1.D1.T50	Double	47.7472	10:56:10.541	Good	2897
C1.D1.T51	Double	1	09:49:41.742	Good	1
C1.D1.T52	Double	0.00499644	10:56:10.541	Good	2895
C1.D1.T53	Double	0.010841	10:56:10.541	Good	2893
C1.D1.T54	Double	108.468	10:56:10.541	Good	2900
C1.D1.T55	Double	105.766	10:56:10.541	Good	2900

(b) Monitoring Data at 5.0 *s*

Figure 4.25 (a) to (b): Monitoring Data in Server

The SCADA energy management system dashboard effectively presents all the necessary real-time dynamic readings from the integrated green energy systems model. These readings have been successfully streamed from MATLAB Simulink to the SCADA dashboard through the server. The behavior of the power parameters has been thoroughly discussed in the monitoring results section of MATLAB Simulink. Upon observation, it is evident that all the manual control switches are green, indicating that all green energy generations, energy storage, AC loads, and DC loads are connected. Since no faults are applied to the systems, the automation is not triggered; hence, all light indicators are red for the automation part. A slight delay was observed, especially when data was streamed from MATLAB to the server. This delay primarily arises from the numerous communication channels built in MATLAB Simulink. However, the data streamed from the server to the SCADA experiences negligible delays. Figure 4.26 displays the monitoring data in the SCADA dashboard at 2.5 s, while Figure 4.27 illustrates the monitoring data in the SCADA dashboard at 5.0 s.

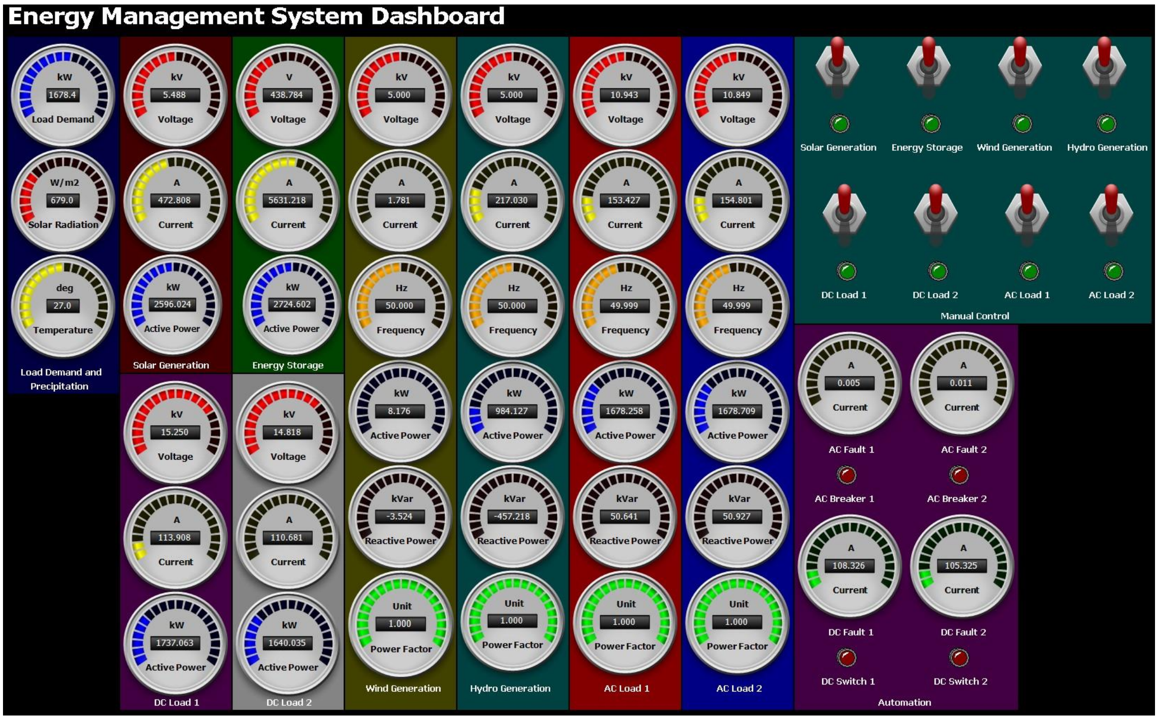


Figure 4.26: Monitoring Data at 2.5 s in SCADA

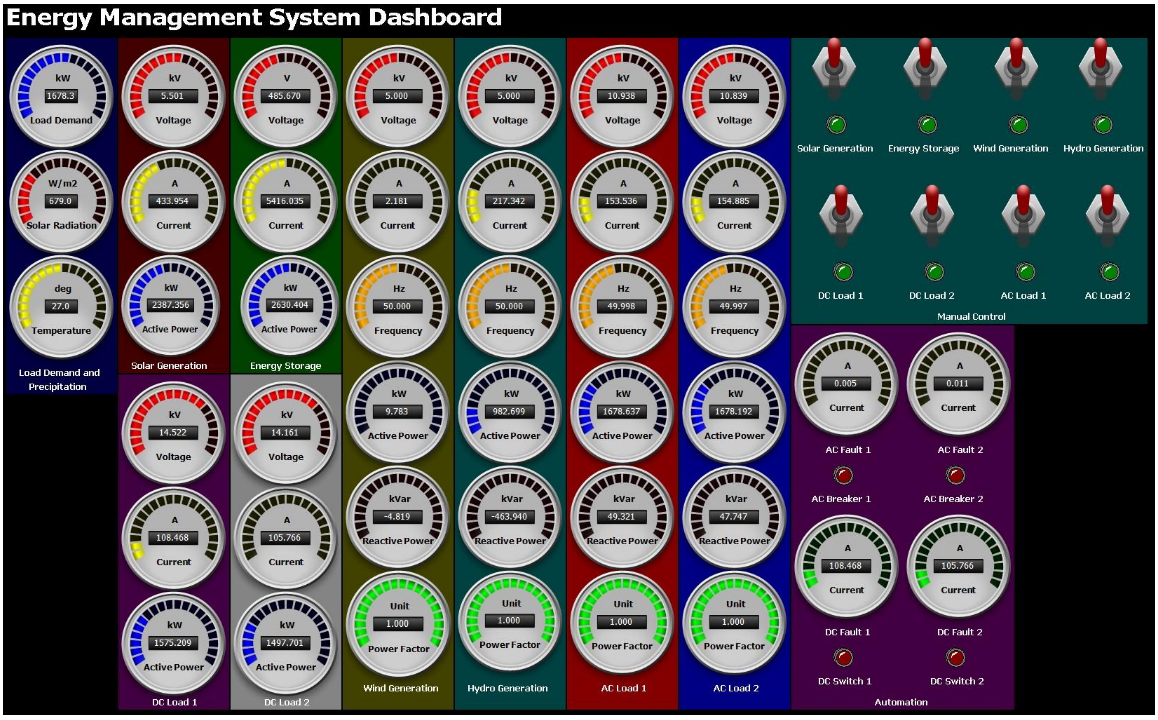


Figure 4.27: Monitoring Data at 5.0 s in SCADA

4.4.3 Real-Time Dynamic Control and Automation Results

The findings in this section affirm the effective implementation of manual control and automation for IGESs using the IIoT communication framework between MATLAB Simulink, the server, and SCADA. In practical industrial scenarios, such manual control facilitated by an IIoT-based SCADA platform can play a crucial role in tasks such as power lines maintenance. For the simulation, AC load 1 and DC load 1 are specifically considered, with manual control executed through SCADA, as evidenced in Figures 4.28 and 4.29, respectively. Figure 4.28 illustrates the initial state of the simulation, with AC load 1 connected to the IGESs model. At simulation time, t_s = 1 s, the first switching of AC load 1 occurs, signified by the change in the signal tag, TG logical state, from 1 to 0, effectively isolating AC load 1 from the model. Subsequently, a second switching operation is conducted to reconnect AC load 1 to the systems model. Observing the graph of AC load 1, a steady-state current is evident from 0 s to 1 s. Following this, the current drops to zero and remains for 1 s as the circuit breaker is opened. Reconnection of AC load 1 to the model is initiated at t_s = 2 s. However, transient current fluctuations occur due to the inductive motor, characterized by a temporary spike in current drawn to magnetize its winding for load operation and model synchronization. The current stabilizes back to normal around t_s = 3 s. Overall, these scenarios indicate successful manual control of AC load 1 from SCADA to the MATLAB Simulink model via the server.

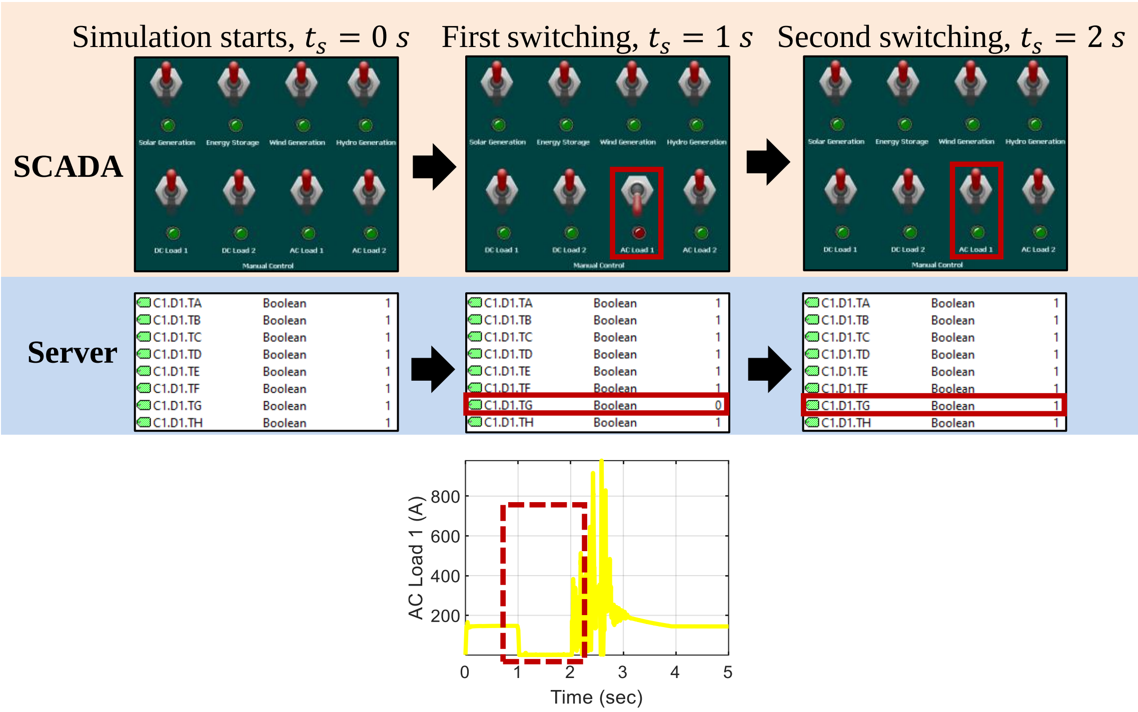


Figure 4.28: Manual Control of AC Load 1

Figure 4.29 depicts the manual control operation for DC load 1, showcasing a similar procedure to that of AC load 1. At t_s = 1 s, the first switching occurs, marked by the transition of the signal tag, TE, from 1 to 0, indicating the isolation of DC load 1 from the model. The second switching takes place at t_s = 2 s. Examining the graph representing DC load 1, at t_s = 1 s, the current is recorded as zero as the circuit breaker is opened. Following the manual closure of the circuit breaker at 2 s, the current swiftly returns to a steady state with minimal spike current. This absence of inductive and capacitive elements in DC load 1 prevents synchronization issues.

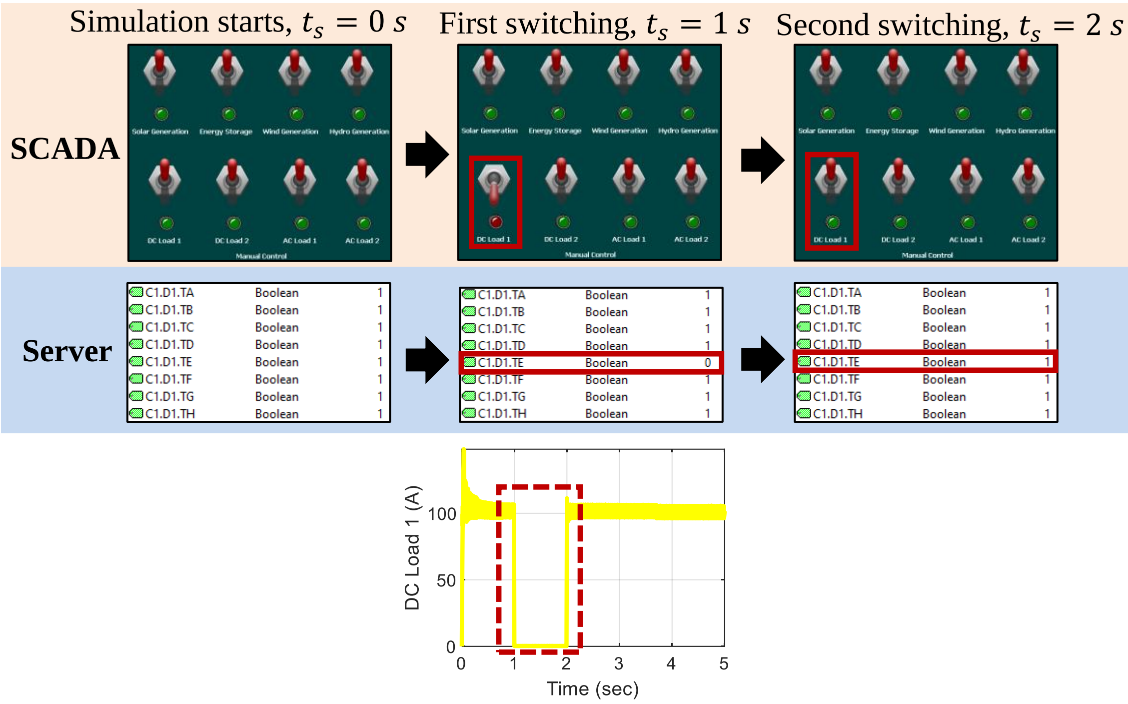


Figure 4.29: Manual Control of DC Load 1

AC fault 1 and DC fault 1 are integrated into the model, facilitated by window scripts employing Equations 3.57 to 3.60 for fault-clearing automation. It is essential to note that green light indicators signify the activation of breakers or switches to clear faults, while red light indicators signal that the breakers or switches remain closed. AC fault 1 occurs between the hydro generator and transformer 1 (refer Figure 3.20). The fault current is measured from the busbar between the circuit breaker and three-phase faults, as depicted in Figure D-4 (e) (Appendix). This fault persists for 0.2 s, spanning from 1.0 s to 1.2 s of simulation time. In Figure 4.30, the fault initiates at t_s = 1 s, causing the fault current to surge to over 30 kA. The signal tag transitions from 1 to 0, indicating the successful and automatic triggering of the breaker, clearing the fault at $t_s \approx 1.03$. A 1-second delay is implemented for the circuit breaker to reclose, ensuring the fault terminates before automatic closure.

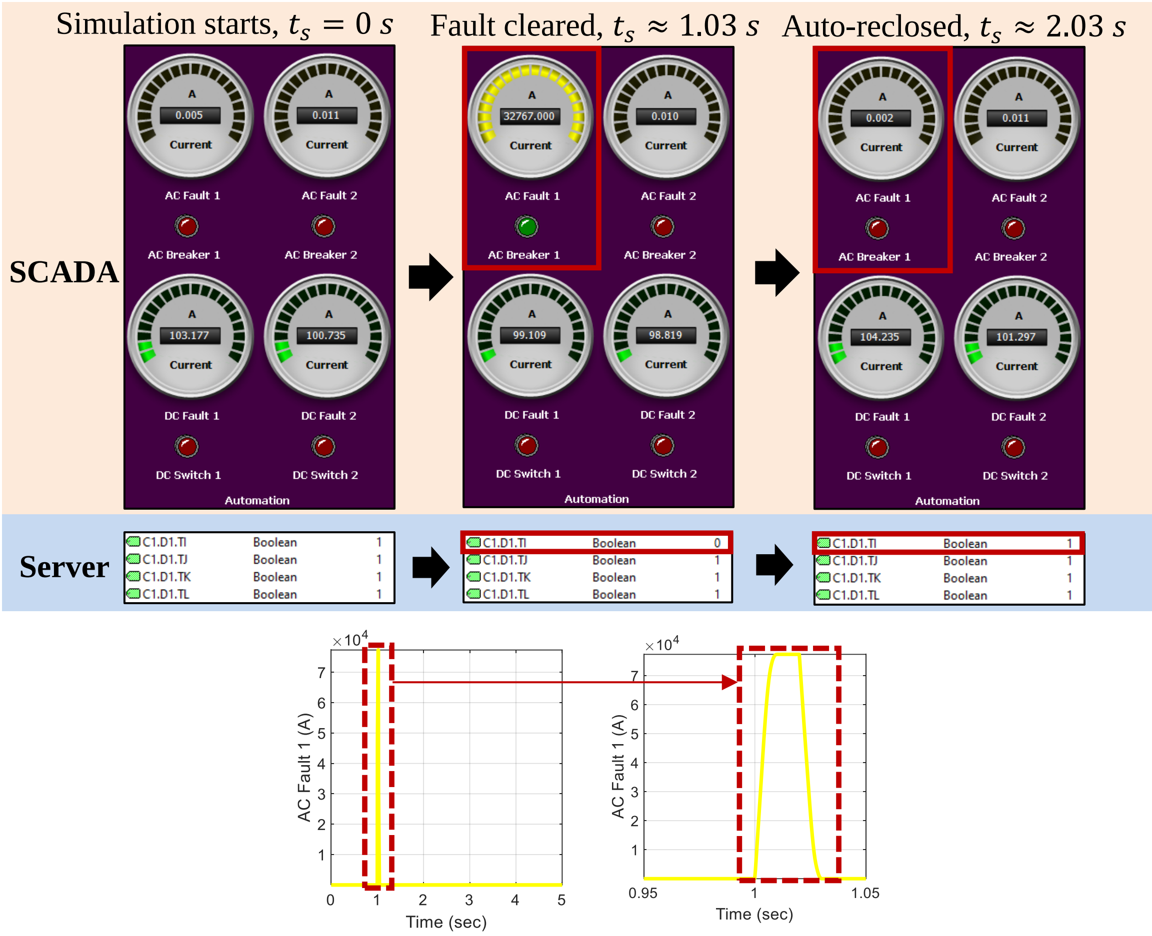


Figure 4.30: Automation Process for Clearing AC Fault 1

Figure 4.30 illustrates the initiation of DC fault 1 through a DC short circuit, with the fault current measurement extracted from the "Goto Block" of IDCLoad1, as indicated in Figure D-4 (a) (Appendix). This setup ensures that the current measurement originates specifically from DC load 1. The DC switch is programmed to trigger automatically if the current surpasses or equals 200 A, as specified by the window scripts. The duration of DC fault 1 is consistent at 0.2 s, spanning from 1 s to 1.2 s of the simulation timeline. The automated operation is confirmed as the switch automatically activates at around 1 s, as indicated by the transition of the TK signal from 1 to 0. Subsequently, the clearance of DC fault 1 swiftly occurs within an extremely short interval around $t_s \approx 1.0002\ s$, as depicted in the results graph presented in Figure 4.31.

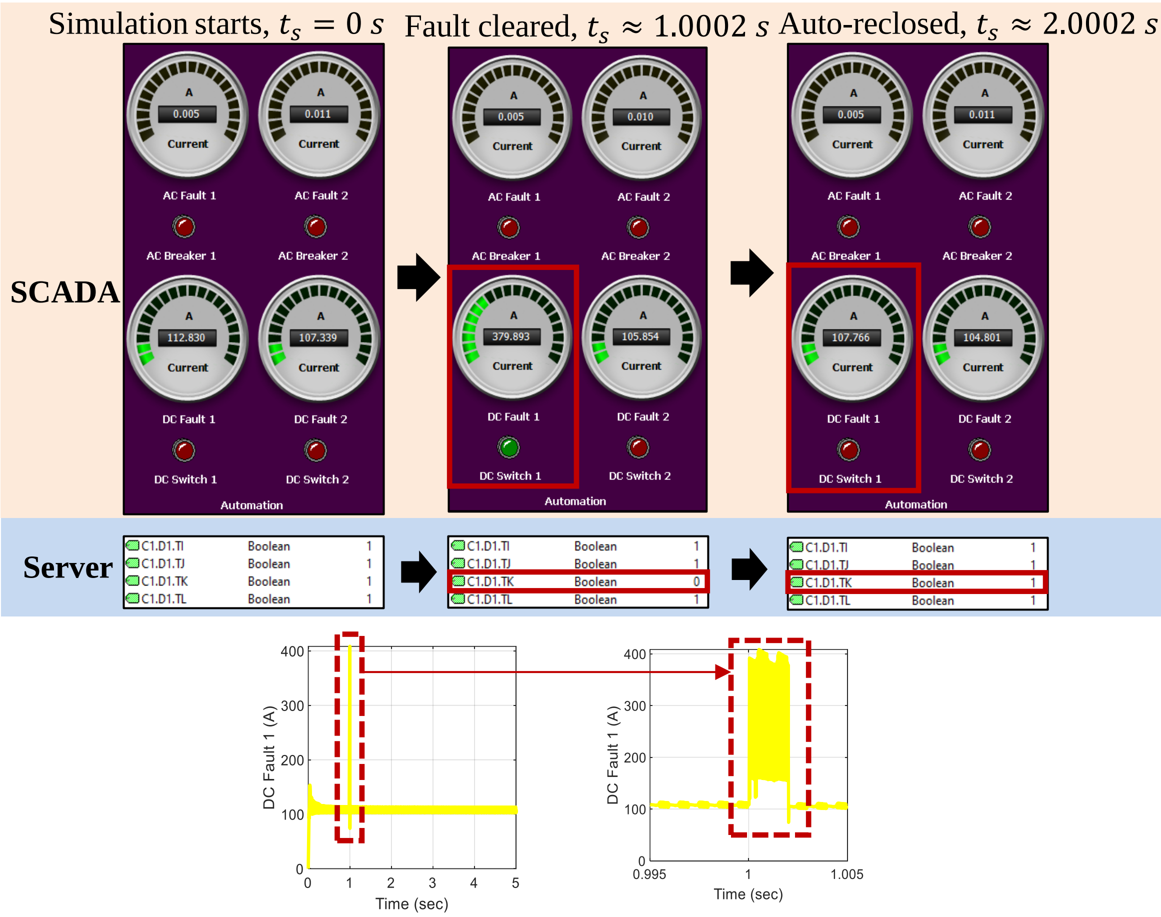


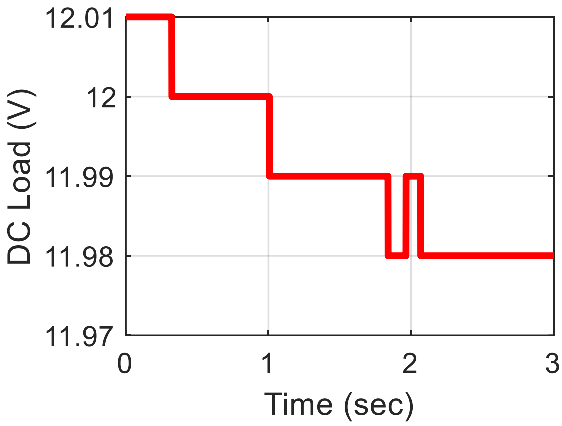
Figure 4.31: Automation Process for Clearing DC Fault 1

4.4.4 Validation Results of Hardware Prototype

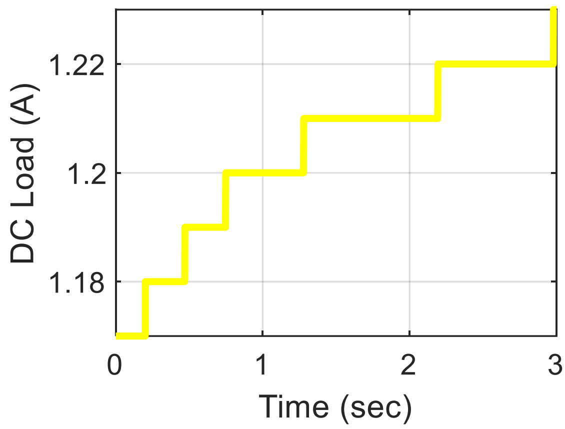
A hardware prototype has been developed to facilitate practical communication data transfer between the SCADA system and the simulation model. This prototype is designed to interface seamlessly with the SCADA system via a server and MATLAB Simulink, enabling real-time monitoring, control, and automation of the actual load. For this purpose, a table fan serves as the AC load, while a DC bulb is employed as the DC load. The operational setup of the hardware prototype is depicted in Figure 4.32.

Figure 4.32: Hardware Prototype for Testing and Validation

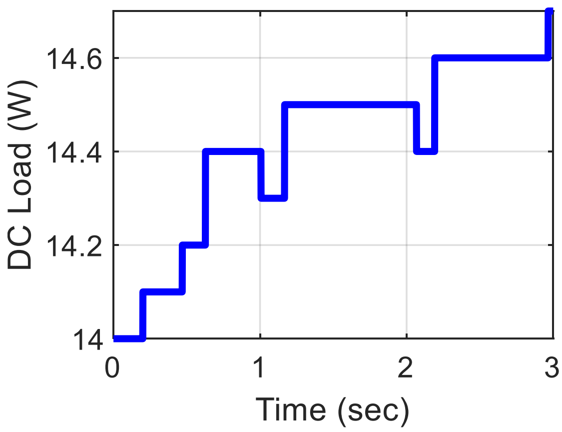
The hardware prototype effectively demonstrates monitoring results in the SCADA system, validated against monitoring data in MATLAB Simulink and the server. Figures 4.33 (a) to (j) exhibit fluctuations in data, reflecting real-time behavior across various power parameters. These monitoring data are streamed from the ThingSpeak cloud, while MATLAB ran simulations for 3 *s* for validation. Additionally, Figure 4.34 displays real-time monitoring data in the server, while Figures 4.35 and 4.36 present real-time monitoring data in the SCADA system.


DC Load (V)
12.01
12
11.99
11.98
11.97
0
1
2
3
Time (sec)

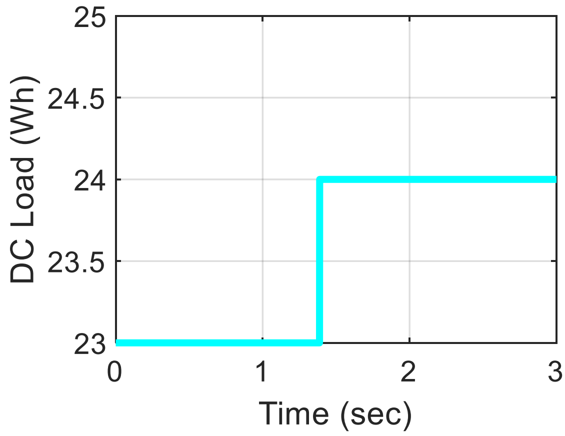
(a)


DC Load (A)
1.22
1.2
1.18
0
1
2
3
Time (sec)

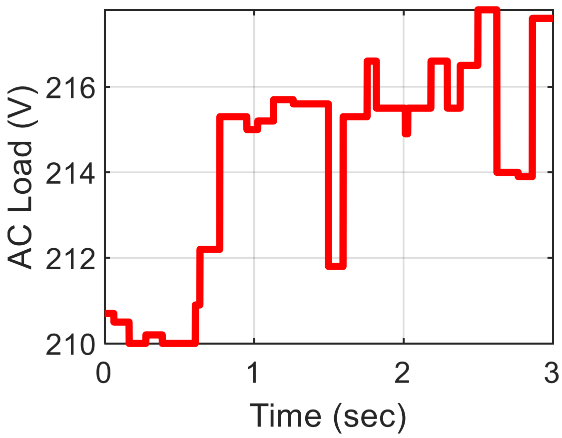
(b)


DC Load (W)
14.6
14.4
14.2
14
0
1
2
3
Time (sec)

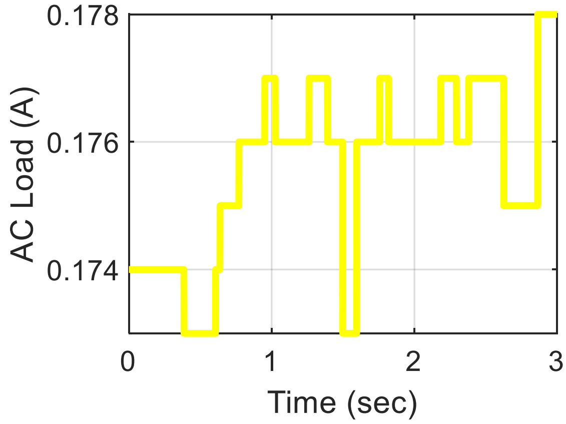
(c)


DC Load (Wh)
25
24.5
24
23.5
23
0
1
2
3
Time (sec)

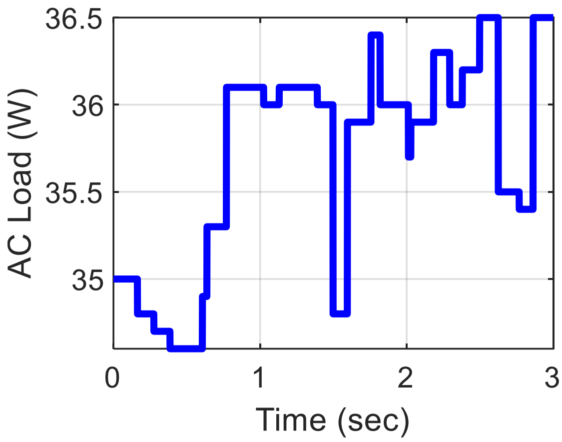
(d)


AC Load (V)
216
214
212
210
0
1
2
3
Time (sec)

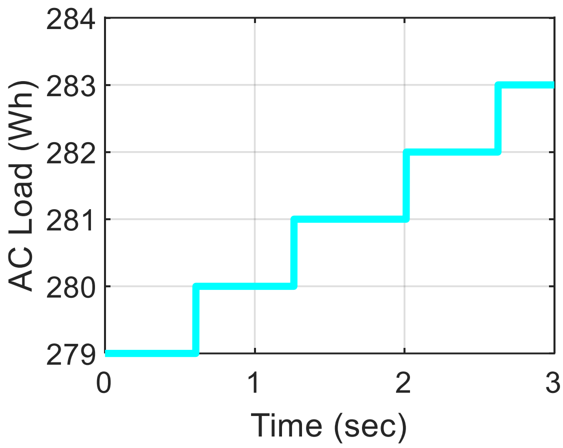
(e)


AC Load (A)
0.178
0.176
0.174
0
1
2
3
Time (sec)

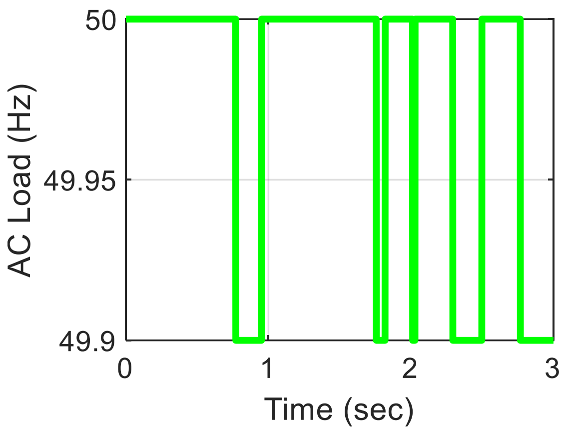
(f)



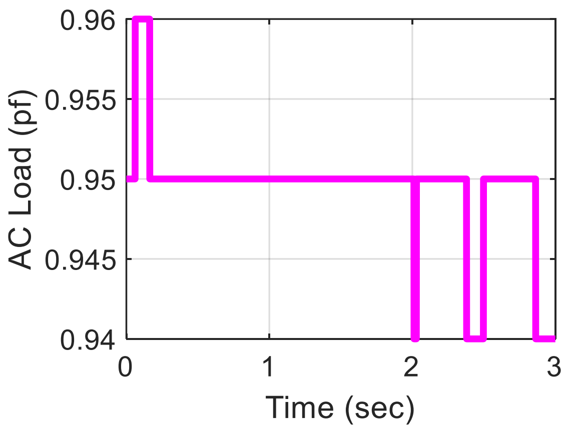
(g)



(h)



(i)



(j)

Figure 4.33 (a) to (j): Real-time Monitoring Data on Actual Loads

Item ID	Data Type	Value	Timestamp	Quality	Update Count
C1.D1.T1	Double	11.99	13:10:50.002	Good	8
C1.D1.T2	Double	1.21	13:11:27.012	Good	17
C1.D1.T3	Double	14.5	13:11:12.989	Good	20
C1.D1.T4	Double	24	13:11:42.017	Good	6
C1.D1.T5	Double	215.3	13:12:14.013	Good	38
C1.D1.T6	Double	0.176	13:12:14.013	Good	28
C1.D1.T7	Double	35.9	13:12:14.029	Good	34
C1.D1.T8	Double	281	13:11:25.026	Good	10
C1.D1.T9	Double	50	13:10:43.013	Good	16
C1.D1.T10	Double	0.95	13:08:58.993	Good	11

(a) Monitoring Data at 1.5 *s*

Item ID	Data Type	Value	Timestamp	Quality	Update Count
C1.D1.T1	Double	11.97	13:15:49.066	Good	12
C1.D1.T2	Double	1.23	13:15:30.065	Good	19
C1.D1.T3	Double	14.7	13:15:28.050	Good	23
C1.D1.T4	Double	25	13:15:49.066	Good	7
C1.D1.T5	Double	216.6	13:16:35.068	Good	53
C1.D1.T6	Double	0.176	13:15:45.064	Good	37
C1.D1.T7	Double	36	13:16:19.093	Good	48
C1.D1.T8	Double	284	13:16:35.083	Good	13
C1.D1.T9	Double	49.9	13:16:35.083	Good	25
C1.D1.T10	Double	0.94	13:16:35.068	Good	18

(b) Monitoring Data at 3.0 *s*

Figure 4.34 (a) to (b): Monitoring Data on Actual Loads in Server

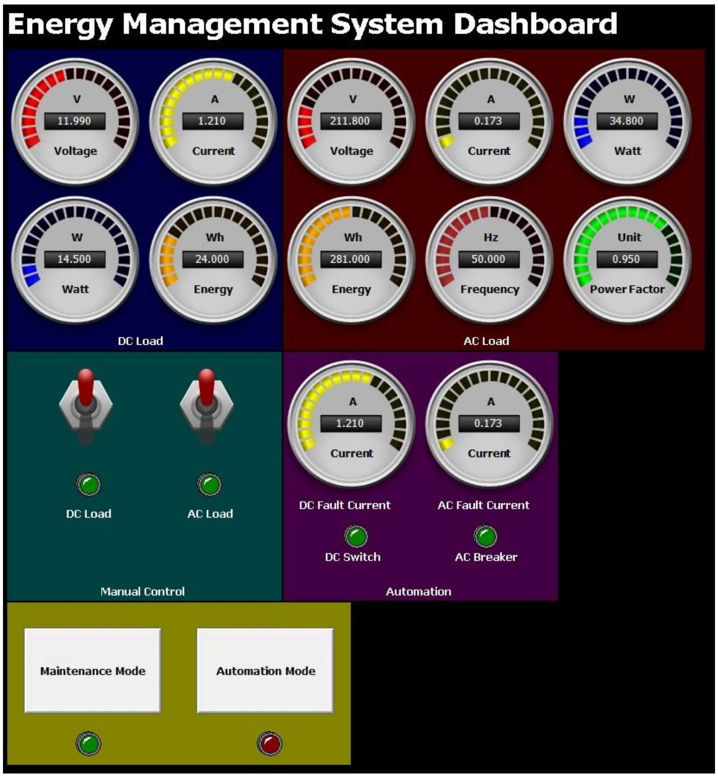


Figure 4.35: Monitoring Data on Actual Loads at 1.5 *s* in SCADA

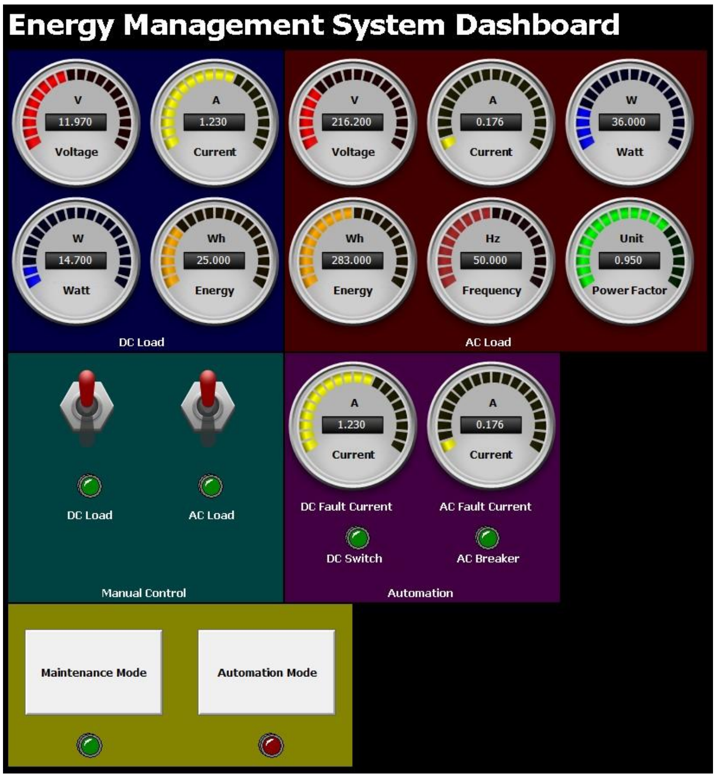


Figure 4.36: Monitoring Data on Actual Loads at 3.0 *s* in SCADA

The maintenance mode is initially engaged to facilitate the switching of the AC load and DC load, validating the real-time manual control functionalities with the hardware prototype. The first switch is executed at approximately 0.5 *s* (for DC load) and 0.6 *s* (for AC load). During this instance, the logical state transitions to 0, signifying the disconnection of the loads. Subsequently, the second switch occurs around 0.9 *s* (for AC load) and 1.0 *s* (for DC load), with the logical state reverting to 1, indicating the reconnection of the loads. The hardware prototype is successfully controlled through manual operation using the SCADA system, as depicted in Figures 4.37 and 4.38.

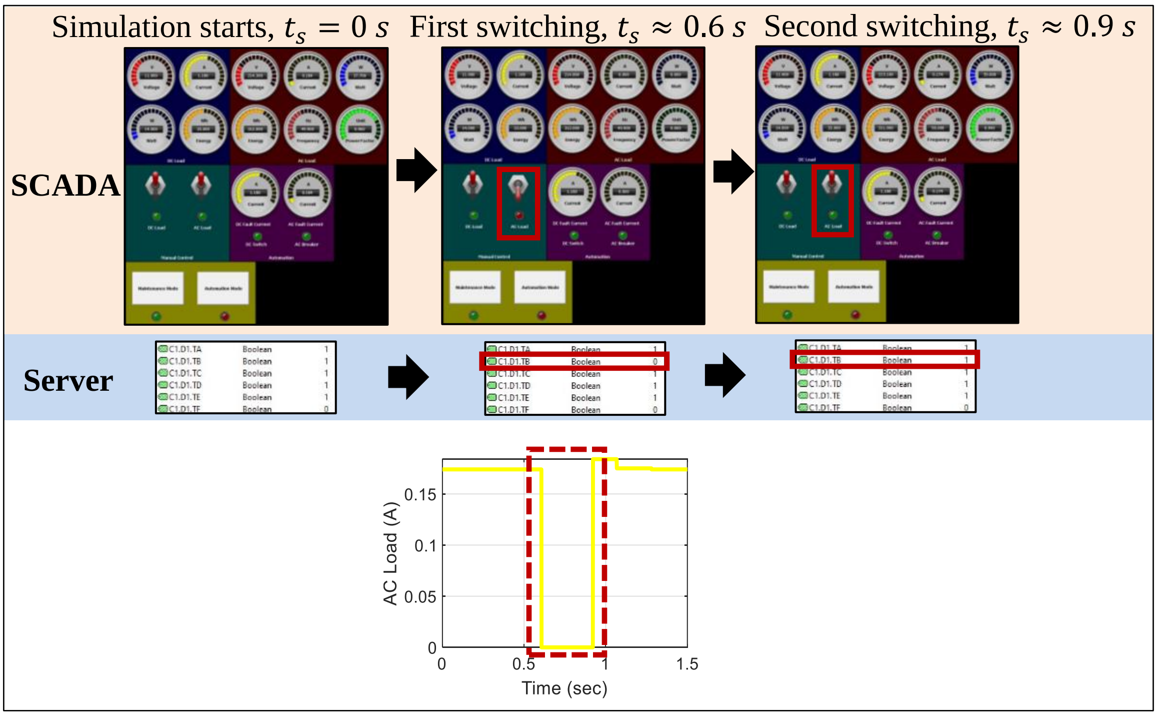


Figure 4.37: Manual Control of Actual AC Load

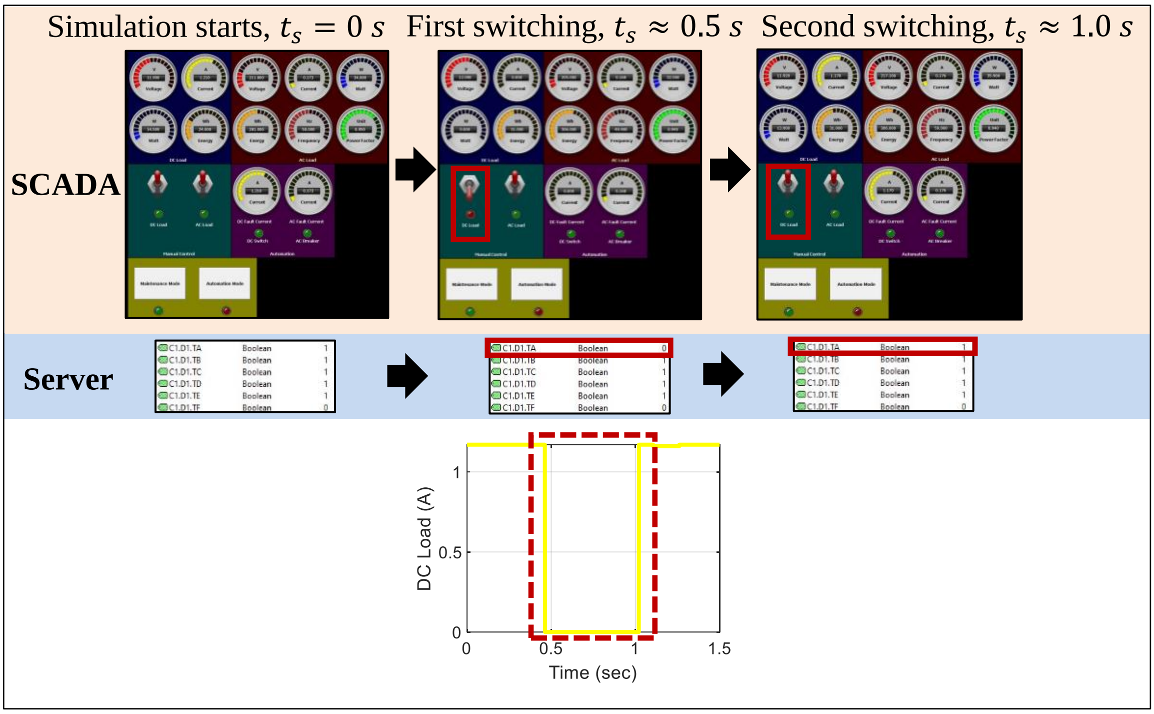


Figure 4.38: Manual Control of Actual DC Load

Fault scenarios for both the AC load and DC load are simulated by adjusting the threshold currents in the conditional script (refer Figure 3.32) to validate the effectiveness of SCADA automation strategies. Given that the steady-state current for the AC load is approximately 0.17 A, the AC threshold current is set to 0.1 A, below the steady-state level. Similarly, for the DC load with a steady-state current of about 1.2 A, the DC threshold current is set to 1 A, also below the steady-state level. These conditions are established to trigger automation operations when the automation mode is activated. At simulation time $t_s = 0\ s$, both AC and DC loads are in maintenance mode, resulting in zero current as the loads are disconnected. At $t_s \approx 0.5\ s$, the maintenance mode is toggled to automation mode, activating all switches for both AC and DC loads. Due to the threshold currents being lower than their respective steady-state currents, the automation strategies are executed, automatically clearing the AC fault around 0.8 s and the DC fault around 0.6 s. These scenarios demonstrate the successful implementation of automation for fault clearing for both AC and DC loads, as depicted in Figures 4.39 and 4.40. The relays reconnect the AC and DC loads only after a 1-second delay, as delay blocks are integrated, as shown in Figure D-5 (Appendix).

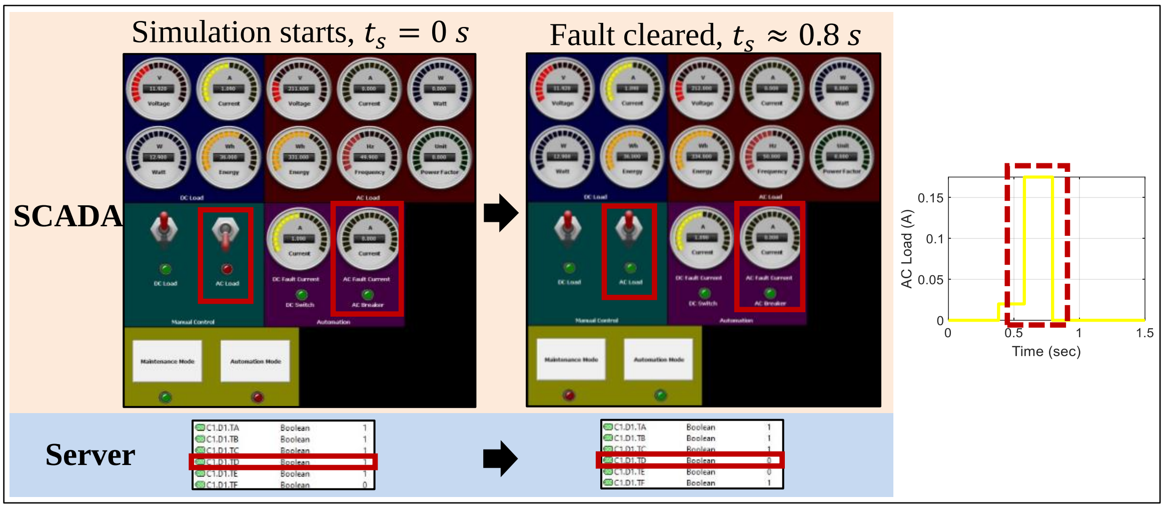


Figure 4.39: Automation Process for Clearing AC Fault

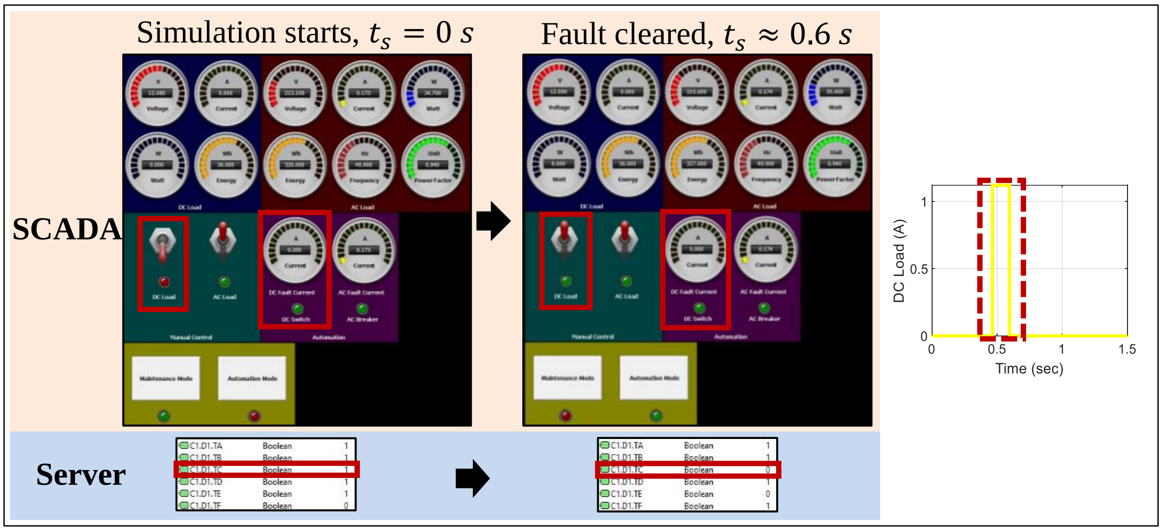


Figure 4.40: Automation Process for Clearing DC Fault

4.5 Chapter Summary

This chapter systematically demonstrates all the results from the proposed works. All the green energy locations in Sarawak have been thoroughly examined to produce 100 SES, 23 WES, and 138 HES. The top 100 optimal SES have been validated using the weighted sum method. Additionally, optimal electrical power lines routing has been designed for all identified green energy locations for integration purposes. The routing for each cluster has been validated and compared with state-of-the-art algorithms to achieve the minimum total distance, minimum elevation difference, and minimum total average ground flash density. In the last section, the results reveal the successful implementation of real-time dynamic monitoring, control, and automation using SCADA systems with MATLAB Simulink through a server. A hardware prototype has been employed to validate the possibility of interfacing it with the SCADA system for real-time dynamic monitoring, control, and automation in a real-world scenario.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Introduction

This chapter presents the overall conclusions drawn from this research, highlighting the key contributions and advancements made in identifying green energy, power line routing, and real-time system integration. The research introduced a novel GIS-based fuzzy TOPSIS and filtration algorithm to identify the suitable green energy locations. Additionally, an advanced GIS-driven fuzzy TSP-BIP algorithm was developed to optimize the integration of these sites into the power grid, ensuring efficient power line routing. Beyond site selection and infrastructure planning, an innovative IIoT-based system was designed for real-time monitoring, control, and automation of the IGESs model. A hardware prototype was implemented and integrated with SCADA to validate the practical applicability of the proposed system. The following sections summarize the research findings, discuss their contributions, and outline limitations and recommendations.

5.2 Research Summary and Findings

This research successfully identified optimal renewable energy sites using a proposed GIS-based fuzzy TOPSIS and filtration algorithm. The methodology initially processed a dataset of 19,237 potential sites, refining it to 17,227 locations based on spatial criteria. Further analysis determined 23 optimal WES with wind speeds above 3 ms^{-1} and filtered 155 HES down to 138 optimal locations. For SES, the fuzzy TOPSIS algorithm evaluated multiple constraints, including climatic, technical, accessibility, environmental, and social factors to identify 1,862 potential sites. A second filtration phase ranked the top

100 SES locations based on their closeness coefficient, with validation against a weighted sum solar suitability map confirming a strong correlation (69.01 %) with high-priority solar zones.

To facilitate the integration of these green energy locations, a GIS-driven fuzzy TSP-BIP algorithm was developed to optimize power line routing in Sarawak. This approach incorporated GIS spatial tools with the proposed fuzzy TSP- BIP algorithm to minimize the distance, elevation differences, and ground flash density. Comparative analyses demonstrated that the proposed method consistently outperformed conventional TSP-BIP approaches across all clusters. Additionally, the validation of the proposed method against seven unique fuzzy TSP algorithms demonstrated its superior performance. The proposed approach consistently recorded the lowest fuzzy values and achieved significantly reduced computational time, confirming its robustness in optimizing power line routing.

Beyond site selection and power line optimization, this research developed an IIoT-based system for real-time monitoring, control, and automation of the IGESs model. This innovative approach retrieves historical data from the grid system operator and Solcast, reuploads it to the ThingSpeak cloud for real-time streaming, and establishes a communication link between MATLAB Simulink and SCADA via a dedicated server for seamless data exchange. A hardware prototype incorporating a Raspberry Pi 4 and other IIoT components was successfully implemented to validate SCADA functionality. Experimental results demonstrated effective real-time monitoring, manual control of both AC and DC loads, and automated fault clearance using SCADA scripting. These findings confirm the practicality and effectiveness of the proposed IIoT-SCADA integration in advancing intelligent green energy system operations.

5.3 Contributions

The proposed novel GIS-based fuzzy TOPSIS and filtration algorithms lies in their ability to thoroughly screen potential green energy locations within a region. These filtration algorithms enhance the capability to identify a large scale of potential green energy locations. Additionally, this research provides readily executable code for the comprehensive proposed algorithm, making it adaptable for use in other case studies to identify potential green energy locations on an extensive scale. Furthermore, the research contributes to the reliable selection, along with coordinates, of the top 100 optimal SES, 23 WES, and 138 HES throughout the Sarawak region (refer Table B-1) (Appendix). The research makes a significant contribution to the field of green energy locations integration by introducing an innovative approach that incorporates the impact of lightning using the GFD parameter. This unique integration of fuzzy logic and GIS tools into the TSP-BIP algorithm has been validated, providing comprehensive algorithms and coding for researchers and investors interested in further exploration of green energy locations integration. Through meticulous evaluation and statistical analysis, the research endeavors not only to demonstrate the comparative merits of fuzzy TSP-BIP over ordinary TSP-BIP but also its supremacy among a spectrum of state-of-the-art fuzzy optimization algorithms. This rigorous validation serves as a foundation for asserting that the fuzzy TSP-BIP method stands out as the ultimate solution, offering unparalleled performance and robustness in addressing complex multi-objective optimization challenges. Beyond technical advancements, the proposed method aims to boost the region's economy and contribute to Sarawak's transition toward a more environmentally conscious future. It establishes optimal power lines routing networks for efficient and resilient GERs utilization. In addition to the aforementioned contributions, the research also significantly contributes to the development of an IIoT-based system for monitoring, control, and automation for IGESs modeling. The modeling of real-time dynamic data provides researchers with valuable insights into the impact of dynamic behavior of input data on their respective models. The integration of a ring topology into the green energy systems model, coupled with dynamic interfacing, encourages more researchers to forecast potential power system issues, enabling proactive planning and management. Furthermore, the successful communication between a) MATLAB Simulink and the SCADA system (simulation model) and b) Hardware prototyping and the SCADA system (hardware simulation) provides valuable assets for researchers to facilitate more effective monitoring and management control. This preparation is crucial for fully utilizing GERs in the upcoming years.

5.4 Limitations and Recommendations

The proposed novel GIS-based fuzzy TOPSIS and filtration algorithms model come with certain limitations and negative effects. Firstly, the preparatory steps are time-consuming, requiring researchers to identify reliable sampling alternatives and utilize structured data in the form of polygon layers and influential criteria in raster maps. Additionally, implementing the model necessitates complex and trustworthy GIS databases for SES identification. Moreover, factors such as input criteria constraints, linguistic fuzzy sets, and fuzzy weights could minimally impact the results, as the MCDM parameters are derived from experts, potentially introducing bias. Furthermore, concerning results validation, the precision of the solar suitability map may be affected by the absence of a filtration function in the weighted sum tool and its inability to accommodate fuzzy values. As recommendations, this research could integrate additional criteria raster layers, such as land controversy, lightning risk, and Gross Domestic Product (GDP), to enhance the reliability of the proposed model for green energy locations identification. To improve the validation of results, obtaining experimental or actual data from specific locations could be beneficial. Additionally, exploring the integration of fuzzy logic into other MCDM techniques, such as AHP, PROMETHEE, VIKOR, ANP, and ELECTRE, could offer crucial insights through parameter analyses, robustness testing, and sensitivity analysis. Moreover, conducting in-depth measurements and empirical research to determine optimal coefficients, especially for factors like lightning can be pursued. This approach aims to reduce human bias in parameter, constraint, and weight definition, ultimately leading to an improved performance of the model.

For the integration of green energy locations, incorporating physical geographical location assessments could enhance result accuracy. Considering additional geographical parameters would also contribute to improving the overall model. Furthermore, conducting techno-economic analyses among parameters to assign weightage would provide clearer insights into their impacts. However, it is imperative to note that the fuzzy rule setting utilizes ρ^n, where ρ is the number of linguistic variables in each input and n represents the number of inputs. This indicates that as the number of inputs increases, the number of fuzzy rules grows exponentially. This presents a challenge in designing the overall model, as it significantly increases memory requirements and computational workload during implementation. Additionally, achieving more accurate results could be possible by increasing the sensitivity level for fuzzy values in fuzzy membership functions. For validation, deploying more TSP optimization algorithms and spending additional time on parameter tuning for state-of-the-art algorithms could provide a more historically accurate and unbiased comparison and validation.

For the IIoT-based system for real-time monitoring, control, and automation for the IGESs model, further investigation, testing, and assessment are warranted. Modeling more power-intelligent electronics and exploring different control strategies, such as employing dynamic voltage regulators and improved voltage source converters, can enhance the power quality of the model. Additionally, as real-time dynamic data modeling has been proposed, researchers can acquire more dynamic data types and integrate them into the model for a more realistic representation and execution. Moreover, the developed hardware prototype can be expanded to have capabilities to measure three-phase induction motors and reactive power. Automation strategies can also be extended beyond fault clearing in the SCADA system to other applications such as load demand control and optimizing energy storage. Furthermore, conducting techno-economic analysis can provide valuable insights into estimating the costs of the overall systems.

REFERENCES

Abdul, D., Wenqi, J., & Tanveer, A. (2022). Prioritization of renewable energy source for electricity generation through AHP-VIKOR integrated methodology. *Renewable Energy*, *184*, 1018–1032. https://doi.org/10.1016/J.RENENE.2021.10.082

Abrams, W., Ghoneim, E., Shew, R., LaMaskin, T., Al-Bloushi, K., Hussein, S., AbuBakr, M., Al-Mulla, E., Al-Awar, M., & El-Baz, F. (2018). Delineation of groundwater potential (GWP) in the northern United Arab Emirates and Oman using geospatial technologies in conjunction with Simple Additive Weight (SAW), Analytical Hierarchy Process (AHP), and Probabilistic Frequency Ratio (PFR) techniques. *Journal of Arid Environments*, *157*, 77–96. https://doi.org/10.1016/J.JARIDENV.2018.05.005

Agarwal, T., & Kumar, V. (2022). A Systematic Review on Bat Algorithm: Theoretical Foundation, Variants, and Applications. *Archives of Computational Methods in Engineering*, *29*(5), 2707–2736. https://doi.org/10.1007/s11831-021-09673-9

Ahmed, M. M., Qays, M. O., Abu-Siada, A., Muyeen, S. M., & Hossain, M. L. (2021). Cost-Effective Design of IoT-Based Smart Household Distribution System. *Designs*, *5*(3), 55. https://doi.org/10.3390/designs5030055

ajevtic. (2024). *Nearest Neighbor algorithm for the Travelling Salesman Problem*. MATLAB Central File Exchange. https://www.mathworks.com/matlabcentral/fileexchange/25542-nearest-neighbor-algorithm-for-the-travelling-salesman-problem

Akram, M., Kahraman, C., & Zahid, K. (2021). Group decision-making based on complex spherical fuzzy VIKOR approach. *Knowledge-Based Systems*, *216*, 106793. https://doi.org/10.1016/J.KNOSYS.2021.106793

Alaidi, A. H., Soong Der, C. S., & Weng Leong, Y. (2021). Systematic Review of Enhancement of Artificial Bee Colony Algorithm Using Ant Colony Pheromone. *International Journal of Interactive Mobile Technologies (IJIM)*, *15*(16), 172. https://doi.org/10.3991/ijim.v15i16.24171

Albogamy, F. R., Khan, S. A., Hafeez, G., Murawwat, S., Khan, S., Haider, S. I., Basit, A., & Thoben, K.-D. (2022). Real-Time Energy Management and Load Scheduling with Renewable Energy Integration in Smart Grid. *Sustainability*, *14*(3), 1792. https://doi.org/10.3390/su14031792

Alghassab, M. (2022). Quantitative assessment of sustainable renewable energy through soft computing: Fuzzy AHP-TOPSIS method. *Energy Reports*, *8*, 12139–12152. https://doi.org/10.1016/J.EGYR.2022.09.049

Ali, M. A., Barakat, M. M., Abokhalaf, M. M., Fadel, Y. H., Kandil, M., Rasmy, M. W., Ali, O. N., Besheer, A. H., Emara, H. M., & Bahgat, A. (2021). Micro-grid Monitoring and Supervision: Web-based SCADA Approach. *Journal of Electrical Engineering & Technology*, *16*(5), 2313–2331. https://doi.org/10.1007/s42835-021-00762-0

Alizadeh, R., Soltanisehat, L., Lund, P. D., & Zamanisabzi, H. (2020). Improving renewable energy policy planning and decision-making through a hybrid MCDM method. *Energy Policy*, *137*, 111174. https://doi.org/10.1016/J.ENPOL.2019.111174

Almasad, A., Pavlak, G., Alquthami, T., & Kumara, S. (2023). Site suitability analysis for implementing solar PV power plants using GIS and fuzzy MCDM based approach. *Solar Energy*, *249*, 642–650. https://doi.org/10.1016/J.SOLENER.2022.11.046

Al-Tashi, Q., Md Rais, H., Abdulkadir, S. J., Mirjalili, S., & Alhussian, H. (2020). A Review of Grey Wolf Optimizer-Based Feature Selection Methods for Classification. In *Evolutionary Machine Learning Techniques* (pp. 273–286). Springer. https://doi.org/10.1007/978-981-32-9990-0_13

Andreopoulou, Z., Koliouska, C., Galariotis, E., & Zopounidis, C. (2018). Renewable energy sources: Using PROMETHEE II for ranking websites to support market opportunities. *Technological Forecasting and Social Change*, *131*, 31–37. https://doi.org/10.1016/J.TECHFORE.2017.06.007

Aravind Seshadri. (2024). *Traveling Salesman Problem (TSP) using Simulated Annealing* . MATLAB Central File Exchange. https://www.mathworks.com/matlabcentral/fileexchange/9612-traveling-salesman-problem-tsp-using-simulated-annealing

Arief, Y. Z., Anak Sa-Or, S. S., Mubarakah, N., Izzwan Saad, M. H., & Eteruddin, H. (2020). Model of grid-connected PV system in Sarawak, Malaysia Rural Area. *2020 4th International Conference on Electrical, Telecommunication and Computer Engineering, ELTICOM 2020 - Proceedings*, 250–259. https://doi.org/10.1109/ELTICOM50775.2020.9230485

Arjomandi, A., Mortazavi, S. A., Khalilian, S., & Garizi, A. Z. (2021). Optimal land-use allocation using MCDM and SWAT for the Hablehroud Watershed, Iran. *Land Use Policy*, *100*, 104930. https://doi.org/10.1016/J.LANDUSEPOL.2020.104930

Asadi, M., & Pourhossein, K. (2021). Wind farm site selection considering turbulence intensity. *Energy*, *236*, 121480. https://doi.org/10.1016/J.ENERGY.2021.121480

Ayough, A., Boshruei, S., & Khorshidvand, B. (2022). A new interactive method based on multi-criteria preference degree functions for solar power plant site selection. *Renewable Energy*, *195*, 1165–1173. https://doi.org/10.1016/J.RENENE.2022.06.087

Baniasadi, P., Foumani, M., Smith-Miles, K., & Ejov, V. (2020). A transformation technique for the clustered generalized traveling salesman problem with applications to logistics. *European Journal of Operational Research*, *285*(2), 444–457. https://doi.org/10.1016/J.EJOR.2020.01.053

Barzehkar, M., Dinan, N. M., Mazaheri, S., Tayebi, R. M., & Brodie, G. I. (2019). Landfill site selection using GIS-based multi-criteria evaluation (case study: SaharKhiz Region located in Gilan Province in Iran). *SN Applied Sciences*, *1*(9), 1–11. https://doi.org/10.1007/s42452-019-1109-9

Bilgili, F., Zarali, F., Ilgün, M. F., Dumrul, C., & Dumrul, Y. (2022). The evaluation of renewable energy alternatives for sustainable development in Turkey using intuitionistic fuzzy-TOPSIS method. *Renewable Energy*, *189*, 1443–1458. https://doi.org/10.1016/J.RENENE.2022.03.058

Bohra, S. S., & Anvari-Moghaddam, A. (2022). A comprehensive review on applications of multicriteria decision-making methods in power and energy systems. *International Journal of Energy Research*, *46*(4), 4088–4118. https://doi.org/10.1002/ER.7517

Boran, F. E., Boran, K., & Menlik, T. (2012). The Evaluation of Renewable Energy Technologies for Electricity Generation in Turkey Using Intuitionistic Fuzzy TOPSIS.

Energy Sources, Part B: Economics, Planning, and Policy, *7*(1), 81–90. https://doi.org/10.1080/15567240903047483

Cao, Y., Kou, Y. X., Xu, A., & Xi, Z. F. (2021). Target Threat Assessment in Air Combat Based on Improved Glowworm Swarm Optimization and ELM Neural Network. *International Journal of Aerospace Engineering*, *2021*. https://doi.org/10.1155/2021/4687167

Chaves, A. A., Vianna, B. L., da Silva, T. T., & Schenekemberg, C. M. (2024). A parallel branch-and-cut and an adaptive metaheuristic to solve the Family Traveling Salesman Problem. *Expert Systems with Applications*, *238*, 121735. https://doi.org/10.1016/J.ESWA.2023.121735

Chen, T., Wang, Y. ting, Wang, J. qiang, Li, L., & Cheng, P. F. (2020). Multistage Decision Framework for the Selection of Renewable Energy Sources Based on Prospect Theory and PROMETHEE. *International Journal of Fuzzy Systems*, *22*(5), 1535–1551. https://doi.org/10.1007/S40815-020-00858-1/METRICS

Chen, Z. S., Yu, C., Chin, K. S., & Martínez, L. (2019). An enhanced ordered weighted averaging operators generation algorithm with applications for multicriteria decision making. *Applied Mathematical Modelling*, *71*, 467–490. https://doi.org/10.1016/J.APM.2019.02.042

Chen, Z. S., Zhang, X., Rodríguez, R. M., Pedrycz, W., & Martínez, L. (2021). Expertise-based bid evaluation for construction-contractor selection with generalized comparative linguistic ELECTRE III. *Automation in Construction*, *125*, 103578. https://doi.org/10.1016/J.AUTCON.2021.103578

Chinomi, N., Leelajindakrairerk, M., Boontaklang, S., & Chompoo-Inwai, C. (2017). Design and Implementation of a smart monitoring system of a modern renewable energy micro-grid system using a low-cost data acquisition system and LabVIEW program. *Journal of International Council on Electrical Engineering*, *7*(1), 142–152. https://doi.org/10.1080/22348972.2017.1345226

Chou, X., Gambardella, L. M., & Montemanni, R. (2021). A Tabu Search algorithm for the Probabilistic Orienteering Problem. *Computers & Operations Research*, *126*, 105107. https://doi.org/10.1016/j.cor.2020.105107

Colak, H. E., Memisoglu, T., & Gercek, Y. (2020). Optimal site selection for solar photovoltaic (PV) power plants using GIS and AHP: A case study of Malatya Province, Turkey. *Renewable Energy*, *149*, 565–576. https://doi.org/10.1016/J.RENENE.2019.12.078

Csiszar, O. (2021). Ordered Weighted Averaging Operators: A Short Review. *IEEE Systems, Man, and Cybernetics Magazine*, *7*(2), 4–12. https://doi.org/10.1109/MSMC.2020.3036378

Cuong-Le, T., Minh, H.-L., Khatir, S., Wahab, M. A., Tran, M. T., & Mirjalili, S. (2021). A novel version of Cuckoo search algorithm for solving optimization problems. *Expert Systems with Applications*, *186*, 115669. https://doi.org/10.1016/j.eswa.2021.115669

Delahaye, D., Chaimatanan, S., & Mongeau, M. (2019). *Simulated Annealing: From Basics to Applications* (pp. 1–35). Springer International Publishing. https://doi.org/10.1007/978-3-319-91086-4_1

Deng, W., Shang, S., Cai, X., Zhao, H., Song, Y., & Xu, J. (2021). An improved differential evolution algorithm and its application in optimization problem. *Soft Computing*, *25*(7), 5277–5298. https://doi.org/10.1007/s00500-020-05527-x

Deng, X., & Lv, T. (2020). Power system planning with increasing variable renewable energy: A review of optimization models. *Journal of Cleaner Production*, *246*, 118962. https://doi.org/10.1016/j.jclepro.2019.118962

Deveci, M., Cali, U., & Pamucar, D. (2021). Evaluation of criteria for site selection of solar photovoltaic (PV) projects using fuzzy logarithmic additive estimation of weight coefficients. *Energy Reports*, *7*, 8805–8824. https://doi.org/10.1016/J.EGYR.2021.10.104

Deveci, M., Pamucar, D., & Oguz, E. (2022). Floating photovoltaic site selection using fuzzy rough numbers based Logarithmic Additive Assessment of the Weight coefficients (LAAW) and Ranking of Alternatives through Functional mapping of criterion subintervals into a Single Interval (RAFSI) model. *Applied Energy*, *324*, 119597. https://doi.org/10.1016/J.APENERGY.2022.119597

Di Placido, A., Archetti, C., & Cerrone, C. (2022). A genetic algorithm for the close-enough traveling salesman problem with application to solar panels diagnostic reconnaissance. *Computers & Operations Research*, *145*, 105831. https://doi.org/10.1016/J.COR.2022.105831

DIVA-GIS. (2023, May 27). *DIVA-GIS*. DIVA-GIS. https://www.diva-gis.org/gdata

Doorga, J. R. S., Rughooputh, S. D. D. V., & Boojhawon, R. (2019). Multi-criteria GIS-based modelling technique for identifying potential solar farm sites: A case study in

Mauritius. *Renewable Energy*, *133*, 1201–1219. https://doi.org/10.1016/J.RENENE.2018.08.105

Durin, X. O., Ming, B. M. X., Karunakaran, P. P., Sin, L. L. W., Nabipour Afrouzi, H., & Mehranzamir, K. (2022). A comprehensive review of the environmental impacts of hydropower projects in Sarawak, Malaysia. *Future Energy*, *1*(3), 1–10. https://doi.org/10.55670/fpll.fuen.1.3.1

Dutta, P., Jaikumar, B., & Arora, M. S. (2022). Applications of data envelopment analysis in supplier selection between 2000 and 2020: a literature review. *Annals of Operations Research*, *315*(2), 1399–1454. https://doi.org/10.1007/S10479-021-03931-6/METRICS

EARTHDATA. (2024, January 23). *Global Hydrometeorology Resource Center*. NASA. https://search.earthdata.nasa.gov/search?portal=ghrc&q=lightning

Eben Upton. (2023). *Raspberry Pi 5: available now!* Raspberry Pi. https://www.raspberrypi.com/news/raspberry-pi-5-available-now/

Effat, H. A., & El-Zeiny, A. M. (2022). Geospatial modeling for selection of optimum sites for hybrid solar-wind energy in Assiut Governorate, Egypt. *The Egyptian Journal of Remote Sensing and Space Science*, *25*(2), 627–637. https://doi.org/10.1016/J.EJRS.2022.03.005

Elkadeem, M. R., Younes, A., Sharshir, S. W., Campana, P. E., & Wang, S. (2021). Sustainable siting and design optimization of hybrid renewable energy system: A geospatial multi-criteria analysis. *Applied Energy*, *295*, 117071. https://doi.org/10.1016/J.APENERGY.2021.117071

Emovon, I., & Oghenenyerovwho, O. S. (2020). Application of MCDM method in material selection for optimal design: A review. *Results in Materials*, *7*, 100115. https://doi.org/10.1016/J.RINMA.2020.100115

Estelaji, F., Naseri, A., Keshavarzzadeh, M., Zahedi, R., Yousefi, H., & Ahmadi, A. (2023). Potential measurement and spatial priorities determination for gas station construction using WLC and GIS. *Future Technology* , *2*(4), 24–32. https://doi.org/10.55670/fpll.futech.2.4.3

Farhadinia, B., & Xu, Z. (2019). Information Measures for Hesitant Fuzzy Sets and Their Extensions. *Uncertainty and Operations Research*, *1*, XI, 120. https://doi.org/10.1007/978-981-13-3729-1

Farsi, M., Tao, M., & Wang, X. (2023). An Integrated MCDM Model for Sustainable Course Planning: An Empirical Case Study in Accounting Education. *Sustainability*, *15*(6), 5024. https://doi.org/10.3390/SU15065024

Fidanova, S. (2021). Ant Colony Optimization. In *Ant Colony Optimization and Applications* (Vol. 947, pp. 3–8). Springer International Publishing. https://doi.org/10.1007/978-3-030-67380-2_2

Firozjaei, M. K., Nematollahi, O., Mijani, N., Shorabeh, S. N., Firozjaei, H. K., & Toomanian, A. (2019). An integrated GIS-based Ordered Weighted Averaging analysis for solar energy evaluation in Iran: Current conditions and future planning. *Renewable Energy*, *136*, 1130–1146. https://doi.org/10.1016/J.RENENE.2018.09.090

Fotova Čiković, K., Martinčević, I., & Lozić, J. (2022). Application of Data Envelopment Analysis (DEA) in the Selection of Sustainable Suppliers: A Review and Bibliometric Analysis. *Sustainability*, *14*(11), 6672. https://doi.org/10.3390/SU14116672

Fu, H. H., & Tzeng, S. Y. (2016). Applying Fuzzy Multiple Criteria Decision Making approach to establish safety-management system for hot spring hotels. *Http://Dx.Doi.Org/10.1080/10941665.2016.1175487*, *21*(12), 1343–1356. https://doi.org/10.1080/10941665.2016.1175487

Gad, A. G. (2022). Particle Swarm Optimization Algorithm and Its Applications: A Systematic Review. *Archives of Computational Methods in Engineering*, *29*(5), 2531–2561. https://doi.org/10.1007/s11831-021-09694-4

Genç, S., Akay, D., Boran, F. E., & Yager, R. R. (2020). Linguistic summarization of fuzzy social and economic networks: an application on the international trade network. *Soft Computing*, *24*(2), 1511–1527. https://doi.org/10.1007/S00500-019-03982-9/METRICS

Gmira, M., Gendreau, M., Lodi, A., & Potvin, J.-Y. (2021). Tabu search for the time-dependent vehicle routing problem with time windows on a road network. *European Journal of Operational Research*, *288*(1), 129–140. https://doi.org/10.1016/j.ejor.2020.05.041

Gribiss, H., Aghelinejad, M. M., & Yalaoui, F. (2023). Configuration Selection for Renewable Energy Community Using MCDM Methods. *Energies*, *16*(6), 2632. https://doi.org/10.3390/en16062632

Grid System Operator. (2024). *System Demand Profile*. Grid System Operator. https://www.gso.org.my/SystemData/SystemDemand.aspx

Gul, M., Celik, E., Aydin, N., Gumus, A. T., & Guneri, A. F. (2016). A state of the art literature review of VIKOR and its fuzzy extensions on applications. *Applied Soft Computing*, *46*, 60–89.

https://www.academia.edu/90377789/A_state_of_the_art_literature_review_of_VIKOR_and_its_fuzzy_extensions_on_applications

Günen, M. A. (2021). A comprehensive framework based on GIS-AHP for the installation of solar PV farms in Kahramanmaraş, Turkey. *Renewable Energy*, *178*, 212–225. https://doi.org/10.1016/J.RENENE.2021.06.078

Guo, C., Tang, H., Niu, B., & Boon Patrick Lee, C. (2021). A survey of bacterial foraging optimization. *Neurocomputing*, *452*, 728–746. https://doi.org/10.1016/j.neucom.2020.06.142

Gupta, V., Sharma, M., Pachauri, R. K., & Babu, K. N. D. (2021). A Low-Cost Real-Time IOT Enabled Data Acquisition System for Monitoring of PV System. *Energy Sources, Part A: Recovery, Utilization, and Environmental Effects*, *43*(20), 2529–2543. https://doi.org/10.1080/15567036.2020.1844351

Haas, J., Prieto-Miranda, L., Ghorbani, N., & Breyer, C. (2022). Revisiting the potential of pumped-hydro energy storage: A method to detect economically attractive sites. *Renewable Energy*, *181*, 182–193. https://doi.org/10.1016/J.RENENE.2021.09.009

Hasti, F., Mamkhezri, J., McFerrin, R., & Pezhooli, N. (2023). Optimal solar photovoltaic site selection using geographic information system–based modeling techniques and assessing environmental and economic impacts: The case of Kurdistan. *Solar Energy*, *262*, 111807. https://doi.org/10.1016/J.SOLENER.2023.111807

Heule, M. J. H., & Kullmann, O. (2017). The science of brute force. *Communications of the ACM*, *60*(8), 70–79. https://doi.org/10.1145/3107239

hossein. (2024). *tsp with ga*. MATLAB Central File Exchange. https://www.mathworks.com/matlabcentral/fileexchange/38381-tsp-with-ga

Huang, Y. F., Ng, J. L., Fung, K. F., Weng, T. K., AlDahoul, N., Ahmed, A. N., Sherif, M., Chaplot, B., Chong, K. L., & Elshafie, A. (2023). Space–time heterogeneity of drought characteristics in Sabah and Sarawak, East Malaysia: implications for developing effective drought monitoring and mitigation strategies. *Applied Water Science*, *13*(10), 1–25. https://doi.org/10.1007/S13201-023-01989-0/FIGURES/12

Ilbahar, E., Cebi, S., & Kahraman, C. (2019). A state-of-the-art review on multi-attribute renewable energy decision making. *Energy Strategy Reviews*, *25*, 18–33. https://doi.org/10.1016/J.ESR.2019.04.014

İLHAN, İ. (2021). An improved simulated annealing algorithm with crossover operator for capacitated vehicle routing problem. *Swarm and Evolutionary Computation*, *64*, 100911. https://doi.org/10.1016/j.swevo.2021.100911

Irik Mukhametzyanov. (2024). *MCDM Sens_Analysis*. MATLAB Central File Exchange. https://www.mathworks.com/matlabcentral/fileexchange/67803-mcdm-sens_analysis

Isia, I., Hadibarata, T., Jusoh, M. N. H., Bhattacharjya, R. K., Shahedan, N. F., Bouaissi, A., Fitriyani, N. L., & Syafrudin, M. (2022). Drought Analysis Based on Standardized Precipitation Evapotranspiration Index and Standardized Precipitation Index in Sarawak, Malaysia. *Sustainability 2023, Vol. 15, Page 734*, *15*(1), 734. https://doi.org/10.3390/SU15010734

Jafari, M., Fazloula, R., Effati, M., & Jamali, A. (2021). Providing a GIS-based framework for Run-Of-River hydropower site selection: a model based on sustainable development energy approach. *Https://Doi.Org/10.1080/10286608.2021.1893310*, *38*(2), 102–126. https://doi.org/10.1080/10286608.2021.1893310

Jain, P., & Potdar, V. (2021). Frameworks for developing an Agro-Prosumer Community Group platform. *PeerJ. Computer Science*, *7*, e765. https://doi.org/10.7717/PEERJ-CS.765

Jawad, K., Mahto, R., Das, A., Ahmed, S. U., Aziz, R. M., & Kumar, P. (2023). Novel Cuckoo Search-Based Metaheuristic Approach for Deep Learning Prediction of Depression. *Applied Sciences*, *13*(9), 5322. https://doi.org/10.3390/APP13095322

Jbaihi, O., Ouchani, F. zahra, Alami Merrouni, A., Cherkaoui, M., Ghennioui, A., & Maaroufi, M. (2022). An AHP-GIS based site suitability analysis for integrating large-scale hybrid CSP+PV plants in Morocco: An approach to address the intermittency of solar energy. *Journal of Cleaner Production*, *369*, 133250. https://doi.org/10.1016/J.JCLEPRO.2022.133250

Jong, F. C., & Ahmed, M. M. (2024). Novel GIS-based fuzzy TOPSIS and filtration algorithms for extra-large scale optimal solar energy sites identification. *Solar Energy*, *268*, 112274. https://doi.org/10.1016/J.SOLENER.2023.112274

Jong, F. C., Ahmed, M. M., Lau, W. K., & Denis Lee, H. A. (2022). A new hybrid Artificial Intelligence (AI) approach for hydro energy sites selection and integration. *Heliyon*, *8*(9). https://doi.org/10.1016/j.heliyon.2022.e10638

Karczmarczyk, A., Jankowski, J., & Wątróbski, J. (2018). Multi-criteria decision support for planning and evaluation of performance of viral marketing campaigns in social networks. *PLOS ONE*, *13*(12), e0209372. https://doi.org/10.1371/JOURNAL.PONE.0209372

Karipoğlu, F., Serdar Genç, M., & Akarsu, B. (2022). GIS-based optimal site selection for the solar-powered hydrogen fuel charge stations. *Fuel*, *324*, 124626. https://doi.org/10.1016/J.FUEL.2022.124626

Katoch, S., Chauhan, S. S., & Kumar, V. (2021). A review on genetic algorithm: past, present, and future. *Multimedia Tools and Applications*, *80*(5), 8091–8126. https://doi.org/10.1007/s11042-020-10139-6

Kaya, E., Gorkemli, B., Akay, B., & Karaboga, D. (2022). A review on the studies employing artificial bee colony algorithm to solve combinatorial optimization problems. *Engineering Applications of Artificial Intelligence*, *115*, 105311. https://doi.org/10.1016/j.engappai.2022.105311

Kaya, İ., Çolak, M., & Terzi, F. (2018). Use of MCDM techniques for energy policy and decision-making problems: A review. *International Journal of Energy Research*, *42*(7), 2344–2372. https://doi.org/10.1002/ER.4016

Kee, K. K., Ting, H. Y., Lim, Y. S., Ting, J. T. W., Peter, M., Ibrahim, K., & Show, P. L. (2022). Feasibility of UTS Smart Home to Support Sustainable Development Goals of United Nations (UN SDGs): Water and Energy Conservation. *Sustainability 2022, Vol. 14, Page 12242*, *14*(19), 12242. https://doi.org/10.3390/SU141912242

Kou, G., Yang, P., Peng, Y., Xiao, F., Chen, Y., & Alsaadi, F. E. (2020). Evaluation of feature selection methods for text classification with small datasets using multiple criteria decision-making methods. *Applied Soft Computing*, *86*, 105836. https://doi.org/10.1016/J.ASOC.2019.105836

Krishna Rao, C., Sahoo, S. K., & Yanine, F. F. (2024). An IoT-based intelligent smart energy monitoring system for solar PV power generation. *Energy Harvesting and Systems*, *11*(1). https://doi.org/10.1515/ehs-2023-0015

Kumar, V., & Kumar, D. (2021). A Systematic Review on Firefly Algorithm: Past, Present, and Future. *Archives of Computational Methods in Engineering*, *28*(4), 3269–3291. https://doi.org/10.1007/s11831-020-09498-y

Kuriqi, A., Pinheiro, A. N., Sordo-Ward, A., Bejarano, M. D., & Garrote, L. (2021). Ecological impacts of run-of-river hydropower plants—Current status and future prospects on the brink of energy transition. *Renewable and Sustainable Energy Reviews*, *142*, 110833. https://doi.org/10.1016/J.RSER.2021.110833

Lawan, S. M., Abidin, W. A. W. Z., & Masri, T. (2020). Implementation of a topographic artificial neural network wind speed prediction model for assessing onshore wind power potential in Sibu, Sarawak. *Egyptian Journal of Remote Sensing and Space Science*, *23*(1), 21–34. https://doi.org/10.1016/J.EJRS.2019.08.003

Lei, L., Tan, Y., Dahlenburg, G., Xiang, W., & Zheng, K. (2021). Dynamic Energy Dispatch Based on Deep Reinforcement Learning in IoT-Driven Smart Isolated Microgrids. *IEEE Internet of Things Journal*, *8*(10), 7938–7953. https://doi.org/10.1109/JIOT.2020.3042007

Li, C., Conejo, A. J., Liu, P., Omell, B. P., Siirola, J. D., & Grossmann, I. E. (2022). Mixed-integer linear programming models and algorithms for generation and transmission expansion planning of power systems. *European Journal of Operational Research*, *297*(3), 1071–1082. https://doi.org/10.1016/j.ejor.2021.06.024

Li, S., Gu, Q., Gong, W., & Ning, B. (2020). An enhanced adaptive differential evolution algorithm for parameter extraction of photovoltaic models. *Energy Conversion and Management*, *205*, 112443. https://doi.org/10.1016/j.enconman.2019.112443

Li, Y., Sun, B., Zeng, Y., Dong, S., Ma, S., & Zhang, X. (2022). Active distribution network active and reactive power coordinated dispatching method based on discrete monkey algorithm. *International Journal of Electrical Power & Energy Systems*, *143*, 108425. https://doi.org/10.1016/j.ijepes.2022.108425

Lim, Y. J., & Afifah Basri, N. (2022). Sites selection for a potential radioactive waste repository in Peninsular Malaysia: GIS-based weight linear combination and multi-criteria decision-making analysis. *Progress in Nuclear Energy*, *149*, 104252. https://doi.org/10.1016/J.PNUCENE.2022.104252

Lin, J., Zhu, L., & Gao, K. (2020). A genetic programming hyper-heuristic approach for the multi-skill resource constrained project scheduling problem. *Expert Systems with Applications*, *140*, 112915. https://doi.org/10.1016/j.eswa.2019.112915

Liu, D., Xue, S., Zhao, B., Luo, B., & Wei, Q. (2021). Adaptive Dynamic Programming for Control: A Survey and Recent Advances. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, *51*(1), 142–160. https://doi.org/10.1109/TSMC.2020.3042876

Liu, J., Li, D., Wu, Y., & Liu, D. (2020). Lion swarm optimization algorithm for comparative study with application to optimal dispatch of cascade hydropower stations. *Applied Soft Computing*, *87*, 105974. https://doi.org/10.1016/j.asoc.2019.105974

Liu, Y., Eckert, C. M., & Earl, C. (2020). A review of fuzzy AHP methods for decision-making with subjective judgements. *Expert Systems with Applications*, *161*, 113738. https://doi.org/10.1016/J.ESWA.2020.113738

Liu, Y., Heidari, A. A., Cai, Z., Liang, G., Chen, H., Pan, Z., Alsufyani, A., & Bourouis, S. (2022). Simulated annealing-based dynamic step shuffled frog leaping algorithm: Optimal performance design and feature selection. *Neurocomputing*, *503*, 325–362. https://doi.org/10.1016/j.neucom.2022.06.075

Lu, Y., Xu, Y., & Herrera-Viedma, E. (2022). Consensus progress for large-scale group decision making in social networks with incomplete probabilistic hesitant fuzzy information. *Applied Soft Computing*, *126*, 109249. https://doi.org/10.1016/J.ASOC.2022.109249

Maaroof, B. B., Rashid, T. A., Abdulla, J. M., Hassan, B. A., Alsadoon, A., Mohammadi, M., Khishe, M., & Mirjalili, S. (2022). Current Studies and Applications of Shuffled Frog Leaping Algorithm: A Review. *Archives of Computational Methods in Engineering*, *29*(5), 3459–3474. https://doi.org/10.1007/s11831-021-09707-2

Mahmoudinazlou, S., & Kwon, C. (2024). A hybrid genetic algorithm for the min–max Multiple Traveling Salesman Problem. *Computers & Operations Research*, *162*, 106455. https://doi.org/10.1016/J.COR.2023.106455

Mansini, R., Ogryczak, W., & Speranza, M. G. (2015). *Linear and Mixed Integer Programming for Portfolio Optimization*. Springer International Publishing. https://doi.org/10.1007/978-3-319-18482-1

Mao, M., Chen, S., Zhao, L., Feng, X., & Ma, F. (2023). Pavement PV array reconfiguration strategy based on traveling salesman problem. *Energy*, *284*, 129324. https://doi.org/10.1016/J.ENERGY.2023.129324

Marcelino, C. G., Leite, G. M. C., Delgado, C. A. D. M., de Oliveira, L. B., Wanner, E. F., Jiménez-Fernández, S., & Salcedo-Sanz, S. (2021). An efficient multi-objective evolutionary approach for solving the operation of multi-reservoir system scheduling in hydro-power plants. *Expert Systems with Applications*, *185*, 115638. https://doi.org/10.1016/J.ESWA.2021.115638

MARILYN TEN. (2023, April 9). *S'wak Energy working hard to restore electricity to urban areas in Sibu*. Borneo Post Online. https://www.theborneopost.com/2023/04/09/swak-energy-working-hard-to-restore-electricity-to-urban-areas-in-sibu/

Martínez-Martínez, Y., Dewulf, J., & Casas-Ledón, Y. (2022). GIS-based site suitability analysis and ecosystem services approach for supporting renewable energy development in south-central Chile. *Renewable Energy*, *182*, 363–376. https://doi.org/10.1016/J.RENENE.2021.10.008

Mary, S. A. S. A., & Suganya, G. (2016). Multi-Criteria Decision Making Using ELECTRE. *Circuits and Systems*, *7*(6), 1008–1020. https://doi.org/10.4236/cs.2016.76085

Mastrocinque, E., Ramírez, F. J., Honrubia-Escribano, A., & Pham, D. T. (2020). An AHP-based multi-criteria model for sustainable supply chain development in the renewable energy sector. *Expert Systems with Applications*, *150*, 113321. https://doi.org/10.1016/J.ESWA.2020.113321

Md Liton Hossain, Ahmed Abu-Siada, S. M. Muyeen, Md Mubashwar Hasan, & Md Momtazur Rahman. (2020). Industrial IoT based condition monitoring for wind energy conversion system. *CSEE Journal of Power and Energy Systems*, *7*(3), 654–664. https://doi.org/10.17775/CSEEJPES.2020.00680

Melo, J. J. R., Ishraque, Md. F., Shafiullah, G. M., & Shezan, Sk. A. (2023). Centralized monitoring of a cost efficient PLC-SCADA based islanded microgrid considering dispatch techniques. *The Journal of Engineering*, *2023*(8). https://doi.org/10.1049/tje2.12293

Mohanrasu, S. S., Janani, K., & Rakkiyappan, R. (2023). A COPRAS-based Approach to Multi-Label Feature Selection for Text Classification. *Mathematics and Computers in Simulation*. https://doi.org/10.1016/J.MATCOM.2023.07.022

Mudaliar, M. D., & Sivakumar, N. (2020). IoT based real time energy monitoring system using Raspberry Pi. *Internet of Things*, *12*, 100292. https://doi.org/10.1016/j.iot.2020.100292

Murugan, G., & Vijayarajan, S. (2023). IoT based secured data monitoring system for renewable energy fed micro grid system. *Sustainable Energy Technologies and Assessments*, *57*, 103244. https://doi.org/10.1016/j.seta.2023.103244

Mustapha, R., Mahmud, M., Paris, T. N. S. T., Musa, M. S., Awang, H., & Burhan, N. M. (2023). The Factors of the Drug Addict Relapse in Malaysia: The Experts Views and Consensus. *Open Journal of Social Sciences*, *11*(04), 160–173. https://doi.org/10.4236/jss.2023.114013

Nadimi-Shahraki, M. H., Taghian, S., & Mirjalili, S. (2021). An improved grey wolf optimizer for solving engineering problems. *Expert Systems with Applications*, *166*, 113917. https://doi.org/10.1016/j.eswa.2020.113917

Narendra, & Fukunaga. (1977). A Branch and Bound Algorithm for Feature Subset Selection. *IEEE Transactions on Computers*, *C–26*(9), 917–922. https://doi.org/10.1109/TC.1977.1674939

NextGIS. (2023). *NextGIS*. NextGIS. https://data.nextgis.com/en/region/MY-13/base

Noorollahi, Y., Ghenaatpisheh Senani, A., Fadaei, A., Simaee, M., & Moltames, R. (2022). A framework for GIS-based site selection and technical potential evaluation of PV solar farm using Fuzzy-Boolean logic and AHP multi-criteria decision-making approach. *Renewable Energy*, *186*, 89–104. https://doi.org/10.1016/J.RENENE.2021.12.124

Panwar, K., & Deep, K. (2021). Discrete Grey Wolf Optimizer for symmetric travelling salesman problem. *Applied Soft Computing*, *105*, 107298. https://doi.org/10.1016/J.ASOC.2021.107298

Parlangeli, G., De Palma, D., & Attanasi, R. (2024). A novel approach for 3PDP and real-time via point path planning of Dubins' vehicles in marine applications. *Control Engineering Practice*, *144*, 105814. https://doi.org/10.1016/J.CONENGPRAC.2023.105814

Patil, S. B., Patole, T. A., Jadhav, R. S., Suryawanshi, S. S., & Raykar, S. J. (2022). Complex Proportional Assessment (COPRAS) based Multiple-Criteria Decision Making (MCDM) paradigm for hard turning process parameters. *Materials Today: Proceedings*, *59*, 835–840. https://doi.org/10.1016/J.MATPR.2022.01.142

Perveen, G., Rizwan, M., & Goel, N. (2019). An ANFIS-based model for solar energy forecasting and its smart grid application. *Engineering Reports*, *1*(5), e12070. https://doi.org/10.1002/ENG2.12070

Ponce, P., Pérez, C., Fayek, A. R., & Molina, A. (2022). Solar Energy Implementation in Manufacturing Industry Using Multi-Criteria Decision-Making Fuzzy TOPSIS and S4 Framework. *Energies*, *15*(23), 8838. https://doi.org/10.3390/EN15238838

Pop, P. C., Cosma, O., Sabo, C., & Sitar, C. P. (2024). A comprehensive survey on the generalized traveling salesman problem. *European Journal of Operational Research*, *314*(3), 819–835. https://doi.org/10.1016/J.EJOR.2023.07.022

Pourpanah, F., Wang, R., Lim, C. P., Wang, X.-Z., & Yazdani, D. (2023). A review of artificial fish swarm algorithms: recent advances and applications. *Artificial Intelligence Review*, *56*(3), 1867–1903. https://doi.org/10.1007/s10462-022-10214-4

Prasanna Rani, D. D., Suresh, D., Rao Kapula, P., Mohammad Akram, C. H., Hemalatha, N., & Kumar Soni, P. (2023). IoT based smart solar energy monitoring systems. *Materials Today: Proceedings*, *80*, 3540–3545. https://doi.org/10.1016/j.matpr.2021.07.293

Ptaszynski, M., Lempa, P., Masui, F., Kimura, Y., Rzepka, R., Araki, K., Wroczynski, M., & Leliwa, G. (2019). "Brute-Force Sentence Pattern Extortion from Harmful Messages for Cyberbullying Detection." *Journal of the Association for Information Systems*, *20*(8), 1075–1127. https://doi.org/10.17705/1jais.00562

Qays, M. O., Ahmed, M. M., Parvez Mahmud, M. A., Abu-Siada, A., Muyeen, S. M., Hossain, M. L., Yasmin, F., & Rahman, M. M. (2022). Monitoring of renewable energy systems by IoT-aided SCADA system. *Energy Science & Engineering*, *10*(6), 1874–1885. https://doi.org/10.1002/ese3.1130

Rahkar Farshi, T., & Orujpour, M. (2021). A multi-modal bacterial foraging optimization algorithm. *Journal of Ambient Intelligence and Humanized Computing*, *12*(11), 10035–10049. https://doi.org/10.1007/s12652-020-02755-9

Rajakal, J. P., Saleem, N. N., Wan, Y. K., Ng, D. K. S., & Andiappan, V. (2023). Low-Carbon Energy Transition for the Sarawak Region via Multi-Period Carbon Emission Pinch Analysis. *Processes*, *11*(5), 1441. https://doi.org/10.3390/pr11051441

Rani, P., Mishra, A. R., Mardani, A., Cavallaro, F., Alrasheedi, M., & Alrashidi, A. (2020). A novel approach to extended fuzzy TOPSIS based on new divergence measures for renewable energy sources selection. *Journal of Cleaner Production*, *257*, 120352. https://doi.org/10.1016/J.JCLEPRO.2020.120352

Rao, C. K., Sahoo, S. K., & Yanine, F. F. (2024). A literature review on an IoT-based intelligent smart energy management systems for PV power generation. *Hybrid Advances*, *5*, 100136. https://doi.org/10.1016/j.hybadv.2023.100136

Razeghi, M., Hajinezhad, A., Naseri, A., Noorollahi, Y., & Farhan Moosavian, S. (2023). Multi-criteria decision-making for selecting a solar farm location to supply energy to reverse osmosis devices and produce freshwater using GIS in Iran. *Solar Energy*, *253*, 501–514. https://doi.org/10.1016/J.SOLENER.2023.01.029

Rios, R., & Duarte, S. (2021). Selection of ideal sites for the development of large-scale solar photovoltaic projects through Analytical Hierarchical Process – Geographic information systems (AHP-GIS) in Peru. *Renewable and Sustainable Energy Reviews*, *149*, 111310. https://doi.org/10.1016/J.RSER.2021.111310

Ruiz, H. S., Sunarso, A., Ibrahim-Bathis, K., Murti, S. A., & Budiarto, I. (2020). GIS-AHP Multi Criteria Decision Analysis for the optimal location of solar energy plants at Indonesia. *Energy Reports*, *6*, 3249–3263. https://doi.org/10.1016/J.EGYR.2020.11.198

S. Muhammad Hossein Mousavi. (2024). *TLBO TSP*. MATLAB Central File Exchange. https://github.com/SeyedMuhammadHosseinMousavi/TLBO-TSP

Saeidi, R., Noorollahi, Y., Aghaz, J., & Chang, S. (2023). FUZZY-TOPSIS method for defining optimal parameters and finding suitable sites for PV power plants. *Energy*, *282*, 128556. https://doi.org/10.1016/J.ENERGY.2023.128556

Saint-Drenan, Y.-M., Besseau, R., Jansen, M., Staffell, I., Troccoli, A., Dubus, L., Schmidt, J., Gruber, K., Simões, S. G., & Heier, S. (2020). A parametric model for wind turbine power curves incorporating environmental conditions. *Renewable Energy*, *157*, 754–768. https://doi.org/10.1016/j.renene.2020.04.123

Sałabun, W., & Urbaniak, K. (2020). A new coefficient of rankings similarity in decision-making problems. *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, *12138 LNCS*, 632–645. https://doi.org/10.1007/978-3-030-50417-5_47/FIGURES/6

Saraswat, S. K., Digalwar, A. K., Yadav, S. S., & Kumar, G. (2021). MCDM and GIS based modelling technique for assessment of solar and wind farm locations in India. *Renewable Energy*, *169*, 865–884. https://doi.org/10.1016/J.RENENE.2021.01.056

Sarawak Energy. (2021, July 12). *Investigation Confirms Murum Transmission Line Tripping Caused By Lightning - Defence System Kicked In Immediately Enabling Progressive Supply Restoration*. Sarawak Energy. https://www.sarawakenergy.com/media-info/media-releases/2021/investigation-confirms-murum-transmission-line-tripping-caused-by-lightning-defence-system-kicked-in-immediately-enabling-progressive-supply-restoration

Sasikumar, G., & Ayyappan, S. (2019). Multi-criteria Decision Making for Solar Panel Selection Using Fuzzy Analytical Hierarchy Process and Technique for Order Preference by Similarity to ideal Solution (TOPSIS): An Empirical Study. *Journal of The Institution of Engineers (India): Series C, 100*(4), 707–715. https://doi.org/10.1007/S40032-019-00520-2

Savira Kamarani, Imam Supriyadi, Yanif Dwi Kuntjoro, & Mochamad Fatchur Rozi. (2023). Implementation of the ASEAN Power Grid Interconnection West Kalimantan and Sarawak to Fulfill Electrical Energy Needs in Khatulistiwa. *International Journal Of Humanities Education and Social Sciences (IJHESS)*, *3*(3). https://doi.org/10.55227/ijhess.v3i3.810

Şengül, Ü., Eren, M., Eslamian Shiraz, S., Gezder, V., & Sengül, A. B. (2015). Fuzzy TOPSIS method for ranking renewable energy supply systems in Turkey. *Renewable Energy*, *75*, 617–625. https://doi.org/10.1016/J.RENENE.2014.10.045

Seyedali Mirjalili. (2024). Grey Wolf Optimizer (GWO) . *MATLAB Central File Exchange*. https://www.mathworks.com/matlabcentral/fileexchange/44974-grey-wolf-optimizer-gwo

Seyyedabbasi, A., & Kiani, F. (2021). I-GWO and Ex-GWO: improved algorithms of the Grey Wolf Optimizer to solve global optimization problems. *Engineering with Computers*, *37*(1), 509–532. https://doi.org/10.1007/S00366-019-00837-7/METRICS

Shahrom, M. A. S. M., .. N., Aziz, M. F. Ab., & Mostafa, S. A. (2023). A Review of Glowworm Swarm Optimization Meta-Heuristic Swarm Intelligence and its Fusion in Various Applications. *Fusion: Practice and Applications, 13*(1), 89–102. https://doi.org/10.54216/FPA.130107

Shahrom, S. F., Aviso, K. B., Tan, R. R., Saleem, N. N., Ng, D. K. S., & Andiappan, V. (2023). Regional Planning and Optimization of Renewable Energy Sources for Improved Rural Electrification. *Process Integration and Optimization for Sustainability*, *7*(4), 785–804. https://doi.org/10.1007/s41660-023-00323-0

Shakil, M., Rashid, Z., Hussain, G. A., & Umer, F. (2020). Power Flow Analysis and Optimization in Ring Distribution Network of Bahawalpur using Newton Raphson Method. *Indian Journal of Science and Technology*, *13*(27), 2720–2732. https://doi.org/10.17485/IJST/V13I27.916

Shami, T. M., El-Saleh, A. A., Alswaitti, M., Al-Tashi, Q., Summakieh, M. A., & Mirjalili, S. (2022). Particle Swarm Optimization: A Comprehensive Survey. *IEEE Access*, *10*, 10031–10061. https://doi.org/10.1109/ACCESS.2022.3142859

Shao, M., Han, Z., Sun, J., Xiao, C., Zhang, S., & Zhao, Y. (2020). A review of multi-criteria decision making applications for renewable energy site selection. *Renewable Energy*, *157*, 377–403. https://doi.org/10.1016/J.RENENE.2020.04.137

Shorabeh, S. N., Firozjaei, H. K., Firozjaei, M. K., Jelokhani-Niaraki, M., Homaee, M., & Nematollahi, O. (2022). The site selection of wind energy power plant using GIS-multi-criteria evaluation from economic perspectives. *Renewable and Sustainable Energy Reviews*, *168*, 112778. https://doi.org/10.1016/J.RSER.2022.112778

Shorabeh, S. N., Samany, N. N., Minaei, F., Firozjaei, H. K., Homaee, M., & Boloorani, A. D. (2022). A decision model based on decision tree and particle swarm optimization algorithms to identify optimal locations for solar power plants construction in Iran. *Renewable Energy*, *187*, 56–67. https://doi.org/10.1016/J.RENENE.2022.01.011

Simonović, P., Ristić, R., Milčanović, V., Polovina, S., Malušević, I., Radić, B., Kanjuh, T., Marić, A., & Nikolić, V. (2021). Effects of run-of-river hydropower plants on fish communities in montane stream ecosystems in Serbia. *River Research and Applications*, *37*(5), 722–731. https://doi.org/10.1002/RRA.3795

Skinderowicz, R. (2022). Improving Ant Colony Optimization efficiency for solving large TSP instances. *Applied Soft Computing*, *120*, 108653. https://doi.org/10.1016/j.asoc.2022.108653

Slowik, A., & Kwasnicka, H. (2020). Evolutionary algorithms and their applications to engineering problems. *Neural Computing and Applications*, *32*(16), 12363–12379. https://doi.org/10.1007/s00521-020-04832-8

Solangi, Y. A., Longsheng, C., & Shah, S. A. A. (2021). Assessing and overcoming the renewable energy barriers for sustainable development in Pakistan: An integrated AHP and fuzzy TOPSIS approach. *Renewable Energy*, *173*, 209–222. https://doi.org/10.1016/J.RENENE.2021.03.141

Solargis. (2023, May 7). *Solargis*. https://solargis.com/maps-and-gis-data/overview

Solcast. (2024). Solar Api and Solar Weather Forecasting Tool. *Solcast*. https://solcast.com

Stefano, N. M., Casarotto Filho, N., Garcia Lupi Vergara, L., & Garbin Da Rocha, R. U. (2015). COPRAS (Complex Proportional Assessment): State of the art research and its applications. *IEEE Latin America Transactions*, *13*(12), 3899–3906. https://doi.org/10.1109/TLA.2015.7404925

Tang, A. S. O., Leong, T. S., Ong, J. H. L., Goh, A., & Chew, L. P. (2023). Cost of Treatment for Myelofibrosis Patients in Ministry of Health Hospitals in Sarawak, Malaysia. *Asian*

Pacific Journal of Cancer Prevention, *24*(3), 733–736. https://doi.org/10.31557/APJCP.2023.24.3.733

Tawanda, T., Nyamugure, P., Kumar, S., & Munapo, E. (2023). A Labelling Method for the Travelling Salesman Problem. *Applied Sciences, 13*(11), 6417. https://doi.org/10.3390/APP13116417

Temel, P., Kentel, E., & Alp, E. (2023). Development of a site selection methodology for run-of-river hydroelectric power plants within the water-energy-ecosystem nexus. *Science of The Total Environment*, *856,* 159152. https://doi.org/10.1016/J.SCITOTENV.2022.159152

Tercan, E., Eymen, A., Urfalı, T., & Saracoglu, B. O. (2021). A sustainable framework for spatial planning of photovoltaic solar farms using GIS and multi-criteria assessment approach in Central Anatolia, Turkey. *Land Use Policy*, *102*, 105272. https://doi.org/10.1016/J.LANDUSEPOL.2020.105272

Ullah, Z., Elkadeem, M. R., Kotb, K. M., Taha, I. B. M., & Wang, S. (2021). Multi-criteria decision-making model for optimal planning of on/off grid hybrid solar, wind, hydro, biomass clean electricity supply. *Renewable Energy*, *179*, 885–910. https://doi.org/10.1016/J.RENENE.2021.07.063

Ullah, Z., Rehman, A. U., Wang, S., Hasanien, H. M., Luo, P., Elkadeem, M. R., & Abido, M. A. (2023). IoT-based monitoring and control of substations and smart grids with renewables and electric vehicles integration. *Energy*, *282*, 128924. https://doi.org/10.1016/j.energy.2023.128924

Ullah, Z., Wang, S., Wu, G., Xiao, M., Lai, J., & Elkadeem, M. R. (2022). Advanced energy management strategy for microgrid using real-time monitoring interface. *Journal of Energy Storage*, *52*, 104814. https://doi.org/10.1016/j.est.2022.104814

Urošević, B. G., & Marinović, B. (2021). Ranking construction of small hydro power plants using multi-criteria decision analysis. *Renewable Energy*, *172*, 1174–1183. https://doi.org/10.1016/J.RENENE.2021.03.115

Vafaei, N., Ribeiro, R. A., & Camarinha-Matos, L. M. (2022). Assessing Normalization Techniques for Simple Additive Weighting Method. *Procedia Computer Science*, *199*, 1229–1236. https://doi.org/10.1016/J.PROCS.2022.01.156

Wang, C. N., Dang, T. T., & Nguyen, N. A. T. (2021). Location Optimization of Wind Plants Using DEA and Fuzzy Multi-Criteria Decision Making: A Case Study in Vietnam. *IEEE Access*, *9*, 116265–116285. https://doi.org/10.1109/ACCESS.2021.3106281

Wang, C. N., Dang, T. T., Nguyen, N. A. T., & Wang, J. W. (2022). A combined Data Envelopment Analysis (DEA) and Grey Based Multiple Criteria Decision Making (G-MCDM) for solar PV power plants site selection: A case study in Vietnam. *Energy Reports*, *8*, 1124–1142. https://doi.org/10.1016/J.EGYR.2021.12.045

Wang, D., Gao, N., Liu, D., Li, J., & Lewis, F. L. (2024). Recent Progress in Reinforcement Learning and Adaptive Dynamic Programming for Advanced Control Applications. *IEEE/CAA Journal of Automatica Sinica*, *11*(1), 18–36. https://doi.org/10.1109/JAS.2023.123843

Wang, S., Liu, M., & Chu, F. (2020). Approximate and exact algorithms for an energy minimization traveling salesman problem. *Journal of Cleaner Production*, *249*, 119433. https://doi.org/10.1016/J.JCLEPRO.2019.119433

Wang, Y., Xue, P., Cao, M., Yu, T., Lane, S. T., & Zhao, H. (2021). Directed Evolution: Methodologies and Applications. *Chemical Reviews*, *121*(20), 12384–12444. https://doi.org/10.1021/acs.chemrev.1c00260

Wang, Z., & Sobey, A. (2020). A comparative review between Genetic Algorithm use in composite optimisation and the state-of-the-art in evolutionary computation. *Composite Structures*, *233*, 111739. https://doi.org/10.1016/j.compstruct.2019.111739

Wei, Q. (2021). Sustainability evaluation of photovoltaic poverty alleviation projects using an integrated MCDM method: A case study in Guangxi, China. *Journal of Cleaner Production*, *302*, 127040. https://doi.org/10.1016/J.JCLEPRO.2021.127040

Wolfshohl, A. K., & Sweers, C. M. (2021). Engineering, Procurement, and Construction Contracts in Texas: Key Provisions, Issues and Pitfalls. *Construction Law Journal*, *17*, 6.

Wu, J., Wang, Y.-G., Burrage, K., Tian, Y.-C., Lawson, B., & Ding, Z. (2020). An improved firefly algorithm for global continuous optimization problems. *Expert Systems with Applications*, *149*, 113340. https://doi.org/10.1016/j.eswa.2020.113340

Wu, Y., Deng, Z., Tao, Y., Wang, L., Liu, F., & Zhou, J. (2021). Site selection decision framework for photovoltaic hydrogen production project using BWM-CRITIC-MABAC: A case study in Zhangjiakou. *Journal of Cleaner Production*, *324*, 129233. https://doi.org/10.1016/J.JCLEPRO.2021.129233

Wu, Y., Zhang, B., Wu, C., Zhang, T., & Liu, F. (2019). Optimal site selection for parabolic trough concentrating solar power plant using extended PROMETHEE method: A case in China. *Renewable Energy*, *143*, 1910–1927. https://doi.org/10.1016/J.RENENE.2019.05.131

Xiong, H., Egusquiza, M., Alberg Østergaard, P., Pérez-Díaz, J. I., Sun, G., Egusquiza, E., Patelli, E., Xu, B., Duan, H., Chen, D., & Luo, X. (2021). Multi-objective optimization of a hydro-wind-photovoltaic power complementary plant with a vibration avoidance strategy. *Applied Energy*, *301*, 117459. https://doi.org/10.1016/J.APENERGY.2021.117459

Yalçin, N., & Pehlivan, N. Y. (2019). Application of the Fuzzy CODAS Method Based on Fuzzy Envelopes for Hesitant Fuzzy Linguistic Term Sets: A Case Study on a Personnel Selection Problem. *Symmetry, 11*(4), 493. https://doi.org/10.3390/SYM11040493

Yarpiz / Mostapha Heris. (2024a). *Ant Colony Optimization (ACO)* . MATLAB Central File Exchange. https://www.mathworks.com/matlabcentral/fileexchange/52859-ant-colony-optimization-aco

Yarpiz / Mostapha Heris. (2024b). *Tabu Search (TS)* . MATLAB Central File Exchange. https://www.mathworks.com/matlabcentral/fileexchange/52902-tabu-search-ts

Yazdani, M., & Jolai, F. (2016). Lion Optimization Algorithm (LOA): A nature-inspired metaheuristic algorithm. *Journal of Computational Design and Engineering*, *3*(1), 24–36. https://doi.org/10.1016/j.jcde.2015.06.003

Yu, Y., Wu, S., Yu, J., Chen, H., Zeng, Q., Xu, Y., & Ding, H. (2022). An integrated MCDM framework based on interval 2-tuple linguistic: A case of offshore wind farm site selection in China. *Process Safety and Environmental Protection*, *164*, 613–628. https://doi.org/10.1016/J.PSEP.2022.06.041

Zafar, S., Alamgir, Z., & Rehman, M. H. (2021). An effective blockchain evaluation system based on entropy-CRITIC weight method and MCDM techniques. *Peer-to-Peer*

Networking and Applications, *14*(5), 3110–3123. https://doi.org/10.1007/S12083-021-01173-8/METRICS

Zahid, F., Tahir, A., Khan, H. U., & Naeem, M. A. (2021). Wind farms selection using geospatial technologies and energy generation capacity in Gwadar. *Energy Reports, 7*, 5857–5870. https://doi.org/10.1016/J.EGYR.2021.08.165

Zambrano-Asanza, S., Quiros-Tortos, J., & Franco, J. F. (2021). Optimal site selection for photovoltaic power plants using a GIS-based multi-criteria decision making and spatial overlay with electric load. *Renewable and Sustainable Energy Reviews*, *143*, 110853. https://doi.org/10.1016/J.RSER.2021.110853

Zedan Shaban, R., & Natheer Alkallak, I. (2021). Organizing sports matches with a hybrid monkey search algorithm. *Indonesian Journal of Electrical Engineering and Computer Science*, *22*(1), 542. https://doi.org/10.11591/ijeecs.v22.i1.pp542-551

Zhang, F., Mei, Y., Nguyen, S., & Zhang, M. (2021). Evolving Scheduling Heuristics via Genetic Programming With Feature Selection in Dynamic Flexible Job-Shop Scheduling. *IEEE Transactions on Cybernetics*, *51*(4), 1797–1811. https://doi.org/10.1109/TCYB.2020.3024849

Zhao, L., Wang, F., & Bai, Y. (2023). Route planning for autonomous vessels based on improved artificial fish swarm algorithm. *Ships and Offshore Structures*, *18*(6), 897–906. https://doi.org/10.1080/17445302.2022.2081423

Zhou, J., Wu, Y., Wu, C., He, F., Zhang, B., & Liu, F. (2020). A geographical information system based multi-criteria decision-making approach for location analysis and evaluation of urban photovoltaic charging station: A case study in Beijing. *Energy*

Conversion and Management, *205*, 112340. https://doi.org/10.1016/J.ENCONMAN.2019.112340

APPENDICES

Appendix A: Journal Publications

1. **Jong, F. C.,** & Ahmed, M. M. (2024). Multi-Criteria Decision-Making Solutions for Optimal Solar Energy Sites Identification: A Systematic Review and Analysis. IEEE Access, 12, 143458.
2. **Jong, F. C.,** & Ahmed, M. M. (2024). Novel GIS-based fuzzy TOPSIS and filtration algorithms for extra-large scale optimal solar energy sites identification. Solar Energy, 268, 112274.
3. **Jong, F. C.,** & Ahmed, M. M. (2024). Power line routing design by GIS-driven fuzzy traveling salesman problem-binary integer programming for green energy integration. Applied Energy, 374, 124092.
4. **Jong, F. C.,** Ahmed, M. M., Lau, W. K., & Sayed, M. A. (2025). Modeling novel hybrid green energy systems with IIoT-based real-time dynamic monitoring, control and automation. Computers and Electrical Engineering, 123, 110141.

Appendix B: Green Energy Locations Identification

```
import arcpy
# Define the paths
points_layer = (r"path\point_layername.shp")
polygon_layers = [
  (r"path\polygon_layername.shp", "inside"),
  (r"path\polygon_layername.shp", "outside"),
                    …]
# Create a temporary feature layer for points
temp_points_layer = "temp_points_layer"
arcpy.MakeFeatureLayer_management(points_layer, temp_points_layer)
# Iterate through the list of polygon layers and conditions
for polygon_path, condition in polygon_layers:
    if condition == "inside":
        arcpy.management.SelectLayerByLocation(temp_points_layer, "INTERSECT",
polygon_path)
    elif condition == "outside":
        arcpy.management.SelectLayerByLocation(temp_points_layer, "INTERSECT",
polygon_path, invert_spatial_relationship=True)
    # Remove the selected points
    arcpy.DeleteFeatures_management(temp_points_layer)
# Delete the temporary feature layer
arcpy.Delete_management(temp_points_layer)
print("Erase operation completed for all polygon layers and conditions.")
```

Figure B-1: First Phase of Green Energy Locations Filtration

```
import arcpy
# Input point layer
points_layer = (r"path\point_layername.shp")
# List of raster datasets to extract values from
raster_layers = [
  (r"path\raster_layername.tif"),
  (r"path\raster_layername.vrt"),
                  …]
field_names = []
# Iterate through the raster layers and add them as separate fields
for i, raster_layer in enumerate(raster_layers):
    field_name = arcpy.ValidateFieldName(f"Value_{i + 1}", arcpy.env.workspace)
    field_names.append(field_name)
    arcpy.sa.ExtractMultiValuesToPoints(points_layer, [[raster_layer, field_name]])
# Print the field names added to the point layer
print("Field names added to the point layer:", field_names)
print("Extraction complete.")
```

Figure B-2: Extraction of Multi-Values to Points

```
import arcpy
input_point_fc = (r"path\point_layername.shp")
# Define the SQL expressions for each criterion
criteria = [
   "Value_1 >= 1200",
   "Value_2 >= 15",
   "Value_2 < 28",
   "Value_3 < 25",
   "Value_4 >= 0",
   "Value_4 < 2200",
   "Value_5 >= 0.01",
   "Value_5 < 50",
   "Value_6 >= 0.1",
   "Value_6 < 50",
   "Value_7 >= 0.3",
   "Value_7 < 45",
   "Value_8 >= 0.3",
   "Value_8 < 45",
   "Value_9 >= 0.1",
   "Value_9 < 20",
   "Value_10 >= 0.1",
   "Value_11 >= 0.1",
  "Value_12 > 0" ]
# Create a list of criteria expressions
expression_list = []
for criterion in criteria:
   expression_list.append(criterion)
# Combine all criteria with AND operator
expression = "AND".join(expression_list)
# Select points that meet the criteria
arcpy.MakeFeatureLayer_management(input_point_fc, "SelectedPointsLayer")
arcpy.SelectLayerByAttribute_management("SelectedPointsLayer",
"NEW_SELECTION", expression)
# Create a new feature class with selected points
output_feature_class = "SelectedPoints.shp"
arcpy.CopyFeatures_management("SelectedPointsLayer", output_feature_class)
# Clear the selection
arcpy.SelectLayerByAttribute_management("SelectedPointsLayer",
"CLEAR_SELECTION")
# Print a message indicating the number of selected points
count = arcpy.GetCount_management(output_feature_class)
print(f"Number of selected points: {count}")
# Clean up by deleting the temporary layer
arcpy.Delete_management("SelectedPointsLayer")
```

Figure B-3: Second Phase of Green Energy Locations Filtration

```
% Load data from Excel file
alternatives = xlsread('path\filename.xlsx');
% Define the criteria and linguistic terms
criteria = {
   [1200, 1400], [1400, 1600], [1600, 1700], [1700, 1800], [1800, 1900];
   [27, 28], [26, 27], [25, 26], [24, 25], [15, 24];
   [15, 25], [10, 15], [5, 10], [2, 5], [0, 2];
   [0, 200], [200, 450], [450, 750], [750, 1200], [1200, 2200];
   [20, 50], [15, 20], [10, 15], [5, 10], [0.01, 5];
   [30, 50], [20, 30], [10, 20], [5, 10], [0.1, 5];
   [30, 45], [20, 30], [15, 20], [10, 15], [0.3, 10];
   [30, 45], [20, 30], [15, 20], [10, 15], [0.3, 10];
   [16, 20], [12, 16], [8, 12], [4, 8], [0.1, 4];
   [0.1, 1], [1, 2], [2, 3], [3, 4], [4, 300];
   [30, 300], [20, 30], [10, 20], [5, 10], [0.1, 5];
   [0, 100], [100, 200], [200, 300], [300, 400], [400, 500]; };
% Define criteria attributes
Attributes = {'benefit', 'cost', 'cost', 'benefit', 'cost', 'cost', 'cost', 'cost', 'cost',
'benefit', 'cost', 'benefit'};
% Define membership functions
param_keys = {'VL', 'L', 'M', 'H', 'VH'};
param_values = {[1, 1, 3], [1, 3, 5], [3, 5, 7], [5, 7, 9], [7, 9, 9]};
params = containers.Map(param_keys, param_values);
% Define fuzzy criteria weight
FCW = {
   [8.43 12.12 17.65],
   [6.87 10.85 16.06],
   [6.57 9.88  14.85],
   [5.56 10.48 15.61],
   [7.02 10.42 15.53],
   [5.81 9.58  14.47],
   [5.76 9.21  14.02],
   [5.76 9.21  14.02],
   [5.76 9.21  14.02],
   [7.27 11.03 16.29],
   [5.76 9.21  14.02],
   [6.21 9.76  14.70]};
```

Figure B-4: Input Definition

```
% Define linguistic terms
linguistic_terms = param_keys;
% Loop through each alternatives point and assign linguistic terms based on criteria
for i = 1:size(alternatives, 1)
   for j = 1:size(alternatives, 2)
      value = alternatives(i, j);
      found = false;
```

```
        for k = 1:size(criteria, 2)
            range = criteria{j, k};
            if value > range(1) && value <= range(2)
                % Assign the appropriate linguistic term
                B{i, j} = linguistic_terms{k};
                found = true;
                break;
            end
        end
        if ~found
 B{i, j} = 'Invalid';
        end
    end
end
B = string(B);
B_num = NaN(size(B));
B_cell = cell(size(B));
% Convert the string matrix alternatives to numerical values and cell arrays of vectors
for i = 1:size(B, 1)
    for j = 1:size(B, 2)
        term = B{i, j};
        if isKey(params, term)
            param_value = params(term);
            B_num(i, j) = param_value(2);
            B_cell{i, j} = param_value;
        else error('Invalid value in B.');
        end
    end
end
```

Figure B-5: Matrix Values to Fuzzy Triangular Membership Values

```
% Get the number of conditions
numConditions = numel(Attributes);
% Initialize arrays to store maximum values and minimum values
maxValues = zeros(1, numConditions);
minValues = zeros(1, numConditions);
% Calculate maximum and minimum values for each condition
for i = 1:numConditions
    if strcmp(Attributes{i}, 'benefit')
        maxValues(i) = max(cellfun(@(x) max(x), B_cell(:, i)));
    elseif strcmp(Attributes{i}, 'cost')
        minValues(i) = min(cellfun(@(x) min(x), B_cell(:, i)));
    end
end
% Normalize fuzzy decision matrix
for i = 1:size(B_cell, 2) % Iterate through columns
```

```
    if strcmp(Attributes{i}, 'benefit')
        C_cell(:, i) = cellfun(@(x) x / maxValues(i), B_cell(:, i), 'UniformOutput', false);
    elseif strcmp(Attributes{i}, 'cost')
        C_cell(:, i) = cellfun(@(x) x / minValues(i), B_cell(:, i), 'UniformOutput', false);
    end
end
```

Figure B-6: Fuzzy Decision Matrix Normalization

```
% Initialize D_cell with zeros
numRows = size(C_cell, 1);
numColsC = size(C_cell{1, 1}, 2);
numColsFCW = numel(FCW);
D_cell = cell(numRows, numColsFCW);
% Weight Normalization
for i = 1:numRows
    for j = 1:numColsFCW
        D_cell{i, j} = zeros(size(C_cell{i, 1}));
        for k = 1:numColsC
            D_cell{i, j}(k) = C_cell{i, j}(k) * FCW{j}(k);
        end
    end
end
```

Figure B-7: Fuzzy Decision Matrix Weight Normalization

```
numColsD_cell = size(D_cell, 2);
for col = 1:numColsD_cell
    max_3rdEC = max(cellfun(@(x) x(3), D_cell(:, col)));
    Smax_3rdEC = D_cell(cellfun(@(x) x(3) == max_3rdEC, D_cell(:, col)), col);
    max_2ndEC = max(cellfun(@(x) x(2), Smax_3rdEC));
    Smax_2ndEC = Smax_3rdEC(cellfun(@(x) x(2) == max_2ndEC, Smax_3rdEC), 1);
    max_1stEC = max(cellfun(@(x) x(1), Smax_2ndEC));
    max_1stEC_idx = find(cellfun(@(x) x(1) == max_1stEC, Smax_2ndEC), 1);
    FPIS{col} = Smax_2ndEC{max_1stEC_idx};
end
for col = 1:numColsD_cell
    min_1stEC = min(cellfun(@(x) x(1), D_cell(:, col)));
    Smin_1stEC = D_cell(cellfun(@(x) x(1) == min_1stEC, D_cell(:, col)), col);
    min_2ndEC = min(cellfun(@(x) x(2), Smin_1stEC));
    Smin_2ndEC = Smin_1stEC(cellfun(@(x) x(2) == min_2ndEC, Smin_1stEC), 1);
    min_3rdEC = min(cellfun(@(x) x(3), Smin_2ndEC));
    min_3rdEC_idx = find(cellfun(@(x) x(3) == min_3rdEC, Smin_2ndEC), 1);
    FNIS{col} = Smin_2ndEC{min_3rdEC_idx};
end
```

Figure B-8: FPIS and PNIS Computation

```
% Distance FPIS and FNIS
for i = 1:size(alternatives, 1)
   for j = 1:numColsD_cell
      D_FPIS{i, j} = sqrt(1/3 * sum((D_cell{i, j} - FPIS{j}).^2));
   end
end
for i = 1:size(alternatives, 1)
   for j = 1:numColsD_cell
      D_FNIS{i, j} = sqrt(1/3 * sum((D_cell{i, j} - FNIS{j}).^2));
   end
end
% Cell to Vector
C_V1 = cell2mat(num2cell([D_FPIS{:}]));
Dvec_FPIS = reshape(C_V1, size(D_FPIS));
C_V2 = cell2mat(num2cell([D_FNIS{:}]));
Dvec_FNIS = reshape(C_V2, size(D_FNIS));
% Distance Summation
SumDis_FPIS = sum(Dvec_FPIS, 2);
SumDis_FNIS = sum(Dvec_FNIS, 2);
```

Figure B-9: Euclidean Distance for FPIS and PNIS

```
% Closeness Coefficient
CC = zeros(size(SumDis_FPIS, 1), size(SumDis_FPIS, 2));
for i = 1:size(SumDis_FPIS, 1)
   CC(i) = SumDis_FNIS(i) / (SumDis_FNIS(i) + SumDis_FPIS(i));
end
% Rank alternatives based on the values and assign ranks to the original positions
[~, rank] = sort(CC, ‘descend’);
[~, idx] = sort(rank);
Ranking = idx;
```

Figure B-10: Closeness Coefficient and Rankings

Table B-1: Identified Green Energy Locations

ID	Latitude	Longitude	ID	Latitude	Longitude	ID	Latitude	Longitude	ID	Latitude	Longitude
S1	1.3666	110.9666	S67	2.1	111.3666	H10	1.752141	112.545624	H76	3.394214	114.175792
S2	1.9666	111.5	S68	1.9333	112.0833	H11	1.89833	112.571149	H77	4.838551	115.448238
S3	1.4166	110.9666	S69	1.8666	111.6333	H12	1.875125	112.605956	H78	4.805066	115.508511
S4	1.2	110.2833	S70	2.6333	111.7	H13	1.791589	112.650045	H79	4.463515	115.376802
S5	2.0333	112.4833	S71	2.0666	111.3166	H14	1.805512	112.677891	H80	4.483606	115.361176
S6	1.4333	110.85	S72	1.75	111.2333	H15	1.69645	112.708056	H81	4.644336	115.318761
S7	1.2833	110.7	S73	1.9	112.3166	H16	1.691809	112.733581	H82	4.597456	115.32769
S8	2.0333	111.35	S74	3.4833	114.0166	H17	1.652361	112.770709	H83	4.568436	115.320993
S9	1.2666	110.0833	S75	1.6666	112.3166	H18	1.905291	112.605956	H84	4.550577	115.320993
S10	1.3	110.1166	S76	3.45	113.4333	H19	2.194177	113.056317	H85	4.497	115.287508
S11	4.1666	114.3666	S77	1.3833	110.9666	H20	2.178827	113.061433	H86	4.483606	115.332155
S12	1.45	110.95	S78	1.3333	110.5	H21	2.025329	113.03585	H87	4.443423	115.396893
S13	1.7	112.3666	S79	2.25	112.2	H22	2.033004	113.145857	H88	4.41217	115.401358
S14	1.1833	111.1833	S80	1.2833	110.6666	H23	2.048353	113.189349	H89	4.371988	115.446005
S15	1.5166	110.85	S81	1.55	109.9833	H24	2.020212	113.189349	H90	4.135358	115.521906
S16	2.1333	112.45	S82	2.3166	111.5666	H25	1.764381	113.181674	H91	4.171075	115.233931
S17	1.7666	112.55	S83	1.7	112.4833	H26	1.994629	113.222606	H92	4.206793	115.169193
S18	2.0166	112.5833	S84	1.4666	110.8	H27	1.935788	113.284006	H93	4.240279	115.086595
S19	2.5166	111.7	S85	1.55	110.7166	H28	2.076495	113.342847	H94	4.264835	115.068736
S20	2.05	112.1333	S86	1.8833	112.5333	H29	1.940905	113.355638	H95	4.340735	114.995069
S21	1.3333	110.9	S87	2.6166	112.0333	H30	1.792523	113.33773	H96	4.217955	115.106687
S22	1.2333	110.8	S88	2.2	111.2666	H31	1.843689	113.409363	H97	4.197864	115.13794
S23	2.4	111.5166	S89	1.7666	112.3	H32	1.513668	113.455412	H98	4.164378	115.077666
S24	1.9	112.0666	S90	1.3166	110.9666	H33	1.56995	113.470762	H99	4.13759	115.240628
S25	1.7166	112.4	S91	2.1	112.4333	H34	1.608325	113.483554	H100	4.126428	115.265184
S26	3.1	113.1333	S92	2.6	112.05	H35	1.884622	113.465646	H101	4.09964	115.274114
S27	1.3	111.1833	S93	1.2666	111.2666	H36	1.85648	113.56542	H102	3.996951	115.294205
S28	2.3333	111.5833	S94	1.3	111.0666	H37	1.88718	113.849391	H103	3.952304	115.122313
S29	2.3	111.3166	S95	1.9166	112.3333	H38	1.779731	113.7752	H104	3.925516	115.350014
S30	1.3333	110.9833	S96	1.3166	110.6833	H39	1.697866	113.721476	H105	3.186605	115.280811
S31	2.1	111.4	S97	1.95	112.3666	H40	1.580183	113.683102	H106	3.229019	115.32769
S32	1.9333	112.3666	S98	1.2333	111.1833	H41	1.87183	113.859625	H107	3.340637	115.401358
S33	1.35	110.85	S99	1.2333	111.2	H42	3.168891	113.990098	H108	3.354032	115.423682

Table B-1 continued

S34	1.3	110.8166	S100	1.5	109.9	H43	3.120283	113.946607	H109	3.438861	115.45047
S35	1.3833	111.0833	W1	2.2333	113.7166	H44	2.872128	114.158946	H110	3.476811	115.48842
S36	1.2	110.8666	W2	2.1166	113.6833	H45	2.805612	114.130805	H111	3.032572	115.097757
S37	1.2166	111.2166	W3	1.1333	110.6	H46	2.731421	114.133363	H112	3.148654	115.070969
S38	1.8166	112.2666	W4	1.4	110.25	H47	2.733979	114.097547	H113	3.068289	115.037483
S39	1.65	110.0833	W5	0.9666	110.7	H48	2.79282	113.982423	H114	3.041501	115.01516
S40	1.8833	112.55	W6	1.4666	111.9666	H49	2.841428	113.8673	H115	3.5795	115.408055
S41	1.3833	110.9833	W7	1.1833	110.55	H50	2.815845	113.744501	H116	3.577268	115.238396
S42	2.2333	111.4166	W8	1.3833	111.6666	H51	2.749329	113.803342	H117	3.639774	115.209375
S43	1.35	110.25	W9	1.5166	110.9333	H52	2.731421	113.591003	H118	2.853983	114.921401
S44	1.5833	111.7	W10	1.7333	111.6333	H53	2.662347	114.054056	H119	2.842821	114.881218
S45	3.8666	114.2666	W11	1.9833	114.45	H54	2.629089	114.056614	H120	2.661184	114.788175
S46	1.3166	111.0666	W12	2.0666	113.6666	H55	2.380722	113.827544	H121	2.516467	114.846751
S47	1.6333	111.4666	W13	2.2833	113.9	H56	2.358399	113.845403	H122	2.661184	114.629675
S48	1.35	110.2666	W14	4.15	113.8833	H57	2.286963	113.878888	H123	2.654293	114.323012
S49	1.5166	109.9833	W15	1.8	109.6833	H58	2.159719	113.941394	H124	2.606054	114.436719
S50	2.0166	112.6166	W16	3.6666	114.1	H59	2.496805	112.905579	H125	2.399315	114.729599
S51	1.3333	110.5166	W17	1.1666	110.3833	H60	3.896495	114.963815	H126	2.554369	114.185186
S52	2.0166	112.5666	W18	1.2166	110.25	H61	3.833989	114.782994	H127	2.471673	114.247208
S53	1.8166	112.65	W19	1.1166	110.3	H62	3.798271	114.979442	H128	2.40276	114.471175
S54	3.6833	114.2666	W20	1.2166	110.0833	H63	3.45672	114.961583	H129	2.313173	114.161067
S55	4.15	114.35	W21	1.15	110.6166	H64	3.385285	114.923633	H130	2.264934	114.219643
S56	1.55	110.85	W22	1.4833	110.8333	H65	3.338405	114.7696	H131	2.192576	114.426382
S57	1.45	110.9666	W23	4.4	115.3833	H66	3.441094	114.561991	H132	2.123663	114.102491
S58	1.3833	110.9	H1	1.805512	111.926062	H67	3.37189	114.57092	H133	2.03063	114.081817
S59	1.1833	110.4	H2	1.601311	112.211478	H68	3.273667	114.81648	H134	2.009956	114.212752
S60	2.3333	111.3333	H3	1.622195	112.230042	H69	3.170978	114.818712	H135	1.916924	114.540088
S61	2.6333	112.05	H4	1.643079	112.250926	H70	3.081684	114.812015	H136	1.848011	114.491849
S62	1.2833	111.1833	H5	1.6454	112.299656	H71	3.186605	114.575385	H137	1.799772	114.54698
S63	1.55	110.4666	H6	1.933137	112.225401	H72	2.996854	114.88345	H138	1.879022	114.312675
S64	2.2666	112.2833	H7	1.912253	112.341424	H73	3.016945	114.827641	-	-	-
S65	1.9666	112.05	H8	1.687168	112.415679	H74	2.985692	114.863359	-	-	-
S66	1.25	110.8833	H9	1.735898	112.529381	H75	3.414305	114.160166	-	-	-

Appendix C: Green Energy Locations Integration

```
import arcpy
import csv
# Constants for converting decimal degrees to kilometres
DD_TO_KM_FACTOR = 111
# Set the shapefile name and the output CSV file path
shapefile_name = “shapefile_name_C_i”
output_csv = r“path\DistanceMatrixData_C_i.csv”
distances = []
# Find the shapefile in the content panel
project = arcpy.mp.ArcGISProject(“CURRENT”)
map = project.listMaps()[0]
# Retrieve the fields (column names) from the shapefile
fields = [field.name for field in arcpy.ListFields(shapefile_name)]
# Ensure that the necessary fields exist in the shapefile
if ‘Latitude’ in fields and ‘Longitude’ in fields:
    # Open the shapefile
    with arcpy.da.SearchCursor(shapefile_name, [“ID”, “Latitude”, “Longitude”]) as
cursor:
        points = [(id, lat, lon) for id, lat, lon in cursor]
    # Calculate pairwise distances in kilometres and fill the matrix
    num_points = len(points)
    for i in range(num_points):
        row = [0.0] * num_points
        id1, lat1, lon1 = points[i]
        for j in range(num_points):
            if i != j:
                id2, lat2, lon2 = points[j]
                distance = math.sqrt((lat1 - lat2) ** 2 + (lon1 - lon2) ** 2) *
DD_TO_KM_FACTOR
                row[j] = distance
        distances.append(row)
    # Write the pairwise distance matrix to a CSV file without labels for the first row
    with open(output_csv, ‘w’, newline=”) as csvfile:
        csv_writer = csv.writer(csvfile)
        for row in distances:
            csv_writer.writerow(row)  # Write rows without labels
    print(“Pairwise distance matrix (in kilometers) has been calculated and saved to”,
output_csv)
else: print(“The ‘Latitude’ and ‘Longitude’ fields are missing in the shapefile.”)
```

Figure C-1: Distance Matrix Data Generation for Each Cluster

```
import arcpy
import pandas as pd
# Reference to the shapefile and raster dataset from the Contents panel
shapefile_layer = “shapefile_layer_Ci”
raster_layer = “Elevation”
elevation_values = []
# Extract the elevation for each coordinate
arcpy.MakeFeatureLayer_management(shapefile_layer, “temp_layer”)
# Create a temporary feature layer
with arcpy.da.SearchCursor(“temp_layer”, [“SHAPE@XY”]) as cursor:
   for row in cursor:
      x, y = row[0]
      elevation = arcpy.GetCellValue_management(raster_layer, f“{x}
{y}”).getOutput(0)
      elevation_values.append(float(elevation))
# Form pairwise matrix
pairwise_matrix = []
for i in range(len(elevation_values)):
   row = []
   for j in range(len(elevation_values)):
      diff = elevation_values[i] - elevation_values[j]
      # Take the absolute value to ensure no negative distances
      row.append(abs(diff))
   pairwise_matrix.append(row)
# Convert the pairwise matrix to a DataFrame
df = pd.DataFrame(pairwise_matrix)
# Save the DataFrame to a CSV file without column labels
output_csv = r“path\ElevationDifferenceMatrixData_Ci.csv”
df.to_csv(output_csv, index=False, header=False)
print(f“Pairwise matrix saved to {output_csv}”)
```

Figure C-2: Elevation Difference Matrix Data Generation for Each Cluster

```
import arcpy
import numpy as np
import csv
# Define the input shapefile and raster file
input_shapefile = “shapefile_name_Ci”
raster_layer = “GroundFlashDensity”
# Define the number of intervals for sampling
num_intervals = 100
mean_matrix = []
# Convert the raster layer to a NumPy array for efficient sampling
raster_array = arcpy.RasterToNumPyArray(raster_layer, nodata_to_value=0)
# Get the cell size of the raster as a float
cell_size = float(arcpy.GetRasterProperties_management
(raster_layer,“CELLSIZEX”).getOutput(0))
```

```
# Get the extent of the raster
desc = arcpy.Describe(raster_layer)
extent = desc.extent
# Iterate through the input shapefiles features
with arcpy.da.SearchCursor(input_shapefile, ["SHAPE@XY"]) as cursor:
    for row in cursor:
        x1, y1 = row[0]
        row_mean = []
        # Second loop to iterate over the coordinates again
        with arcpy.da.SearchCursor(input_shapefile, ["SHAPE@XY"]) as inner_cursor:
            for inner_row in inner_cursor:
                x2, y2 = inner_row[0]
                if (x1, y1) == (x2, y2):
                    mean = 0
                else:
                    col1 = int((x1 - extent.XMin) / cell_size)
                    row1 = int((extent.YMax - y1) / cell_size)
                    col2 = int((x2 - extent.XMin) / cell_size)
                    row2 = int((extent.YMax - y2) / cell_size)
                    sampled_values = []
                    # Generate additional sample points between the two coordinates
                    for i in range(num_intervals + 1):
                        # Calculate intermediate points along line connecting two coordinates
                        x_interp = x1 + (x2 - x1) * (i / num_intervals)
                        y_interp = y1 + (y2 - y1) * (i / num_intervals)
                        # Calculate the row and column indices for the interpolated point
                        col_interp = int((x_interp - extent.XMin) / cell_size)
                        row_interp = int((extent.YMax - y_interp) / cell_size)
                        # Sample values from the raster array at the interpolated point
                        sampled_value = raster_array[row_interp, col_interp]
                        sampled_values.append(sampled_value)
                    # Calculate the mean of the sampled values
                    mean = np.nanmean(sampled_values)
                row_mean.append(mean)
        mean_matrix.append(row_mean)
# Define the CSV file path
csv_file_path = r"path\AverageGroundFlashDensityMatrixData_Ci.csv"
# Save the mean matrix as a CSV file
with open(csv_file_path, 'w', newline='') as csv_file:
    writer = csv.writer(csv_file)
    writer.writerows(mean_matrix)
print(f"Mean matrix saved to {csv_file_path}")
```

Figure C-3: Average Ground Flash Density Matrix Data Generation for Each Cluster

```
%Load input data and FIS system
DCi = xlsread("path\DistanceMatrixData_Ci.csv");
EDCi = xlsread("path\ElevationDifferenceMatrixData_Ci.csv");
AGFDCi = xlsread("path\AverageGroundFlashDensityMatrixData_Ci.csv");
FISCi = readfis("path\FISCi.fis");
Size = size(DCi, 2);
Output = cell(Size, 1);
p = 1; q = 1; r = 1; % Initialize p, q, and r
% Evaluate the FIS model for each dataset
for i = 1:Size
   D = DCi (:, p);
   ED = EDCi (:, q);
   AGFD = AGFDCi (:, r);
   All = evalfis(FISCi, [D, ED, AGFD]);
   Output{i} = All;
   p = p + 1; q = q + 1; r = r + 1;
end
% Reshape the output
Reshape = cell2mat(Output);
FuzzyMatrix = reshape(Reshape', Size, [])';
% Set diagonal elements of the FuzzyMatrix to 0
FuzzyMatrix(1:Size+1:end) = 0;
```

Figure C-4: Fuzzy Matrix Data Generation for Each Cluster

```
% Creating pairs and converting the distance square matrix to a distance column vector
numberOfGEs = size(FuzzyMatrix, 1);
GEPairs = zeros(numberOfGEs * numberOfGEs, 2);
distanceVector = zeros(numberOfGEs * numberOfGEs, 1);
for g = 1:numberOfGEs
   GEPairs((g - 1) * numberOfGEs + 1:g * numberOfGEs, 1) = g;
   GEPairs((g - 1) * numberOfGEs + 1:g * numberOfGEs, 2) = 1:numberOfGEs;
   distanceVector((g - 1) * numberOfGEs + 1:g * numberOfGEs) = FuzzyMatrix(g, :)';
end
```

Figure C-5: Creation of Pairs and Distance Vectors

```
% Equality Constraints
Aeq = spones(1:length(GEPairs));
beq = numberOfGEs;
Aeq = [Aeq; spalloc(2 * numberOfGEs, length(GEPairs), 2 * numberOfGEs *
(numberOfGEs + numberOfGEs - 1))];
g = 1;
for count = 1:2:(2 * numberOfGEs - 1)
   columnSum = sparse(GEPairs(:, 2) == g);
   Aeq(count + 1, :) = columnSum';
```

```
    rowSum = GEPairs(:, 1) == g;
    Aeq(count + 2, :) = rowSum’;
    g = g + 1;
end
beq = [beq; ones(2 * numberOfGEs, 1)];
nonExists = sparse(distanceVector == 0);
Aeq(2 * g, :) = nonExists’;
beq = [beq; 0];
% Binary Bounds
intcon = 1:length(distanceVector);
lb = zeros(length(distanceVector), 1);
ub = ones(length(distanceVector), 1);
```

Figure C-6: Equality Constraints and Binary Bounds

```
% Optimize using intlinprog
opts = optimoptions(‘intlinprog’, ‘CutGeneration’, ‘Advanced’, ‘NodeSelection’,
‘mininfeas’, ‘Display’, ‘off’);
[decisionVariables, optimumCost, exitflag, output] = intlinprog(distanceVector, intcon,
[], [], Aeq, beq, lb, ub, opts);
% Subtour Detection
x = decisionVariables;
x(x < 0.0001) = 0;
r = find(x);
substuff = GEPairs(r, :);
unvisitedSubTours = ones(length(r), 1);
tours = cell(0);
numberOfTours = 0;
curr = 1;
startour = find(unvisitedSubTours, 1);
while ~isempty(startour)
    home = substuff(startour, 1);
    nextpt = substuff(startour, 2);
    visitedSubTour = nextpt;
    unvisitedSubTours(startour) = 0;
    while nextpt ~= home
        [srow, scol] = find(substuff == nextpt);
        trow = srow(srow ~= startour);
        scol = 3 - scol(trow == srow);
        startour = trow;
        nextpt = substuff(startour, scol);
        visitedSubTour = [visitedSubTour, nextpt];
        unvisitedSubTours(startour) = 0;
    end
    tours{curr} = visitedSubTour;
    curr = curr + 1;
    startour = find(unvisitedSubTours, 1);
end
```

```
numberOfTours = length(tours);
% Subtour Constraints
A = spalloc(0, length(distanceVector), 0);
b = [];
while numberOfTours > 1
   b = [b; zeros(numberOfTours, 1)];
   A = [A; spalloc(numberOfTours, length(distanceVector), numberOfGEs)];
   for count = 1:numberOfTours
      inequalityConstraintNumber = size(A, 1) + 1;
      subTourId = tours{count};
      subTourPairs = nchoosek(1:length(subTourId), 2);
      for jj = 1:size(subTourPairs, 1)
         subTourVariable = (sum(GEPairs == subTourId(subTourPairs(jj, 1)), 2)) & ...
            (sum(GEPairs == subTourId(subTourPairs(jj, 2)), 2));
         A(inequalityConstraintNumber, subTourVariable) = 1;
      end
      b(inequalityConstraintNumber) = length(subTourId) - 1;
   end
```

Figure C-7: Optimization with Subtour Detection and Constraints

```
% Reoptimization
   [decisionVariables, optimumCost, exitflag, output] = intlinprog(distanceVector, intcon,
A, b, Aeq, beq, lb, ub, opts);
  x = decisionVariables;
   x(x < 0.0001) = 0;
   r = find(x);
   substuff = GEPairs(r, :);
   unvisitedSubTours = ones(length(r), 1);
   tours = cell(0);
   numberOfTours = 0;
   curr = 1;
   startour = find(unvisitedSubTours, 1);
   while ~isempty(startour)
      home = substuff(startour, 1);
      nextpt = substuff(startour, 2);
      visitedSubTour = nextpt;
      unvisitedSubTours(startour) = 0;
      while nextpt ~= home
         [srow, scol] = find(substuff == nextpt);
         trow = srow(srow ~= startour);
         scol = 3 - scol(trow == srow);
         startour = trow;
         nextpt = substuff(startour, scol);
         visitedSubTour = [visitedSubTour, nextpt];
         unvisitedSubTours(startour) = 0;
      end
```

```
        tours{curr} = visitedSubTour;
        curr = curr + 1;
        startour = find(unvisitedSubTours, 1);
    end
    numberOfTours = length(tours);
    fprintf('TSP Configuration:%d\n', tours);
end
```

Figure C-8: Re-Optimization and Subtour Elimination Loop

Table C-1: Fuzzy Rule Setting

R	d	Δe	$\overline{GFD}$	f	R	d	Δe	$\overline{GFD}$	f	R	d	Δe	$\overline{GFD}$	f	R	d	Δe	$\overline{GFD}$	f	R	d	Δe	$\overline{GFD}$	f
1	VL	VL	VL	EL	26	L	VL	VL	VL	51	M	VL	VL	VL	76	H	VL	VL	L	101	VH	VL	VL	ML
2	VL	VL	L	VL	27	L	VL	L	VL	52	M	VL	L	L	77	H	VL	L	ML	102	VH	VL	L	ML
3	VL	VL	M	VL	28	L	VL	M	L	53	M	VL	M	ML	78	H	VL	M	ML	103	VH	VL	M	M
4	VL	VL	H	L	29	L	VL	H	ML	54	M	VL	H	ML	79	H	VL	H	M	104	VH	VL	H	MH
5	VL	VL	VH	ML	30	L	VL	VH	ML	55	M	VL	VH	M	80	H	VL	VH	MH	105	VH	VL	VH	MH
6	VL	L	VL	VL	31	L	L	VL	VL	56	M	L	VL	L	81	H	L	VL	ML	106	VH	L	VL	ML
7	VL	L	L	VL	32	L	L	L	L	57	M	L	L	ML	82	H	L	L	ML	107	VH	L	L	M
8	VL	L	M	L	33	L	L	M	ML	58	M	L	M	ML	83	H	L	M	M	108	VH	L	M	MH
9	VL	L	H	ML	34	L	L	H	ML	59	M	L	H	M	84	H	L	H	MH	109	VH	L	H	MH
10	VL	L	VH	ML	35	L	L	VH	M	60	M	L	VH	MH	85	H	L	VH	MH	110	VH	L	VH	H
11	VL	M	VL	VL	36	L	M	VL	L	61	M	M	VL	ML	86	H	M	VL	ML	111	VH	M	VL	M
12	VL	M	L	L	37	L	M	L	ML	62	M	M	L	ML	87	H	M	L	M	112	VH	M	L	MH
13	VL	M	M	ML	38	L	M	M	ML	63	M	M	M	M	88	H	M	M	MH	113	VH	M	M	MH
14	VL	M	H	ML	39	L	M	H	M	64	M	M	H	MH	89	H	M	H	MH	114	VH	M	H	H
15	VL	M	VH	M	40	L	M	VH	MH	65	M	M	VH	MH	90	H	M	VH	H	115	VH	M	VH	VH
16	VL	H	VL	L	41	L	H	VL	ML	66	M	H	VL	ML	91	H	H	VL	M	116	VH	H	VL	MH
17	VL	H	L	ML	42	L	H	L	ML	67	M	H	L	M	92	H	H	L	MH	117	VH	H	L	MH
18	VL	H	M	ML	43	L	H	M	M	68	M	H	M	MH	93	H	H	M	MH	118	VH	H	M	H
19	VL	H	H	M	44	L	H	H	MH	69	M	H	H	MH	94	H	H	H	H	119	VH	H	H	VH
20	VL	H	VH	MH	45	L	H	VH	MH	70	M	H	VH	H	95	H	H	VH	VH	120	VH	H	VH	VH
21	VL	VH	VL	ML	46	L	VH	VL	ML	71	M	VH	VL	M	96	H	VH	VL	MH	121	VH	VH	VL	MH
22	VL	VH	L	ML	47	L	VH	L	M	72	M	VH	L	MH	97	H	VH	L	MH	122	VH	VH	L	H
23	VL	VH	M	M	48	L	VH	M	MH	73	M	VH	M	MH	98	H	VH	M	H	123	VH	VH	M	VH
24	VL	VH	H	MH	49	L	VH	H	MH	74	M	VH	H	H	99	H	VH	H	VH	124	VH	VH	H	VH
25	VL	VH	VH	MH	50	L	VH	VH	H	75	M	VH	VH	VH	100	H	VH	VH	VH	125	VH	VH	VH	EH

```
while true
   % Define ThingSpeak read channel parameters
   channelID = Channel_ID;
   readAPIKey = 'Read_API_Key';
   % Define the time range for historical data retrieval
   startTime = datetime('YYYY-MM-DD HH:MM:SS', 'InputFormat', 'yyyy-MM-dd HH:mm:ss', 'TimeZone', 'UTC');
   endTime = datetime('YYYY-MM-DD HH:MM:SS', 'InputFormat', 'yyyy-MM-dd HH:mm:ss', 'TimeZone', 'UTC');
   % Read data from ThingSpeak channel for multiple fields
   [data, timestamps] = thingSpeakRead(channelID, 'Fields', [1, 2, 3], 'DateRange', [startTime, endTime], 'ReadKey', readAPIKey);
   % Convert timestamps to a numeric array
   timestamps_numeric = datenum(timestamps);
   % Convert numeric timestamps to datetime
   timestamps = datetime(timestamps_numeric, 'ConvertFrom', 'datenum', 'TimeZone', 'UTC');
   % Get the current time
   current_time = datetime('now', 'TimeZone', 'local');
   % Extract the time component from all timestamps
   times = timestamps - dateshift(timestamps, 'start', 'day');
   % Calculate the time difference between the current time and all times
   time_diff = abs(times - timeofday(current_time));
   % Find the index of the timestamps which are in the past compared to the current time
   past_indices = find(times <= timeofday(current_time));
   % Processing demand data
   if ~isempty(past_indices)
      % Find the index of the closest timestamp in the past
      [~, idx] = min(time_diff(past_indices));
      % Get the index in the original data array
      idx = past_indices(idx);
      % Display the values corresponding to the closest timestamp in the past for all fields
      disp('Data:');
      disp(['Current Time:', datestr(current_time, 'HH:MM')]);
      disp(['Closest Past Timestamp: ', datestr(timestamps(idx), 'HH:MM')]);
      disp(['Value at Closest Past Timestamp (Field 1): ', num2str(data(idx, 1))]);
      disp(['Value at Closest Past Timestamp (Field 2): ', num2str(data(idx, 2))]);
      disp(['Value at Closest Past Timestamp (Field 3): ', num2str(data(idx, 3))]);
   else
      disp('No past data available.');
   end
      demand = (str2double(num2str(data(idx, 1))))*100;
      GHI = str2double(num2str(data(idx, 2)));
      temperature = str2double(num2str(data(idx, 3)));
      % ThingSpeak Channel ID and Write API Key
      channelID1 = Channel_ID_New;
```

```
        writeAPIKey = ‘Write_API_Key’;

        % Create ThingSpeak URL
        baseURL = ‘https://api.thingspeak.com/update’;
        thingSpeakWriteURL = sprintf(‘%s?api_key=%s’, baseURL, writeAPIKey);
        % Construct the data string
        data = sprintf(‘field1=%.2f&field2=%.2f&field3=%.2f’, demand, GHI,
temperature);
        % Send data to ThingSpeak
        response = webwrite(thingSpeakWriteURL, data);
    % Pause for 5 seconds before the next iteration
    pause(10);
end
```

Figure D-1: Read Current Time Data and Write on ThingSpeak Cloud

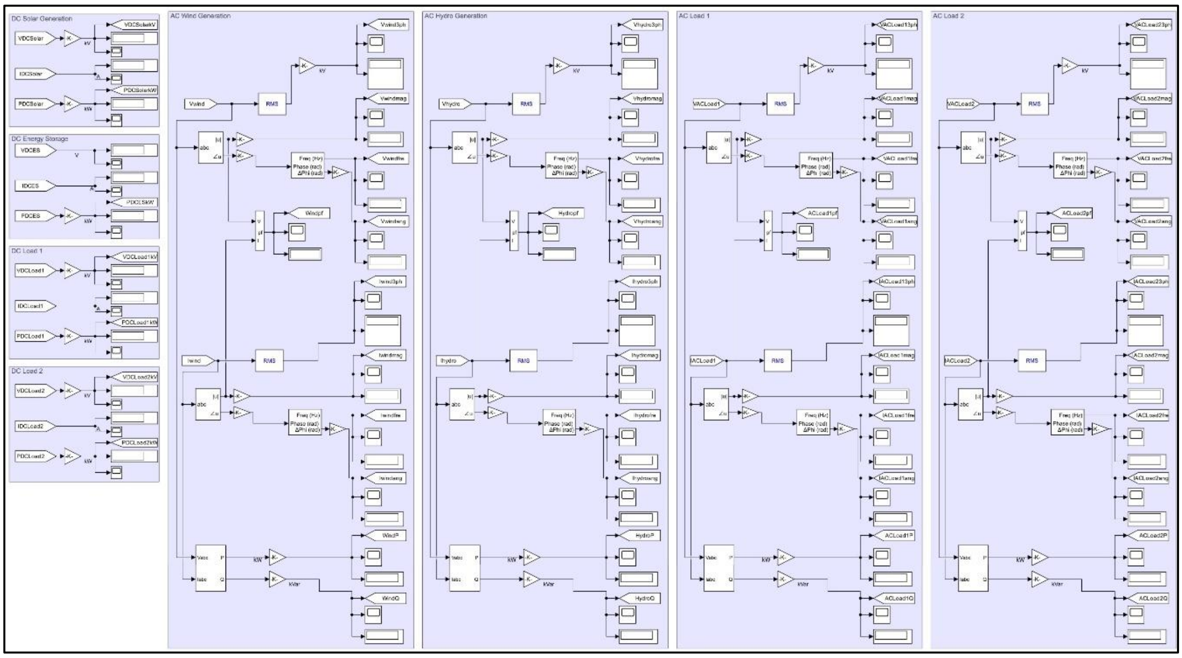
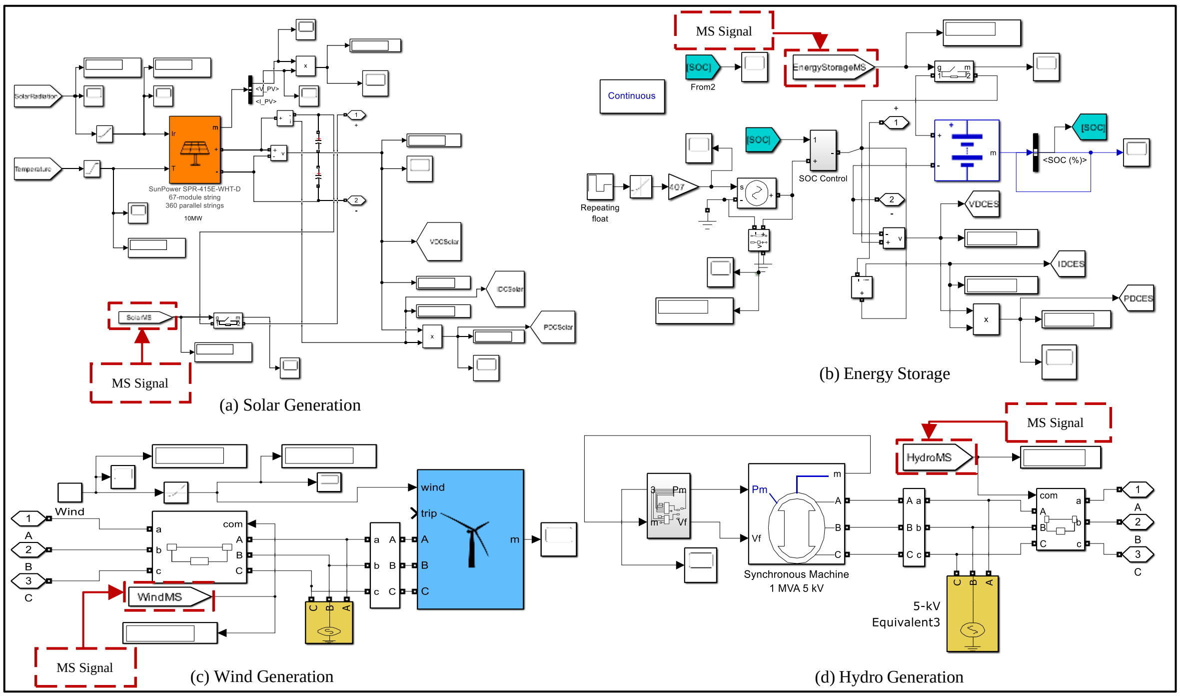
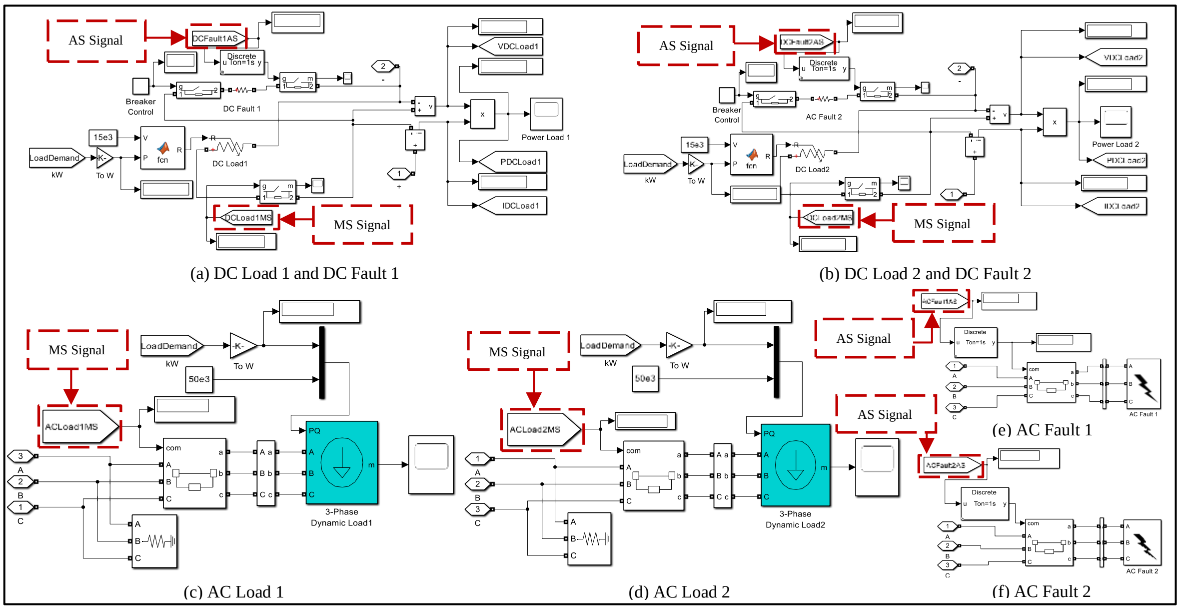


Figure D-2: Establishment of Monitoring Framework in MATLAB Simulink



Figures D-3 (a) to (d): Integration of Manual Switches with Signal Channels into Green Energy Generations and Energy Storage Systems



Figures D-4 (a) to (f): Integration of Manual and Automatic Switches into Loads (AC and DC) and Faults (AC and DC)

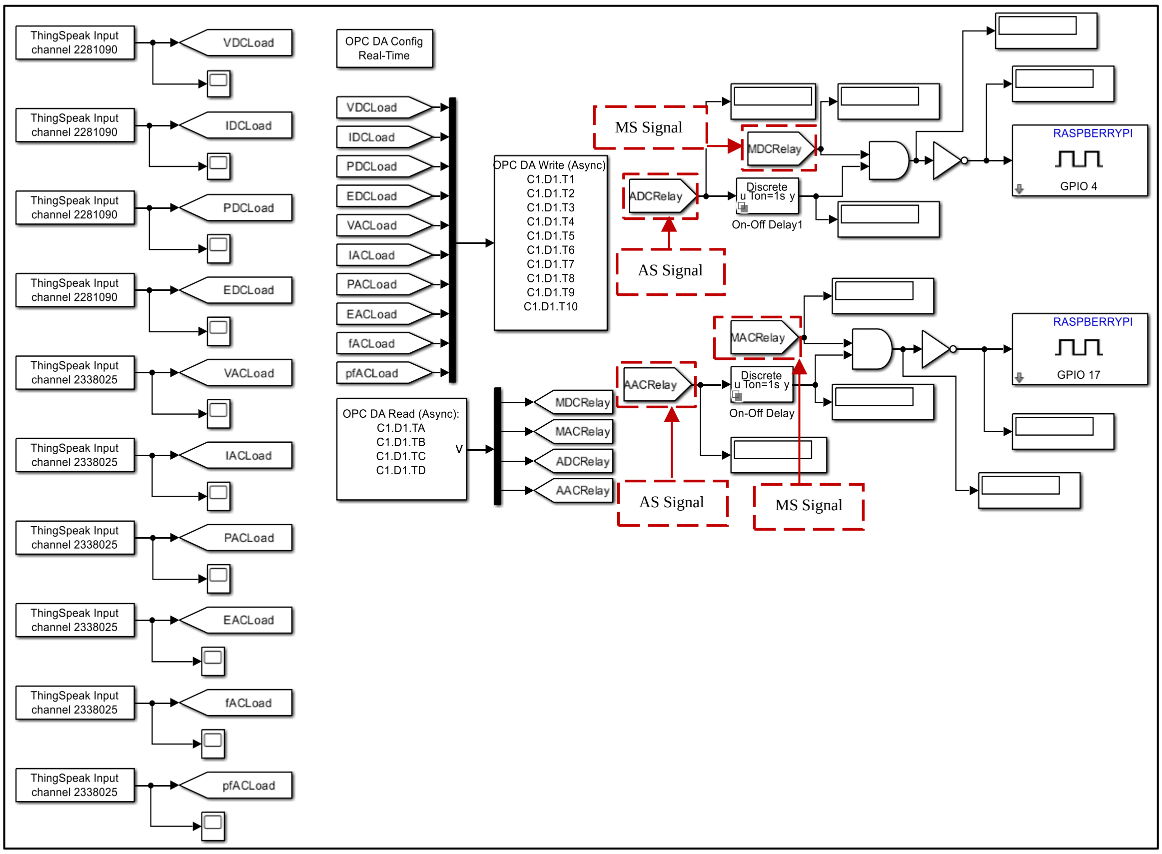


Figure D-5: MATLAB Communication Framework for Hardware Prototype