engineering proceedings

MDPI

Proceeding Paper

Real-Time Intelligent Energy Management System for Microgrid Operation †

David Díaz *,‡, Jefersson Solarte ‡ and Javier Revelo-Fuelagán ‡

Department of Electronics, Faculty of Engineering, Universidad de Nariño, Pasto 520002, Colombia; dariosolarte12@udenar.edu.co (J.S.); javierrevelof@udenar.edu.co (J.R.-F.)

* Correspondence: dsdiazhack666@udenar.edu.co

† Presented at the Sixth International Conference on Information Systems and Software Technologies (ICI2ST 2026), Quito, Ecuador, 23–25 September 2026.

‡ These authors contributed equally to this work.

Abstract

The increasing penetration of intermittent renewable energy sources, together with the operational complexity of energy storage and conventional generation, requires advanced energy management strategies for microgrids. This work presents the Intelligent Energy Management System (SIGE), an integrated platform for hybrid microgrid operation that combines low-latency IoT monitoring, probabilistic time-series forecasting, stochastic optimization with model predictive control (MPC), multidimensional statistical anomaly detection, and a multi-agent system based on large language models (LLMs). The platform operates in a closed loop, acquiring real-time data through the MQTT protocol, generating 72 h probabilistic weather forecasts using a PatchTST Transformer, converting these forecasts into solar-power scenarios through a grey-box PV model, and solving the stochastic economic dispatch using Pyomo and Gurobi. A 14-day closed-loop evaluation was conducted using real measurements from the microgrid, including photovoltaic generation and load demand. The MPC-based strategies achieved a net operating profit of USD 357.18, compared with USD 29.48 for a rule-based heuristic strategy, while the stochastic MPC operated within 0.27% of a measured-data benchmark based on the corresponding measured operating trajectories. The complete optimization cycle was solved in less than 0.7 s, substantially below the 15 min control interval. These results demonstrate the feasibility of integrating probabilistic forecasting, mathematical optimization, IoT monitoring, and AI-based supervision into a closed-loop decision-support platform for microgrid operation with distributed renewable energy resources.

Keywords: microgrids; intelligent management; model predictive control (MPC); stochastic optimization; PatchTST; probabilistic forecasting; LLM agents; IoT

Check for updates

Academic Editor: Samuel Lascano

Published: 18 September 2026

Copyright: © 2026 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license.

1. Introduction

Microgrids, defined as integrated generation, storage, and consumption systems capable of operating in islanded or grid-connected mode, represent a fundamental paradigm in the energy transition. However, their efficient operation faces considerable challenges: photovoltaic generation is inherently intermittent and driven by meteorological variability; diesel generators, although dispatchable, exhibit non-linear operational costs and environmental externalities; battery storage introduces inter-temporal coupling between consecutive dispatch intervals; and the external grid imposes exchange limits and time-varying tariffs [1].

Decision-making in this context therefore requires continuously solving a multi-period optimization problem under uncertainty, driven by real-time meter data and forecasts of future conditions. Traditional rule-based heuristics and simple deterministic optimization fail to capture the stochastic nature of renewable resources and do not exploit the temporal granularity offered by modern data-acquisition systems [2].

The SIGE platform is therefore developed for the optimal operation of microgrids and integrates the following functionalities: (1) real-time acquisition and monitoring of electrical variables through smart energy meters installed in the microgrid and connected using IoT technology, utilizing the MQTT protocol for data transmission and MongoDB for scalable persistence; (2) multi-horizon time-series forecasting of solar irradiance and load demand, allowing the progressive integration of machine learning models, including a PatchTST Transformer model trained on six years of ERA5 reanalysis data; (3) stochastic optimization with model predictive control (MPC), solving an economic energy dispatch problem that considers multiple weather scenarios weighted by probability, with a 24 h horizon and hourly resolution; (4) real-time anomaly detection using three complementary statistical strategies (thresholds, rate of change, contextual Z-score), which feed a multi-agent system based on LLMs to generate diagnostics and operational recommendations in natural language; (5) an interactive operator interface that includes a single-line diagram editor, graphical visualization of dispatch plans, real-time monitoring of energy meters, and dynamically AI-generated dashboards. The central objective of the platform is to provide a real-time decision-support system that, using live data from microgrid components, predicts future conditions and enables automated microgrid operation, thereby reducing operational costs and maximizing supply reliability.

Intelligent microgrid management has been addressed from several fronts, with energy time-series forecasting being a prominent research direction. While statistical models such as ARIMA and recurrent architectures such as LSTM achieve accurate short-horizon forecasts, they struggle with long-term dependencies and complex exogenous inputs. In response, the Temporal Fusion Transformer (TFT) [3] introduces attention mechanisms and adaptive gating, enabling multi-horizon forecasts with uncertainty quantiles, which is a critical capability for renewable-generation and demand forecasting [4,5]. More recently, patch-based channel-independent Transformer architectures such as PatchTST [6] have outperformed attention-gating models on long-horizon forecasting benchmarks, providing an effective basis for probabilistic weather forecasting of renewable generation.

Regarding optimal microgrid operation, stochastic programming has been widely adopted to model uncertainty in renewable generation and demand [7–11]. Model predictive control (MPC) [12] complements this approach through a receding-horizon strategy that periodically recomputes the optimal solution as new real-time information becomes available. Algebraic modeling environments such as Pyomo [13] facilitate the formulation of these problems, while solvers such as Gurobi [14] and HiGHS [15] enable their efficient solution.

In the domain of artificial intelligence for power-system supervision, large language models (LLMs) have recently been integrated as cognitive assistants for operators. Frameworks like LangChain [16] and LangGraph allow the orchestration of multi-agent workflows that process energy meter data, detect anomalous patterns, and generate diagnostics in natural language. However, the integration of these three dimensions, namely forecasting, stochastic optimization, and AI agents, into a closed-loop platform operating on real microgrid data remains an open research gap that this work addresses [17].

Beyond economic optimality, recent work emphasizes resilient operation under cyber–physical disturbances. Han et al. [18] propose a resilient optimal dispatch strategy for ship-integrated energy systems based on an enhanced traffic jam optimizer, addressing

degraded operating conditions and false-data injections. In SIGE, resilience is addressed at the data layer: the multidimensional statistical anomaly detection described in Section 2.2 and the LLM-based supervisory layer identify compromised or erroneous measurements before they propagate into the optimization loop, complementing the economic dispatch with a security-aware operating envelope.

In this context, this research work presents the following contributions: first, the integration of a probabilistic weather-forecasting pipeline based on the PatchTST Transformer [6], which produces 72 h P10/P50/P90 climate scenarios that are converted into solar-power bands by a grey-box calibrated PV physics model; second, the embedding of these probabilistic scenarios into a stochastic MPC control loop, enabling the dispatch formulation to explicitly account for forecast uncertainty; and third, the development of a multi-agent system based on LLMs capable of processing statistical anomaly analyses to provide real-time operational assistance without human intervention. The relationship between the identified research gaps and the contributions of this work is summarized in Table 1.

Table 1. Synthesis of related work, research gaps, and SIGE contributions.

References	Area	Limitation (Gap)	How SIGE Addresses It
[3–6]	Forecasting	Forecasts not embedded in the control loop	Modular probabilistic weather forecaster (PatchTST) + calibrated PV model feeding MPC scenarios
[7–12]	Stochastic opt. & MPC	Evaluated offline/in isolation	Closed-loop MPC on real IoT data, 15 min recompute
[16,17]	LLM infrastructure	No safety envelope for control	Advisory-only LLM: allowlist + numeric verification + audit
[18]	Resilience	Focused on ship-integrated systems	Data-layer resilience: anomaly detection + LLM supervision
[2]	Heuristics	No stochastic capture	Stochastic MPC with probabilistic scenarios
[1]	Microgrid paradigm	Motivates scope	SIGE hybrid closed-loop scope
This work	Integrated EMS	Limited integration of forecasting, stochastic optimization, and AI supervision in closed loop	Hybrid closed-loop platform (IoT + forecasting + stochastic MPC + LLM supervision) on real microgrid data

The remainder of this paper is organized as follows: Section 1 introduces the microgrid paradigm and outlines the development of the SIGE platform. Section 2 details the methodology, describing the multilayer technological architecture that integrates IoT instrumentation, probabilistic weather forecasting with the PatchTST Transformer [6] and a grey-box calibrated PV physics model, the stochastic optimization loop (MPC), and the multi-agent system based on LLMs for anomaly detection and operational assistance. Subsequently, Section 3 presents the experimental results obtained during system validation, including performance metrics and the analysis of the microgrid's behavior. Finally, Section 4 discusses the scope and limitations of the study, culminating with the conclusions that validate the viability of this hybrid approach for automated decision-making in distributed energy infrastructure.

2. Materials and Methods

SIGE is organized into three technological layers that communicate via REST APIs, WebSockets, Redis, and MQTT, as shown in Figure 1:

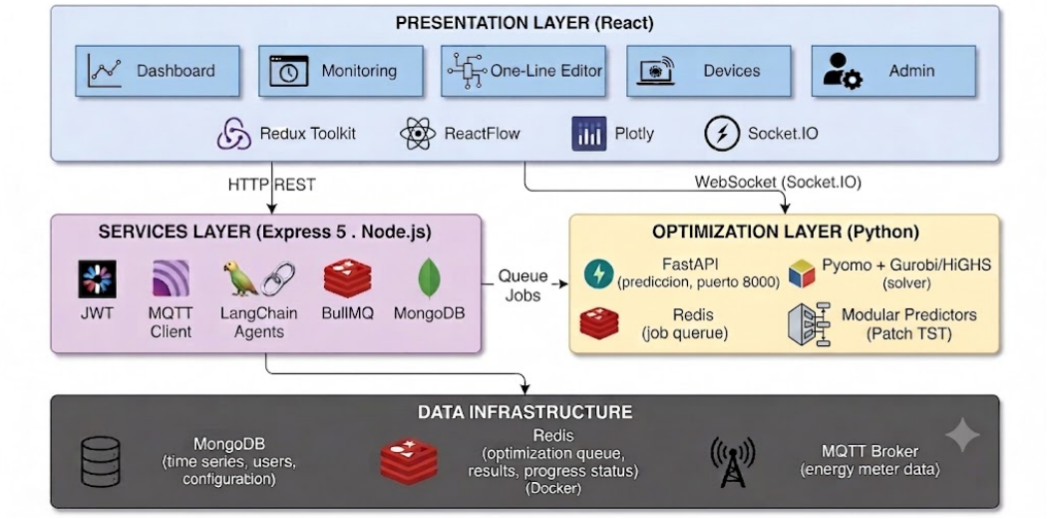


Figure 1. Multilayer technological architecture of the SIGE system.

The first layer corresponds to the implementation of functional modules accessible via tabbed navigation: Optimization Dashboard, Real-time Monitoring, Single-line Diagram Editor, IoT Device Management, AI-generated Dashboard (server-driven UI), and Operator Administration. The application state is managed by using Redux Toolkit with specialized reducers. Real-time communication with the backend is carried out via Socket.IO, while CRUD operations use a REST API with JWT authentication.

The service layer, running on Node.js with Express 5, acts as the central orchestrator and performs the following functions: manages authentication, maintains the connection with the MQTT broker for data ingestion, coordinates MPC cycles, executes the AI agent pipeline, and serves as a bridge with Python services.

The optimization layer, implemented in Python, contains two independent services: a FastAPI API that exposes forecasts generated by modular predictors and a Pyomo-based resolution engine that builds and solves the stochastic economic dispatch model using Gurobi as the main solver, with HiGHS as an open-source alternative. The data flow in the platform follows a continuous pipeline divided into two main circuits, shown in Figure 2:

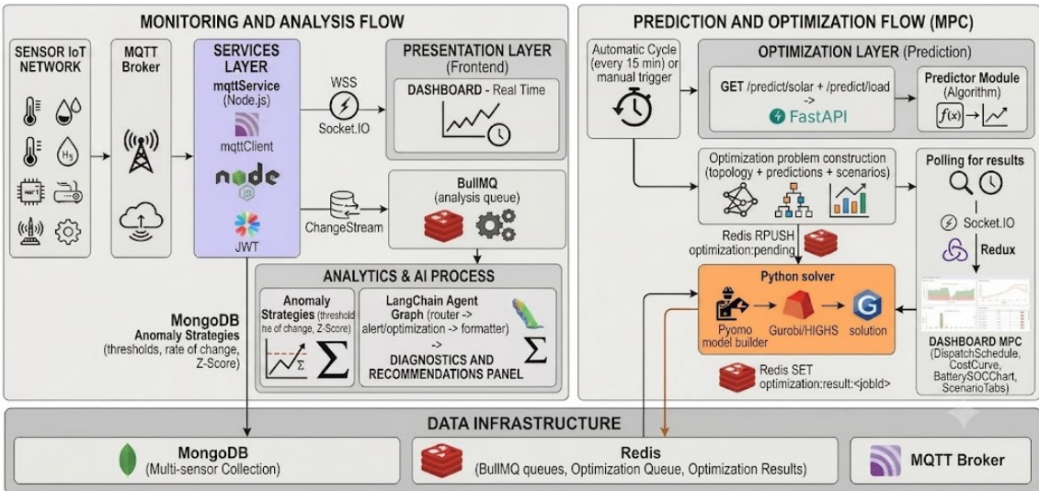


Figure 2. Data flow.

The platform connects to an external MQTT broker to which the energy meters installed in the microgrid are linked. These devices publish their measurements to specific topics,

and the system maintains an active subscription to authorized topics, allowing it to receive energy meter readings in real time [19].

Validated data is persisted in MongoDB using a dynamic storage model, where each device has its own partitioned collection. This strategy facilitates time-series queries and allows the establishment of differentiated retention policies depending on the nature of the sensor. Simultaneously, each reading is transmitted to the frontend using rooms; this allows updating the monitoring graphs with sub-second latencies. Figure 3 shows the real-time monitoring interface with energy meter readings.



Figure 3. Real-time monitoring module with energy measurement logging.

The ChangeStream functionality of MongoDB allows the platform to detect every new insertion without the need for periodic polling. The `changeStreamService` maintains open cursors on all active collections, queuing data asynchronously in the Node.js BullMQ library.

Regarding the forecasting system, the platform integrates a modular interface that decouples the forecasting engine from the rest of the system, allowing the modular implementation of various algorithms without altering backend orchestration.

For production execution, the system implements an API developed in FastAPI that decouples the prediction engine from the rest of the platform. This architecture allows incorporating and substituting different forecasting models without requiring modifications to the orchestration flow or the other system components.

Currently, the platform offers two interchangeable layers through this interface. The first layer is a deterministic historical predictor implemented in Python using SciPy 1.10.1 to parse preprocessed `.mat` time-series files, which serves as a fallback baseline. The second layer is a probabilistic weather-forecasting pipeline whose operational deep-learning provider is the PatchTST, an open-source deep learning Transformer model [6]; the Open-Meteo numerical weather prediction (NWP) service is used as an external reference for the validation reported in Section 3.

The PatchTST model is a channel-independent patch-based Transformer trained on six years of ERA5 reanalysis data for Pasto, Nariño, and produces 72 h probabilistic forecasts of the variables governing photovoltaic generation and load, specifically irradiance (global, direct, and diffuse) and temperature, with P10, P50, and P90 quantiles. ERA5 is a long-term atmospheric reanalysis product used here both for model training and, in the closed-loop evaluation, as the realized weather reference against which the dispatch is settled. The

Open-Meteo numerical weather prediction service is used as an independent external reference for the forecast-validation exercise reported in Section 3, and is not part of the operational chain. Each PatchTST forecast is then passed through a grey-box calibrated PV physics model: a pvlib-based physical plant whose parameters are first calibrated from measured data (losses, inverter efficiency, temperature coefficient) and then corrected by a residual-learning stage that captures non-physical effects (soiling, shading, degradation). In the present validation the residual-correction stage is driven by synthetic residuals with a physical ground truth, whereas the plant parameters themselves are calibrated from real measurements; closing this stage with live sensor data is outlined as future work. This two-level chain converts the climate scenarios into solar-power bands (P10/P50/P90 in kW) that feed the stochastic MPC loop, replacing the previous deterministic approach and allowing optimization under climate variability.

2.1. Stochastic Optimization and Model Predictive Control

The mathematical core of SIGE is a stochastic economic-dispatch problem formulated as a quadratic program with continuous variables over a 24-period (hourly) horizon that explicitly accounts for uncertainty.

The optimization model determines, hour by hour and under each stochastic scenario, the power (kW) to be generated by each available diesel unit to guarantee supply, as well as the power exchange with the main grid, dynamically identifying when energy imports are required to cover internal demand.

Regarding storage management, the model controls the bidirectional power flows of the battery units, defining the net charge and discharge power in each period to optimize resource utilization. The state of charge of each battery at the end of each hour (kWh) is tracked to ensure precise energy accounting and compliance with equipment physical limits.

The uncertainty of solar generation is represented by $S = 3$ discrete scenarios anchored to the quantiles produced by the forecasting chain: Sunny (P90), Cloudy (P50), and Rainy (P10), with probabilities $\text{prob}[s] = \{0.2,\ 0.6,\ 0.2\}$ reflecting the empirical frequency of clear, mixed, and overcast days at the study site. For each scenario, the corresponding irradiance quantile is propagated through the calibrated PV model described above, yielding a solar-power trajectory $P_{\text{solar}}[t, s]$ that is injected as a pre-calculated stochastic parameter into the power balance constraint. The scenario set is regenerated at every MPC execution from the freshest forecast, so that the probability-weighted objective in Equation (1) hedges the dispatch against the full range of plausible conditions. With $S = 1$ (P50 only) the formulation reduces to the deterministic case used as a baseline in Section 3.

Energy management in the microgrid is posed as an optimization problem that minimizes the total operating cost, as shown in Equation (1), weighted by the probabilities $\text{prob}[s]$ of each scenario, including quadratic cost functions for the diesel generator, grid tariffs, and linear penalties for battery degradation:

$$\min \sum_{s} \text{prob}[s] \cdot \sum_{t} \left[\sum_{d} (c_d + b_d P_d + a_d P_d^2) C_{\text{comb}} + (C_{\text{fijo}} + C_{\text{var}} P_{\text{grid}}) + \sum_{b} \text{deg}(P_{\text{ch}} + P_{\text{dis}}) \right] \quad (1)$$

The solution space is bounded by the power balance, defined as $P_{\text{diesel}} + P_{\text{grid}} + P_{\text{solar}} + P_{\text{discharge}} - P_{\text{charge}} \geq P_{\text{load}}$, as well as by the operational limits of the diesel generators, expressed by $P_{\min}[d] \leq P_{\text{diesel}}[d, t, s] \leq P_{\max}[d]$. Additionally, the system is subject to power grid constraints, $P_{\text{grid,min}} \leq P_{\text{grid}}[t, s] \leq P_{\text{grid,max}}$, and battery state of charge limits, defined by $SOC_{\min}[b] \leq SOC[b, t, s] \leq SOC_{\max}[b]$.

Battery dynamics are presented by Equations (2) and (3), which describe the temporal evolution of the state of charge (SOC) of each battery over the optimization horizon. These relationships consider charge and discharge efficiencies, the initial state of charge, and the

intertemporal coupling between consecutive periods. The discretization step is $\Delta t = 0.25$ h; SOC is expressed in kWh, whereas P_{ch} and P_{dis} are in kW, so the product (kW) · (h) yields kWh and the SOC balance is dimensionally consistent,

$$SOC[b,0,s] = SOC_{ini}[b] + \left(\eta_{ch}[b] \cdot P_{ch}[b,0,s] - \frac{P_{dis}[b,0,s]}{\eta_{dis}[b]}\right) \Delta t \quad (2)$$

$$SOC[b,t,s] = SOC[b,t-1,s] + \left(\eta_{ch}[b] \cdot P_{ch}[b,t,s] - \frac{P_{dis}[b,t,s]}{\eta_{dis}[b]}\right) \Delta t, \quad t > 0 \quad (3)$$

although the objective function already penalizes simultaneous charge and discharge through the degradation term deg > 0, an explicit complementarity constraint is imposed to guarantee physical feasibility. Binary indicators $y_{ch}[b,t,s], y_{dis}[b,t,s] \in \{0,1\}$ enforce that at most one operation mode is active per battery, period, and scenario:

$$y_{ch}[b,t,s] + y_{dis}[b,t,s] \leq 1 \quad (4)$$

$$P_{ch}[b,t,s] \leq P_{ch,max}[b] \cdot y_{ch}[b,t,s] \quad (5)$$

$$P_{dis}[b,t,s] \leq P_{dis,max}[b] \cdot y_{dis}[b,t,s] \quad (6)$$

Resolution and Predictive Control Loop (MPC)

The problem is solved using Gurobi as the main solver, with HiGHS as an open-source alternative for user selection. In the deployed system the closed MPC loop recalculates the full 24 h horizon every 15 min, implementing only the action of the first time period according to the receding-horizon strategy. Communication between the Node.js backend and the Python solver is done via Redis. The backend queues optimization jobs in a shared list within Redis, and the solver consumes them for processing; once the model is solved, the results and progress state are stored in the same Redis server from which the backend retrieves them through periodic polling. The 14-day closed-loop evaluation reported in Section 3 adopts the same receding-horizon principle but resolves the horizon on an hourly cadence; this reduces the computational burden of the multi-day experiment while leaving the executed first action unchanged, so the economic comparison across strategies remains valid.

2.2. Anomaly Detection and AI Multi-Agent System

The statistical detection strategies employed establish that each meter reading is subjected to three complementary analyses executed in parallel:

1. Threshold strategy: Verifies that each electrical variable is within predefined physical ranges, with configurable limits by device type (solar panels, inverters, batteries, meters, microgrid).
2. Rate of change: Calculates the relative variation between consecutive readings $|(x_t - x_{t-1})/x_{t-1}|$. To avoid division by zero when the previous reading is null (e.g., photovoltaic power at night), the strategy switches to an absolute-difference test $|x_t - x_{t-1}|$ compared against an absolute threshold specific to each variable, instead of forcing a spurious 100% jump. Thresholds are defined per variable type with physical justification: voltage ±10% (grid regulation), active/reactive power ±50% (normal PV and demand ramps), current ±50%, and state of charge ±20% (bounded by the battery management system); unclassified variables fall back to a 50% relative threshold. This design prevents false positives during null-base transitions while preserving sensitivity to genuine abrupt changes.
3. Contextual Z-score: Evaluates each reading against the mean and standard deviation of a rolling historical window (30 readings by default), flagging values with $|Z| > 3.0$.

A strategy log aggregates the results of the three techniques and assigns an alert level: `Critical` when two or more strategies detect an anomaly, `Warning` when only one detects it, and `Normal` in the absence of signals.

Multi-Agent System Based on LLMs

SIGE integrates a cognitive system implemented with LangChain and LangGraph that processes the statistical analysis results and acts as an operator copilot. The state graph contains four specialized nodes:

- Router: Evaluates the presence of anomalies in the previous analysis and routes the flow to the alert or optimization node accordingly.
- Alert: Under anomalous conditions, invokes the LLM (Llama 3.3 70B via Groq, or GPT-4o-mini via OpenAI as an alternative) to generate a diagnostic summary and actionable recommendations.
- Optimization: In a normal operating state, the LLM suggests guidelines for preventive maintenance and operational efficiency.
- Formatter: Structures the final response with visualization data for the frontend.

The LLM responses are validated against `Zod` schemas to ensure their structural integrity before being transmitted to the frontend via Socket.IO. Beyond structural validation, the system enforces a multi-layer safety design: (i) the LLM is strictly advisory and is never granted the ability to execute control actions; its output is a recommendation that the physical control loop executes exclusively through the MPC and human operator confirmation, so the LLM remains outside the control data path; (ii) an allowlist restricts the recommended action types to those predefined by the system; (iii) numerical verification checks that any suggested setpoint lies within physically admissible ranges derived from the input data; (iv) every recommendation is logged with its justification to support traceability and audit. In the operator interface, recommendations are presented as suggestions that require operator confirmation before any effect.

2.3. Operator Interface

The platform includes a single-line diagram editor implemented with ReactFlow that allows operators to build and modify the microgrid's topology using configurable nodes (solar panel, diesel generator, power grid, inverter, battery, and load). The topology defined in the editor is automatically translated into solver constraints when triggering an optimization. Each diagram node maps to its corresponding entry in the `sources`, `storage`, `loads`, or `grid` arrays consumed by the Pyomo model. Figure 4 shows the single-line diagram editor of a grid topology.

AI-Generated Dashboard and Multi-Agent Analysis

SIGE offers complementary features powered by artificial intelligence. Specifically, the Server-Driven Dashboard module employs an LLM (Groq/OpenAI) to dynamically generate a JSON dashboard specification. This specification defines widgets (metric cards, energy meter tables, line and bar charts) that are rendered with real data from the user's authorized energy meters. In the absence of an LLM API key, the system produces a default dashboard based on the latest available readings. Figure 5 shows the AI agent console with real-time analysis and recommendations generated by the LangGraph graph.

https://doi.org/10.3390/engproc2026153029

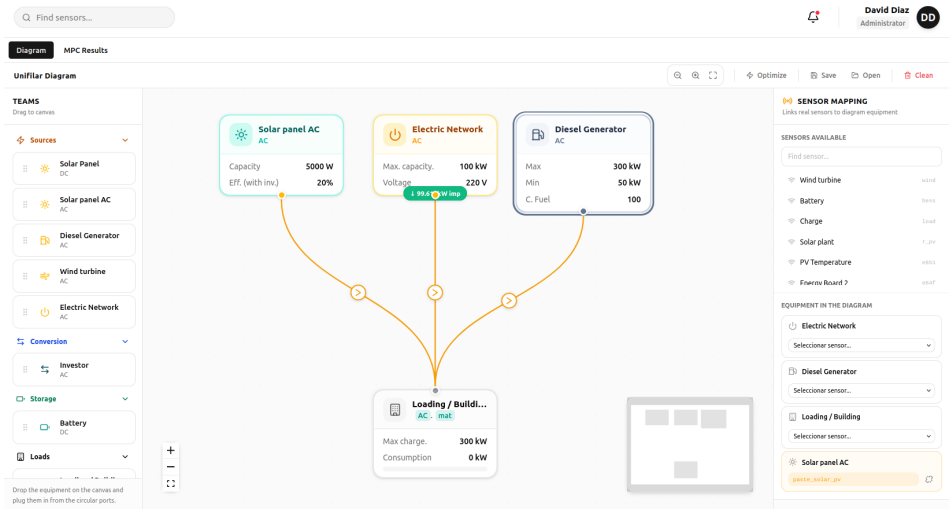


Figure 4. Single-line diagram editor that is automatically converted into solver constraints.

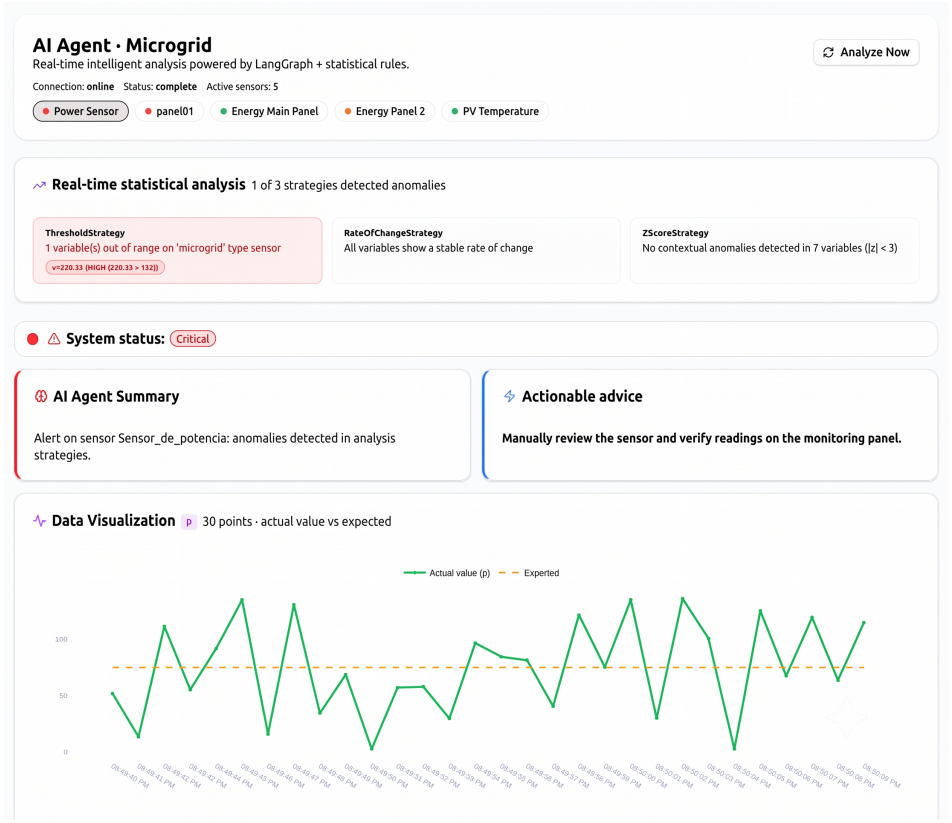


Figure 5. AI agent server-driven UI console.

3. Results

The experimental validation of SIGE was conducted on a hybrid microgrid testbed integrating IoT energy meters. The obtained results are presented following the functional flow of the SIGE platform. Initially, the configuration used during the validation is described, followed by an analysis of the optimization cycle results, then the performance of the prediction model, and finally, the functional integration of the different modules that make up the system.

Figure 6 presents the topology used during the experimental validation of the system. The considered microgrid consists of photovoltaic panels, a diesel generator, three electrical loads, and the connection to the main grid. This microgrid was configured on the platform

using the single-line diagram editor and subsequently automatically translated to the optimization model implemented in Pyomo 6.10.0.

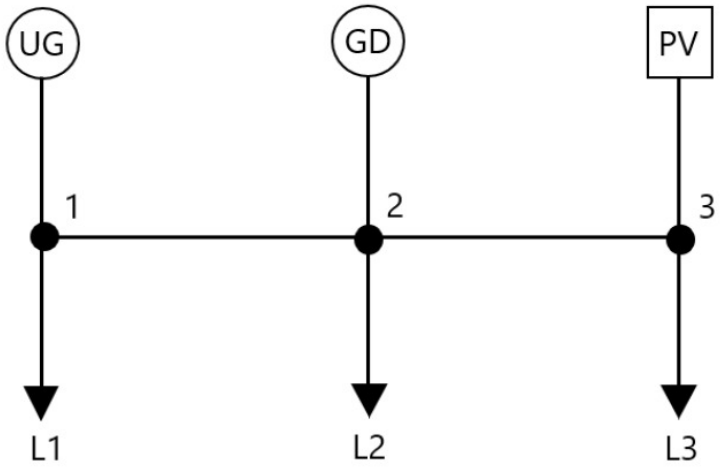


Figure 6. Microgrid topology used for system validation.

This topology serves as the test scenario for evaluating the predictive-control cycle and verifying the integration between the graphical editor, the mathematical model, and the optimization engine.

The obtained result shows the power distribution among the available sources during the optimization horizon and the projected evolution of the batteries' state of charge. Operational indicators used to monitor each execution cycle are also generated from this solution.

Figure 7 presents the result of an optimization cycle executed over a 24 h prediction horizon. The dashboard shows the optimal power dispatch obtained for the different generation sources and the expected cost associated with the scenarios considered by the model.

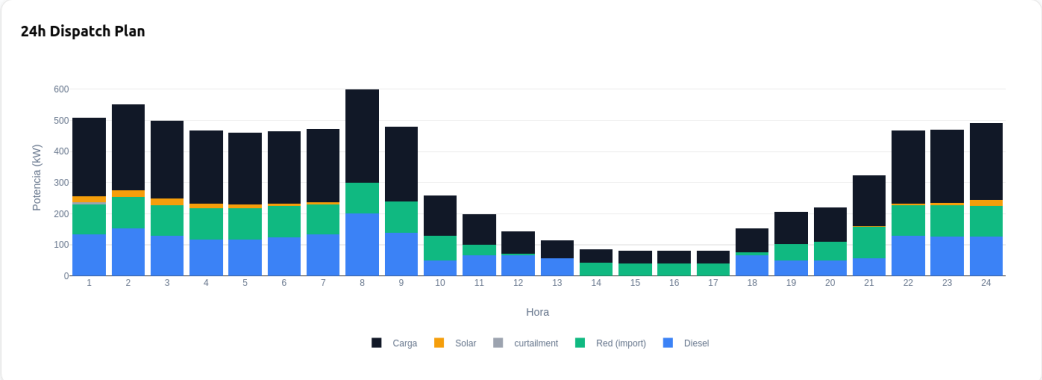


Figure 7. Optimization cycle result using MPC for a 24 h horizon.

The obtained dispatch plan satisfies the projected demand by distributing the power among the available resources according to the model's operational constraints. Figure 8 summarizes the main indicators generated after each execution of the MPC cycle.

The computed indicators include the total operating cost, the maximum dispatched power, and the optimization status. These metrics enable continuous monitoring of system behavior across successive receding-horizon iterations of the MPC. These results characterize the optimization layer; however, its performance depends on the availability of accurate forecasts for the energy variables considered by the model.



Figure 8. Indicators obtained after the execution of the optimization cycle.

Closed-Loop Comparative Evaluation

To quantify the economic value of the optimization-based dispatch, a 14-day closed-loop simulation was executed (18–31 July 2026) with an hourly receding horizon: at each hour, the MPC is solved with a 24 h lookahead and only the first action of the base scenario is implemented; the state then advances with the realized values, namely the demand and photovoltaic output retrieved from the system's physical energy meters and persisted in MongoDB (collections `pasto_load` and `pasto_solar_pv`), and the site weather taken as the ERA5 reanalysis record for the corresponding timestamps. The solar-power bands that drive the optimization are produced by the PatchTST chain described in Section 2, whose residual-correction stage currently relies on synthetic residuals; the demand and PV trajectories used to advance the loop, however, are real measurements from the microgrid instrumentation. An initial SOC of 0.65, identical for all strategies, is used, and the state of charge is propagated between hours from the realized closed-loop state. Four strategies were compared: S-MPC (stochastic, $S = 3$ scenarios), D-MPC (deterministic, $S = 1$, P50 only), HEUR (priority-list heuristic: PV first, then battery within its SOC limits, then diesel and grid), and MPC-PI (measured-data benchmark MPC: the deterministic controller fed with the realized series, i.e., the theoretical upper bound). Time-of-use tariffs (1 USD = 4000 COP) were valley (00–05) 0.0113 USD/kWh, mid-peak (06–18, 22–23) 0.0200 USD/kWh, and peak (19–21) 0.0350 USD/kWh, plus a fixed term of 0.0100 USD/h; diesel consumption was $(0.5 + 0.5P + 0.001P^2)$ L/h at $C_{\text{comb}} = 0.0250$ USD/L. Table 2 summarizes the results, and Figure 9 shows the cumulative net cost over the period.

Table 2. Closed-loop comparative evaluation over 14 days (18–31 July 2026). Costs in USD (1 USD = 4000 COP).

Strategy	Net Cost, 14 Days (USD)	Renewables (%)	Battery Cycles/Day	Grid Import (kWh)	Diesel (L)
S-MPC (stochastic, $S = 3$)	−357.18	61.2	0.51	3	2353
D-MPC (deterministic, $S = 1$)	−357.18	61.2	0.51	3	2353
HEUR (priority list)	−29.48	61.2	0.04	35	12
MPC-PI (measured-data benchmark)	−358.15	61.2	0.51	3	2361

The net cost is negative because the microgrid is a net exporter (PV generation of 184 kWh/day against a 90 kWh/day load); the surplus is sold at the variable tariff of each period, and the battery performs valley-to-peak arbitrage (charging at 0.0113 USD/kWh and discharging at 0.0350 USD/kWh), with the SOC traversing its operational range [0.2, 0.95]

without violations. Three observations follow. First, the optimization-based strategies achieved a net operating result approximately 12 times larger in magnitude than the priority-list heuristic (−357 vs. −29 USD over the 14 days, at 1 USD = 4000 COP), which neither exploits the battery arbitrage nor the peak-hour export and imports 35 kWh/day from the grid. Second, the stochastic and deterministic MPC variants produce nearly identical results (0.00% difference): because only the first action of the base scenario (P50) is implemented and the three quantile-anchored scenarios share that base trajectory, the implemented first step coincides with the deterministic one in this net-exporting configuration; the stochastic advantage, if any, materializes in the expected cost of the lookahead rather than in the executed action. Third, the measured-data benchmark controller (MPC-PI) achieves −358 USD and dominates all forecast-based strategies as expected, with S-MPC operating within 0.27% of this theoretical reference, which provides quantitative evidence that, for this configuration, the economic value comes primarily from operation (arbitrage and peak-hour export) rather than from forecast accuracy. Balance violations were zero in all strategies.

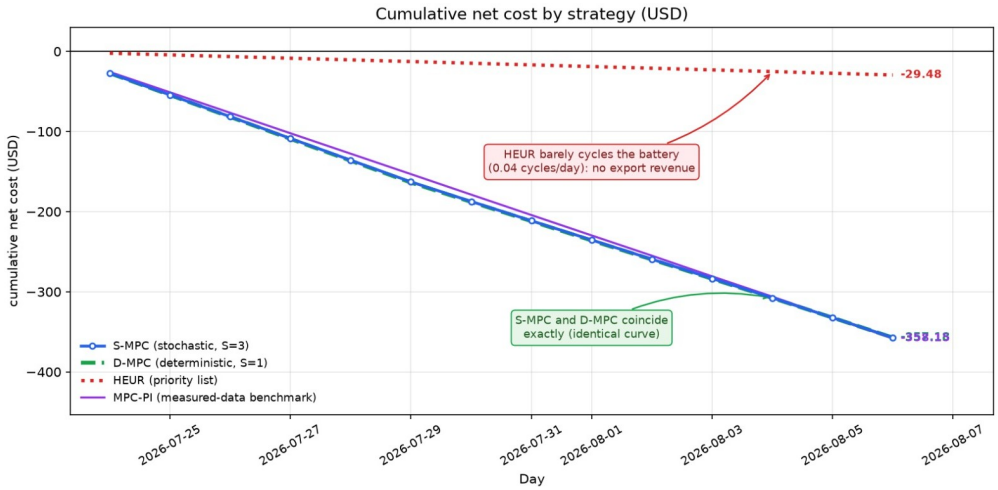


Figure 9. Cumulative net operating cost over the 14-day closed-loop evaluation for the four strategies (in USD).

Figure 10 presents the behavior of the PatchTST weather forecaster during validation against the Open-Meteo reference, showing the predicted and reference global horizontal irradiance (GHI) together with the P10–P90 uncertainty band produced by the model.

The forecasts reproduce the dominant trends of the analyzed variables, with most observations falling within the model's estimated uncertainty intervals. The quantitative evaluation over a 200-window test set (72 h horizon) is summarized in Table 3. For global horizontal irradiance, PatchTST achieves a MAE of 53.9 W/m^2 (nMAE 13.0%), a skill score of +91.2% over the persistence baseline (MAE 206.5 W/m^2) and slightly better than the hourly climatology baseline (54.9 W/m^2), an R^2 of 0.864, and an empirical P10–P90 coverage of 78.0%, close to the nominal 80%—the property that supports the use of these quantiles as MPC scenario anchors (Section 2.1). The decomposition and temperature variables show the same pattern, with near-nominal coverage for diffuse radiation (80.4%) and temperature (69.2%), indicating that the quantile head captures the dominant variability of the radiation–temperature response.

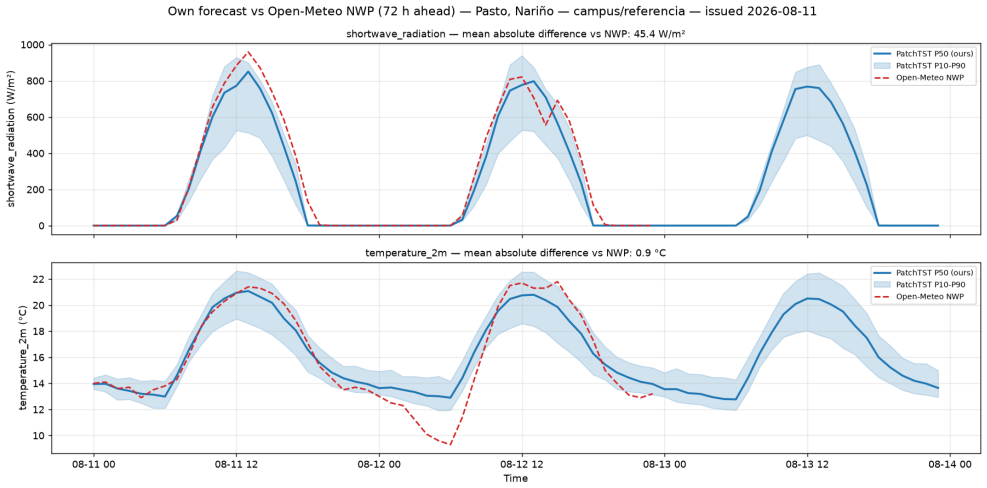


Figure 10. Comparison between the PatchTST probabilistic forecast (P50 with P10–P90 band) and the Open-Meteo reference for global horizontal irradiance (GHI).

Table 3. PatchTST test-set metrics (72 h horizon, 200 windows) against baselines.

Variable	MAE	RMSE	nMAE (%)	Skill vs. Pers. (%)	P10–P90 Cov. (%)
GHI (W/m^2)	53.9	103.8	13.0	+91.2	78.0
Direct normal irradiance (W/m^2)	90.4	162.9	25.0	+89.8	77.9
Diffuse radiation (W/m^2)	21.8	42.3	12.6	+90.8	80.4
Temperature (°C)	0.9	1.2	6.7	+99.2	69.2

Beyond the evaluation of the optimization model and the forecaster, the functional integration of the constituent SIGE components was verified.

The platform continuously ingests IoT meter data via MQTT, persists measurements in MongoDB Atlas, executes the optimization routines, and updates the operator interface via WebSockets. In addition, the system incorporates modules for real-time monitoring, dynamic dashboard generation, and operator assistance through an LLM-based multi-agent system.

To substantiate the real-time operation claims, this section reports the system performance metrics measured on the deployment environment. A reproducible benchmark suite measures the end-to-end latency and throughput of each communication and computation stage; the hardware and software stack used for the measurements are declared so that the results can be reproduced. Table 4 summarizes the measured values.

Table 4. System performance metrics (measured on the deployment environment; 120 MPC cycles; 7000 MQTT messages; 50 concurrent REST connections; hardware: 4-core CPU, 5.6 GB RAM, Python 3.10.12, Pyomo 6.10.0, Gurobi 13.0.2, HiGHS 1.15.1 fallback).

Subsystem	Metric	p50	p95	p99
MPC solver	model build, 24 h × 3 scenarios (s)	0.205	0.317	0.384
MPC solver	solve, MILP (Gurobi) (s)	0.265	0.373	0.406
MPC solver	total cycle $t_{\text{build}} + t_{\text{solve}}$ (s)	0.494	0.607	0.695
MQTT (end-to-end)	publish → receive latency (ms)	243	348	509
MongoDB (insert)	write latency (ms)	97	137	950
WebSocket (push)	backend emission (ms)	0.11	0.38	0.88
WebSocket (push)	end-to-end, client (ms)	342	503	705
REST API	latency under load, 50 conn. (ms)	330	451	669

The MPC timing was measured over 120 complete build-and-solve cycles of the production model (24 h horizon, three scenarios, 648+ binaries), all terminated with status `optimal`. The total cycle solves in 0.494 s (p50) and 0.695 s (p99)—a margin of approximately 1482× with respect to the 900 s (15 min) control interval, confirming that the solver stage is not the bottleneck of the closed loop. The deployed model is a mixed-integer linear program: the quadratic diesel cost is linearized by segments so that it runs on the community (size-limited) Gurobi license and on the open-source HiGHS fallback; the quadratic cost itself is evaluated in the closed-loop objective. For the communication stages, 7000 MQTT messages were published at a sustained 10 msg/s with zero errors, giving an end-to-end publish-to-receive latency of 243 ms (p50) and 509 ms (p99) over a remote broker; the MongoDB (Atlas) insert path adds 97 ms (p50); the backend Socket.IO emission is submillisecond (0.11 ms p50) while the end-to-end client latency (342 ms p50) is dominated by the remote database insert and network. The REST API sustains 145 requests/s under 50 concurrent connections with zero HTTP errors (330 ms p50 under load; 179 ms p50 without load). Resource consumption during 10 min of sustained load (5 s sampling): backend Node.js service 3.5% CPU/113 MB RAM, forecasting service (FastAPI + PatchTST) 0.4% CPU/1.19 GB RAM, Redis 0.9% CPU/11 MB RAM.

The integrated operation of these components enabled the complete data-acquisition, processing, optimization, and visualization cycle within the proposed architecture, thereby verifying inter-service communication.

4. Discussion

The obtained results demonstrate the viability of the proposed architecture for integrating data acquisition, forecasting, optimization, and operator assistance into a unified platform. Rather than evaluating each component in isolation, the performed validation demonstrates the system's capacity to coordinate their operation through a decoupled, service-oriented architecture.

A primary contribution of the platform is the modular design of the forecasting subsystem. The implementation of an abstract interface, together with a FastAPI-based service, decouples the forecasting engine from the remainder of the architecture, enabling the incorporation or substitution of alternative models without modifying the backend orchestration flow. This design choice facilitates platform evolution as new algorithms or more representative datasets become available.

Within this architecture, the PatchTST weather-forecasting module provides the probabilistic input to a grey-box calibrated PV physics model; together, they constitute the implementation that validates the incorporation of advanced forecasting models within the platform. Because PatchTST was trained on six years of ERA5 reanalysis data for Pasto and the PV calibration currently relies on synthetic data with a physical ground truth, the present validation focuses on demonstrating architectural readiness for integration rather than exhaustive predictive-performance evaluation against real sensor data; the closed-loop dispatch itself, however, is evaluated over real demand and PV measurements from the microgrid instrumentation.

In the closed-loop experiment reported in Section Closed-Loop Comparative Evaluation, the MPC received its solar-power scenarios from the PatchTST-based forecasting chain described in Section 2, which is also the provider configured in the production environment (`FORECASTER = patchtst`, `MPC_INTERVAL_MINUTES = 15`); the Open-Meteo NWP service acts as an external reference and the deterministic MatlabPredictor as a fallback baseline. Thanks to the modular design implemented via the prediction API, providers can be substituted without modifying the rest of the system's architecture.

https://doi.org/10.3390/engproc2026153029

Finally, the LLM-based multi-agent system processed the outputs of the statistical anomaly detector and generated structured natural-language responses. However, verification of these diagnostics was limited to structural compliance with Zod schemas and did not include a quantitative evaluation of recommendation accuracy (e.g., false-positive rate, fault coverage, or response latency), nor an assessment of perceived utility by human operators. Validating the system's effectiveness as an assistance tool therefore requires a controlled operator study contrasting actions taken with and without AI-agent support.

5. Conclusions

This work presented the development and validation of SIGE, a modular platform for intelligent microgrid management that integrates IoT monitoring, MPC-based optimization, and operator-assistance tools within a decoupled and extensible architecture. The platform demonstrates the feasibility of hybrid-microgrid dispatch through MPC cycles that combine mathematical optimization with real-time intelligent analysis, with the closed-loop evaluation performed over real microgrid measurements.

The platform processes real-time IoT sensor streams, updates operator visualizations with low latency, and automatically generates optimization-model constraints from the topology defined in the single-line diagram editor. The communication architecture between the service orchestrator and the optimization engine, which is based on job queues with periodic polling, completed forecasting and dispatch cycles at 15-min intervals, indicating applicability to operational environments with intra-hour response requirements.

Future work will propose closing the calibration loop of the grey-box PV model with real IoT sensor data, replacing the synthetic residuals with measured generation so that the PatchTST-driven power bands reflect the actual plant; extending the scenario set beyond the three quantile-anchored scenarios (e.g., Monte Carlo sampling from the empirical forecast-error distribution) and evaluating a measured-data benchmark benchmark over the full scenario tree; extending the same two-level calibration scheme to the wind and battery (BESS) assets; and evaluating the platform using larger datasets and different microgrid configurations, as well as incorporating new prediction models taking advantage of the implemented modular design.

Author Contributions: Conceptualization, D.D. and J.S.; methodology, D.D., J.S. and J.R.-F.; software, D.D. and J.S.; validation, D.D. and J.S.; formal analysis, D.D., J.S. and J.R.-F.; investigation, D.D. and J.S.; resources, J.R.-F.; data curation, D.D. and J.S.; writing—original draft preparation, D.D. and J.S.; writing—review and editing, D.D., J.S. and J.R.-F.; visualization, D.D. and J.S.; supervision, J.R.-F.; project administration, J.R.-F.; funding acquisition, J.R.-F. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by Vicerrectoría de Investigaciones e Interacción Social (VIIS) of Universidad de Nariño, grant number Project ID-3214 (Agreement No. 29, dated 17 January 2025).

Data Availability Statement: The source codes and data of the PatchTST forecaster and the PV calibration pipeline will be structured as closed institutional repositories pending formal scientific licensing guidelines.

Acknowledgments: The authors acknowledge the support of the research project "Desarrollo de una solución tecnológica para el análisis de flujos de potencia en redes eléctricas de distribución desbalanceadas con generación distribuida" ID-3214, supported by Agreement No. 29, dated 17 January 2025, by VIIS of the Universidad de Nariño. We also thank the Department of Electronic Engineering and the Electrical and Electronic Engineering Research Group (GIIEE) at the Universidad de Nariño for their support of the academic and administrative processes necessary to carry out this project.

Conflicts of Interest: The authors declare no conflicts of interest related to this scientific development.

References

1. Pantoja, A.; Fajardo, D.; Revelo, J. ALTERNAR: Análisis de Oportunidades Energéticas con Fuentes Alternativas en el Departamento de Nariño. In *Premio AMBAR 2016*; ASOCODIS: Bogotá, Colombia, 2016. Available online: https://docs.upme.gov.co/SIMEC/PERS/Nari%C3%B1o/Proyectos/AnalisisOportunidadesEnergeticas%20(1).pdf (accessed on 5 July 2026).
2. Carrasco, F.; Kristjanpoller, F.; Verdugo, I.; Viveros, P.; Pascual, R. Forecast-informed daily energy dispatch for cost and emission reduction: Evidence from an emerging economy power system. *Energy Convers. Manag.* **2026**, *364*, 121611. [CrossRef]
3. Lim, B.; Arik, S.O.; Loeff, N.; Pfister, T. Temporal Fusion Transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **2021**, *37*, 1748–1764. [CrossRef]
4. Rao, C.K.; Sahoo, S.K.; Yanine, F.F. A literature review on an IoT-based intelligent smart energy management systems for PV power generation. *Hybrid Adv.* **2024**, *5*, 100136. [CrossRef]
5. Tasmant, H.; Bossoufi, B.; Alaoui, C.; Siano, P. A review of machine learning and IoT-based energy management systems for AC microgrids. *Comput. Electr. Eng.* **2025**, *127*, 110563. [CrossRef]
6. Nie, Y.; Nguyen, N.H.; Sinthong, P.; Kalagnanam, J. A Time Series is Worth 64 Words: Long-term Forecasting with Transformers. *arXiv* **2023**, arXiv:2211.14730.
7. Birge, J.R.; Louveaux, F. *Introduction to Stochastic Programming*, 2nd ed.; Springer: Berlin/Heidelberg, Germany, 2011. [CrossRef]
8. Parisio, A.; Rinaldi, E.; Glielmo, L. Model predictive control of microgrids. *IEEE Trans. Control Syst. Technol.* **2014**, *22*, 1622–1628. [CrossRef]
9. Su, W.; Wang, J.; Roh, J. Stochastic energy management for a microgrid with renewable energy resources and energy storage systems. *IEEE Trans. Smart Grid* **2014**, *5*, 1747–1755.
10. Zia, M.F.; Elbouchikhi, E.; Benbouzid, M. Microgrids energy management systems: A critical review on methods, solutions, and prospects. *Appl. Energy* **2018**, *222*, 1033–1055. [CrossRef]
11. Hua, H.; Qin, Y.; Hao, C.; Cao, J. Optimal energy management strategies for energy Internet via deep reinforcement learning approach. *Appl. Energy* **2019**, *239*, 598–609. [CrossRef]
12. Rawlings, J.B.; Mayne, D.Q.; Diehl, M.M. *Model Predictive Control: Theory, Computation, and Design*, 2nd ed.; Nob Hill Publishing: Madison, WI, USA, 2017; ISBN 978-0-9759377-3-0.
13. Bynum, M.L.; Hackebeil, G.A.; Hart, W.E.; Laird, C.D.; Nicholson, B.L.; Siirola, J.D.; Watson, J.P.; Woodruff, D.L. *Pyomo—Optimization Modeling in Python*, 3rd ed.; Springer: Berlin/Heidelberg, Germany, 2021; ISBN 978-3-030-68928-3.
14. Gurobi Optimization, LLC. *Gurobi Optimizer Reference Manual*, version 13.0.2; Gurobi Optimization, LLC: Beaverton, OR, USA, 2024.
15. Huangfu, Q.; Hall, J.A.J. Parallelizing the dual revised simplex method. *Math. Program. Comput.* **2018**, *10*, 119–142. [CrossRef]
16. Chase, H. LangChain: Building Applications with LLMs Through Composability. 2022. Available online: https://github.com/langchain-ai/langchain (accessed on 5 July 2026).
17. Lami, B.; Alsolami, M.; Alferidi, A.; Slama, S.B. A Smart Microgrid Platform Integrating AI and Deep Reinforcement Learning for Sustainable Energy Management. *Energies* **2025**, *18*, 1157. [CrossRef]
18. Han, W.; Cui, J.; Wang, X.; Chen, X. Resilient optimal dispatch of ship-integrated energy system and air lubrication using an enhanced traffic jam optimizer. *J. Mar. Sci. Eng.* **2026**, *14*, 779. [CrossRef]
19. Arciniegas, A.F.; Imbajoa, D.E.; Revelo, J. Diseño e implementación de un Sistema de Medición Inteligente para AMI de la microrred de la Universidad de Nariño. *Enfoque UTE* **2017**, *8*, 300–314. Available online: https://ingenieria.ute.edu.ec/index.php/revista/article/view/136 (accessed on 5 July 2026). [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.