engineering proceedings

MDPI

Proceeding Paper

Open-RPS: An Open-Source Platform for Three-Phase Power Flow Analysis in Electric Power Systems †

Andrés Miramag *, Santiago Ojeda *, Didier Villaquirán and Javier Revelo-Fuelagán

Department of Electronics Engineering, Universidad de Nariño, Pasto 520001, Colombia; didier.v@udenar.edu.co (D.V.); javierrevelof@udenar.edu.co (J.R.-F.)

* Correspondence: miramagandres@udenar.edu.co (A.M.); santiagoandres@udenar.edu.co (S.O.)

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

Abstract

Open-RPS v1.0 (radial power solver) is an open-source software platform for three-phase power flow analysis in balanced and unbalanced radial distribution systems. The platform implements the backward/forward sweep (BFS) method in Python through an object-oriented, modular architecture designed for transparency, auditability, and reproducibility. Numerical validation is performed against the IEEE 4-bus, 13-bus, and 123-bus test feeders. The IEEE 123-bus case resolves 132 nodes, including four voltage regulators and four capacitor banks; the complete voltage-magnitude comparison covers 179 non-zero phase values, with a mean absolute error of 0.4857% and a maximum error of 1.4500%. The IEEE 4-bus and 13-bus cases are additionally validated against GridLAB-D, confirming close agreement with an independent reference implementation. These results support Open-RPS as a transparent, license-free, and reproducible platform for teaching, algorithmic validation, and research on radial distribution feeders within the documented validation scope.

Keywords: power flow; forward/backward sweep; IEEE test feeders; Python; open-source; radial distribution networks; reproducible scientific software

1. Introduction

Power flow analysis in electrical distribution networks is a fundamental task for evaluating voltage profiles, technical losses, operating conditions, and the effects associated with incorporating distributed energy resources [1]. Unlike transmission systems, distribution networks present predominantly radial topologies, unbalanced single- and three-phase loads, high R/X ratios, distribution transformers, voltage regulators, shunt capacitor banks, and laterals with varying phase configurations. These characteristics reduce the efficiency of classical methods formulated for meshed networks and justify the use of specialized algorithms. Among these methods, the backward/forward sweep (BFS) has consolidated itself as one of the most widely used alternatives due to its computational efficiency and suitability for the analysis of radial feeders [1–3].

As Kersting explains, conventional iterative techniques commonly used in transmission network power-flow studies exhibit poor convergence characteristics when applied to radial distribution feeders, which is why an iterative technique specifically designed for radial systems is required [1]. This technique, referred to as the ladder iterative technique, is derived from a modification of the "ladder" network theory of linear systems, adapted to account for the nonlinear behavior introduced by constant-power (kW, kvar) loads [1].

Check for updates

Academic Editor: Fredy Gavilanes

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.

The method exploits the radial structure of the feeder by performing two alternating sweeps instead of solving the complete system of nonlinear equations simultaneously, as required by Newton–Raphson-based formulations. In the backward sweep, starting from the end nodes of the feeder and moving toward the source, the load and line currents are computed at each node by applying Kirchhoff's current law (KCL). In the forward sweep, starting from the source node and moving toward the end nodes, the line currents obtained in the previous stage are used to update the node voltages by applying Kirchhoff's voltage law (KVL), accounting for the voltage drop across each branch impedance [1]. These two sweeps are repeated iteratively until the voltage mismatch at every node and phase, relative to the nominal line-to-neutral voltage, falls below a specified convergence tolerance [1]. This structure allows the BFS method to naturally accommodate the unbalanced, three-phase nature of distribution feeders, since each sweep can be formulated directly in terms of phase impedance and current matrices, without requiring the admittance matrix inversion typical of conventional load-flow formulations.

1.1. State of the Art of Computational Tools

In the literature, various software platforms exist for the analysis of electric power systems. MATPOWER [4] is an open-source tool oriented towards steady-state power flow, optimal power flow (OPF), and operational analysis in transmission systems; its main formulation employs the Newton–Raphson method for large-scale networks, and although it can run on GNU Octave, its native environment is MATLAB, which introduces licensing dependencies in academic contexts. OpenDSS [5], developed by EPRI, is a specialized simulation engine for distribution networks that allows modeling distributed resources, variable loads, harmonics, and quasi-static time-series (QSTS) simulations; however, it operates primarily as a compiled application with a scripting interface (COM/Python), which limits the direct inspection of its internal algebraic structures. GridLAB-D [6] is oriented towards smart grid simulation using an agent-based approach, with multi-domain coupling, such as buildings, markets, and transportation. However, the complexity of its architecture and its orientation toward multi-purpose simulations make it less suitable for basic algorithmic validation of the BFS method. In this work, GridLAB-D is used as an external benchmark tool to evaluate the consistency of the results obtained with Open-RPS against a specialized distribution network simulation platform. pandapower [7] is an open platform in Python with direct integration into the data science ecosystem, which supports power flow, OPF, and contingency analysis in transmission and distribution networks; however, it does not support the development of unbalanced electric power systems.

Table 1 synthesizes the main characteristics of these tools in relation to the objective of this article.

From the above review, it is clear that in teaching, reproducible research, and algorithmic validation of the BFS method, there remains a persistent need for lightweight, auditable platforms specifically oriented towards the three-phase analysis of radial feeders. These platforms must allow explicit verification against standardized benchmarks and, when possible, an additional comparison with specialized simulation tools, such as GridLAB-D.

Table 1. Reference tools for electrical systems analysis and their relationship with the Open-RPS approach.

Tool	Language/Environment	Main Focus	Relative Limitation
MATPOWER [4]	MATLAB/Octave	Power flow and OPF in transmission	MATLAB dependency; no unbalanced three-phase analysis

Table 1. *Cont.*

Tool	Language/Environment	Main Focus	Relative Limitation
OpenDSS [5]	Delphi (COM/Python)	Detailed distribution and distributed resources	Compiled black-box engine; limited inspection of internal algebraic structures
GridLAB-D [6]	C/C++ with modules	Agent-based smart grid simulation	Steep learning curve; complex multi-domain architecture
pandapower [7]	Python	General electrical system modeling	No unbalanced three-phase analysis
Open-RPS (this work)	Python	Three-phase BFS power flow in radial feeders	Currently limited to radial topologies

1.2. Contributions of the Article

In this context, Open-RPS v1.0 occupies a complementary niche to that of existing tools: it offers an open, traceable, and extensible implementation of the BFS algorithm for radial distribution networks, with a modular structure oriented towards reproducible validation. The main contributions are as follows:

- Modular implementation of the three-phase BFS method in Python, organized into clearly delimited functional layers: electrical components, topology, solver, reports, and utilities.
- Description of the software architecture and design decisions that favor the reproducibility of the computational environment.
- Quantitative validation of results against the IEEE 4-bus, 13-bus, and 123-bus feeders, with direct comparison of voltage magnitudes and phase angles, branch currents, and total losses.
- Comparison of the results obtained with simulations carried out in GridLAB-D, with the purpose of contrasting the numerical consistency of Open-RPS against a specialized distribution network tool.
- Identification of an extensible base for future studies on distributed resource integration, optimization, and quasi-static time-series analysis.

This article is organized as follows. Section 2 describes the materials and methods, including the architecture of Open-RPS v1.0, the formulation of the BFS algorithm, and the validation cases employed. Section 3 presents the results obtained for the IEEE 4-bus and 13-bus systems, including the comparison with the IEEE reference values and with GridLAB-D. Section 4 discusses the scope of the results, numerical precision, the advantages of the proposed architecture, and the current limitations of the platform. Finally, Section 5 presents the main conclusions and future lines of work.

2. Materials and Methods

Open-RPS originates from a migration of RPS, a toolbox originally developed in MATLAB [8]. From this tool, it inherits its internal structure and object-oriented programming (OOP) logic.

2.1. Architecture of Open-RPS v1.0

Open-RPS v1.0 was structured as a software package in Python 3.13.5 (compatible with later 3.x releases). The platform organizes its logic into three main functional layers within the root package `rps/`, as illustrated in Figure 1.

The three functional layers in which the platform's code modules are distributed are described in detail below:

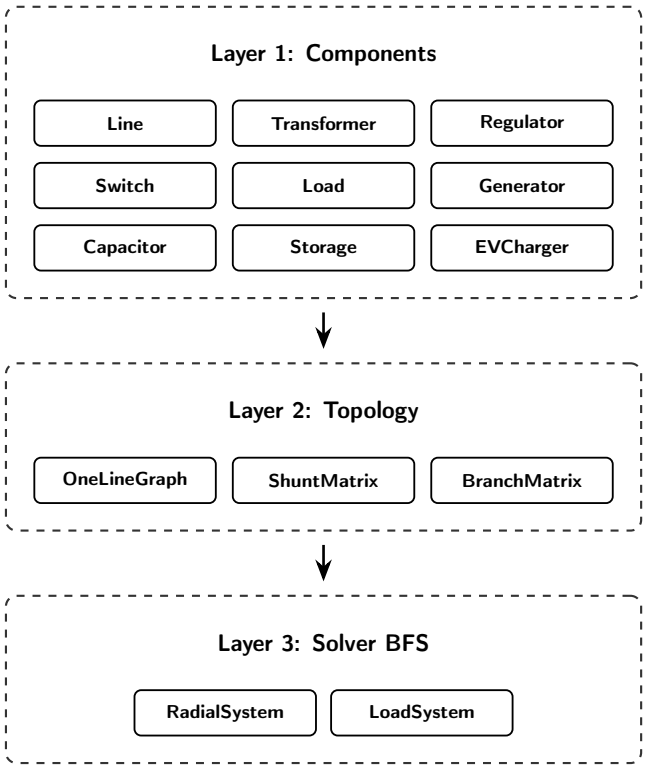


Figure 1. General structure of the modular architecture of Open-RPS.

2.1.1. Layer 1: Components (`rps/components/`)

This layer manages the physical modeling and governing equations of the electrical components that make up the radial distribution network. The Distribution Line (`line.py`) models sequence impedances and admittances. The Transformer (`transformer.py`) models electromagnetic couplings for 12 different configurations (Yg-Yg, D-D, Y-D). The ZIP Load (`load.py`) represents unbalanced three-phase electrical loads using the ZIP model. The layer also includes voltage regulators (`regulator.py`), shunt capacitors (`capacitor.py`), network switches (`switch.py`), PQ and PV type generators (`generator.py`), battery energy storage (`storage.py`), and profile-based electric vehicle chargers (`ev_charger.py`), which are fundamental elements in the real operation of an EPS.

2.1.2. Layer 2: Topology (`rps/topology/`)

Layer 2 is responsible for the spatial ordering and interconnection analysis of the electrical system. Its functions lie in processing the circuit topology and organizing the generalized matrices of the branch elements according to the EPS connections:

- `BranchMatrix`**:** Reads branch parametrization and automatically builds the generalized matrices of all branch elements for the application of the BFS method for power flow calculations.
- `ShuntMatrix`**:** Compiles and groups all shunt injection currents from passive elements and power injections from the network nodes.
- `OneLineGraph`**:** Builds and models the system as a directed topological graph using the specialized library `NetworkX` [9]. This class executes a radial topological ordering that automatically identifies the root node (supply substation) and allows efficient ancestor and descendant searches through a breadth-first search (BFS) traversal, optimizing the sequencing of power calculations.

2.1.3. Layer 3: BFS Solver (`rps/solvers/`)

It consists of the computational and processing core of the tool, where the mathematical solvers of the system reside:

- `RadialSystem`**:** Performs the three-phase unbalanced power flow calculation using the backward/forward sweep (BFS) method by current summation, working with dictionary-type data structures to represent the network topology. In the backward sweep, it recursively calculates branch currents from the endpoints toward the substation, and in the forward sweep, it propagates phase voltages using the generalized matrices defined in Layer 2. Additionally, this solver performs PV node compensation using the reactive current injection method.
- `LoadSystem`**:** In the input phase, it coordinates data reading, initializes load profiles, and manages the global iterative simulation process, also applying the BFS method by current summation. Unlike `RadialSystem`, it processes the equations using a purely matrix-based approach supported by NumPy [10] and SciPy [11]. Furthermore, it implements PV node compensation based on the positive-sequence admittance matrix, automatically structuring the results into tabular reports.

2.2. BFS Algorithm

The solver implements the backward/forward sweep (BFS) method for three-phase unbalanced power flow calculations in radial networks [1].

The iterative procedure for each iteration k comprises the following sequential unbalanced three-phase analysis:

1. **Initialization and normalization:** Reading of electrical data and normalization to per-unit (p.u.) values using the corresponding line-to-neutral voltage (V_{base}) and three-phase power (S_{base}) bases. The voltages at all nodes are initialized in a flat profile starting from the corrected source voltage:

$$[VLN_{abc}]_m^{(0)} = [VLN_{abc}]_{\text{source}} \quad \forall m \in \mathcal{N} \tag{1}$$

   where $\mathcal{N}$ represents the set of nodes in the system.
2. **Backward Sweep:** Based on the voltages from the previous iteration $[VLN_{abc}]_m^{(k)}$, the currents injected or consumed by the shunt elements (loads, generators, capacitors) at each node m are calculated.
   Subsequently, the graph is traversed in reverse order (from the terminal leaf nodes toward the source). For each branch element connecting an upstream node n with a downstream node m, the current at the receiving end $[I_{abc}]_m^{(k)}$ is accumulated by summing the currents of the child branches and the local shunt injection at node m. Then, the current at the sending end $[I_{abc}]_n^{(k)}$ is calculated by applying the generalized matrices $[c]$ and $[d]$ of the element:

$$[I_{abc}]_n^{(k)} = [c]\,[VLN_{abc}]_m^{(k)} + [d]\,[I_{abc}]_m^{(k)} \tag{2}$$

3. **Forward Sweep:** The update of the line-to-neutral phase voltages is performed by traversing the graph in the direct path (from the source downstream). For each branch, the downstream node voltage $[VLN_{abc}]_m^{(k)}$ is updated as a function of the upstream node voltage $[VLN_{abc}]_n^{(k)}$ and the receiving end current calculated in the backward sweep $[I_{abc}]_m^{(k)}$, according to

$$[VLN_{abc}]_m^{(k)} = [A]\,[VLN_{abc}]_n^{(k)} - [B]\,[I_{abc}]_m^{(k)} \tag{3}$$

https://doi.org/10.3390/engproc2026153022

where $[A]$ and $[B]$ are the inverse transmission matrices characterizing the voltage drop across the section. In simplified line sections with negligible capacitance ($[\gamma_{abc}] \approx \mathbf{0}$), the matrices simplify to $[A] = \mathbf{I}$ (identity matrix) and $[B] = [Z_{abc}]$, reducing to the series voltage drop:

$$[VLN_{abc}]_m^{(k)} = [VLN_{abc}]_n^{(k)} - [Z_{abc}]\,[I_{abc}]_m^{(k)} \tag{4}$$

4. **Convergence Check:** The iterative process is repeated by incrementing the index k until the stopping tolerance defined by the error in the node voltage magnitude between consecutive iterations is met:

$$\|\Delta V\|_\infty = \max_{m \in \mathcal{N}} \left(\left\| [VLN_{abc}]_m^{(k)} - [VLN_{abc}]_m^{(k-1)} \right\|_\infty \right) < \varepsilon \tag{5}$$

where the tolerance is set to $\varepsilon = 10^{-5}$ p.u. and with a limit of 100 iterations.

Missing phases at each node are identified using the phase configuration field of each element and are omitted from the system of equations. Loads are represented using the constant power (PQ) model. The active branch losses are calculated as

$$P_{\text{loss},km} = \text{Re}\{V_k I_{km}^*\} - \text{Re}\{V_m I_{km}^*\} \tag{6}$$

2.3. Validation Cases

Numerical validation was performed by comparing the results obtained with Open-RPS against the reference values of the test feeders from the IEEE PES Distribution System Analysis Subcommittee [12]. Complementarily, the solutions obtained with Open-RPS were compared with results generated in GridLAB-D to incorporate an additional computational benchmark in the validation process.

The IEEE 4-bus, 13-bus, and 123-bus systems were evaluated, which allow verifying the performance of the solver under representative conditions of balanced and unbalanced three-phase radial networks and a larger radial feeder.

- **IEEE 4-bus:** This feeder corresponds to a minimal system with a step-down transformer (12.47/4.16 kV) and an unbalanced three-phase load. This case enables evaluation of the algorithm's behavior under extreme unbalance conditions and verifying the correct representation of the transformer and line voltage drops per phase.
- **IEEE 13-bus:** This feeder corresponds to an unbalanced radial distribution system that includes a voltage regulator, shunt capacitors, a step-down transformer, and mixed loads of ZIP type in Wye and Delta connections. This case allows evaluating the model's capacity to represent multiple typical elements of distribution networks and analyze their effect on voltage profiles, branch currents, and system losses.
- **IEEE 123-bus:** This larger unbalanced radial feeder includes 132 nodes, voltage regulators, capacitor banks, a transformer, and multiple load branches. It is used to assess the behavior of the implementation beyond the small benchmark systems and to document the current scalability of the platform.

The validation considered nodal voltage magnitudes, branch currents, total active losses, and convergence indicators to evaluate both the numerical precision and the computational performance of the implemented BFS solver.

The error metrics were calculated using the absolute percentage error between the values obtained with the evaluated tool and the reference IEEE values, according to

$$\varepsilon_{\%} = \left| \frac{V_x - V_{\text{IEEE}}}{V_{\text{IEEE}}} \right| \times 100 \tag{7}$$

where V_x represents the voltage magnitude obtained with Open-RPS or GridLAB-D, and V_{IEEE} corresponds to the reference value reported by the IEEE PES Distribution System Analysis Subcommittee.

If V_{IEEE} is very small, the percentage-error expression becomes numerically ill-conditioned because the denominator amplifies small absolute differences. If $V_{\text{IEEE}} = 0$, the percentage error is undefined and is not reported; the comparison must instead use the absolute error $|V_x - V_{\text{IEEE}}|$ (or a clearly stated engineering tolerance). In the present IEEE voltage data, the reported reference values are non-zero, so the percentage metric is applied only to those entries.

3. Results

This section presents the validation results obtained with Open-RPS for the IEEE 4-bus, 13-bus, and 123-bus systems. The analysis includes convergence indicators, voltage magnitudes, branch currents, and active losses compared with the IEEE reference values and, for the 4-bus and 13-bus cases, with results generated in GridLAB-D.

3.1. Global Convergence and Validation Summary

Table 2 presents the convergence summary and the global validation metrics of the evaluated IEEE systems. All three feeders converged within the tolerances established for each reference case.

Table 2. Global validation summary—IEEE test systems.

Feeder	Iter.	Buses	Losses (kW)	V_{min} (p.u.)	V_{max} (p.u.)
IEEE 4	23	4	594.35	0.763	1.000
IEEE 13	7	14	80.53	0.984	1.063
IEEE 123	5	132	100.859	0.9769	1.0437

3.2. IEEE 4-Bus System

The IEEE 4-bus system includes 24 analysis variations, of which 12 correspond to balanced conditions and 12 to unbalanced conditions. In this work, the first case was evaluated, corresponding to an unbalanced three-phase configuration with a step-down transformer (12.47/4.16 kV), Yg-Yg connection, and an ACSR 336,400 26/7 overhead line. This case allows verifying the behavior of the BFS solver under unbalanced conditions and evaluating the accuracy of the calculations for voltages, branch currents, and active losses. Open-RPS converged in 23 iterations and obtained total losses of 594.35 kW.

Figure 2 shows the topological schematic of the IEEE 4-bus feeder used as a validation case.

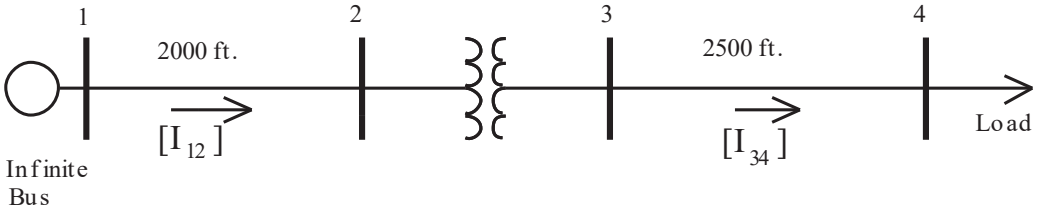


Figure 2. Topological schematic of the IEEE 4-bus feeder.

Table 3 compares the line-to-neutral voltage (V_{LN}) magnitudes obtained with Open-RPS and GridLAB-D against the IEEE reference values for the four buses of the system.

Table 3. Comparison of voltage V_{LN} magnitudes—IEEE 4-bus system (tolerance: 0.5%).

Node	Phase	IEEE 4-Bus Ref. (V)	Open-RPS (V)	RPS Error (%)	GridLAB-D (V)	GridLAB-D Error (%)
n1	A	7199.6	7199.6	0.000	7199.6	0.001
	B	7199.6	7199.6	0.000	7199.6	0.001
	C	7199.6	7199.6	0.000	7199.6	0.001
n2	A	7164.0	7163.7	0.004	7163.7	0.004
	B	7110.0	7110.5	0.007	7110.5	0.007
	C	7082.0	7082.0	0.000	7082.0	0.000
n3	A	2305.0	2305.5	**0.022**	2305.5	**0.021**
	B	2255.0	2254.7	0.013	2254.7	0.015
	C	2203.0	2202.8	0.009	2202.8	0.009
n4	A	2175.0	2175.0	0.000	2175.0	0.001
	B	1930.0	1929.8	0.010	1929.8	0.010
	C	1833.0	1832.7	0.016	1832.7	0.019

Reference: IEEE PES DSAS—Table 3 (Yg-Yg Step-Down, 12.47/4.16 kV). Maximum Open-RPS error: 0.022% at node n3, phase A. Maximum GridLAB-D error: 0.021% at node n3, phase A. All values are within the 0.5% tolerance.

The voltage results show high agreement among Open-RPS, GridLAB-D, and the IEEE reference. The maximum error for Open-RPS was 0.022%, located at node n3, phase A, while the maximum error for GridLAB-D was 0.021% at the same node and phase. Both errors remained below the 0.5% tolerance established for this case.

Figure 3 presents the voltage magnitude profile for phase A, B, and C for the IEEE 4-bus system obtained with Open-RPS.

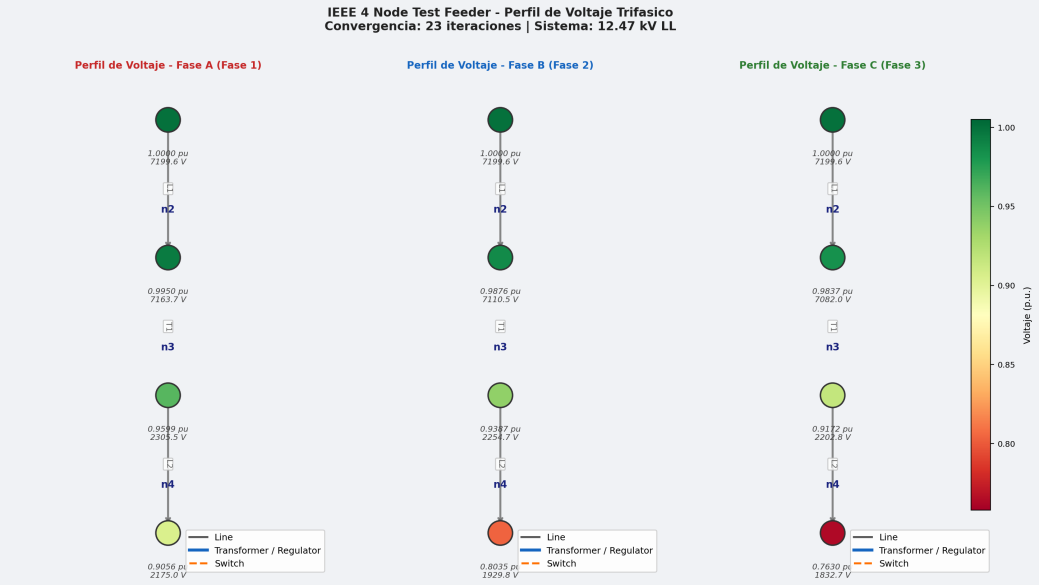


Figure 3. Voltage profile of phases A, B, and C for the IEEE 4-bus system.

The voltage angles presented deviations smaller than 0.05 compared to the IEEE reference values. Table 4 presents the branch currents obtained with Open-RPS and GridLAB-D compared with the official values of the IEEE 4-bus feeder.

Taken together, these results indicate that Open-RPS accurately reproduces the main electrical variables of the IEEE 4-bus system and maintains a level of agreement comparable to that obtained with GridLAB-D.

Table 4. Branch currents—IEEE 4-bus system.

Source	I_a (A) $\angle\theta$	I_b (A) $\angle\theta$	I_c (A) $\angle\theta$
Branch 1–2			
IEEE 4 node feeder	230.1 ∠−35.9°	345.7 ∠−152.6°	455.1 ∠84.7°
Open-RPS	230.07 ∠−35.91°	345.73 ∠−152.64°	455.07 ∠84.65°
GridLAB-D	230.1 ∠−35.9°	345.7 ∠−152.6°	455.1 ∠84.6°
Branch 3–4			
IEEE 4 node feeder	689.7 ∠−35.9°	1036.0 ∠−152.6°	1364.0 ∠84.7°
Open-RPS	689.66 ∠−35.91°	1036.37 ∠−152.64°	1364.13 ∠84.65°
GridLAB-D	689.7 ∠−35.9°	1036.4 ∠−152.6°	1364.1 ∠84.6°

Reference: IEEE PES DSAS—Table 4. Open-RPS and GridLAB-D values show agreement with the official reference in magnitude and angle for both branches.

3.3. IEEE 13-Bus System

The IEEE 13-bus feeder is an unbalanced radial distribution system that includes a voltage regulator (RG60), two shunt capacitors, a step-down transformer (buses 633–634), and 17 loads with mixed ZIP models in Wye and Delta connections (14 buses, 10 lines, 1 transformer, 1 regulator, 2 capacitors). The regulator RG60 taps are [10, 8, 11] for phases A, B, and C [12]. This case allows evaluating the simultaneous representation of multiple control and distribution elements. Open-RPS converged in 7 iterations (17.7 ms) with total losses of 80.53 kW.

Figure 4 shows the single-line diagram of the feeder with the location of the control elements.

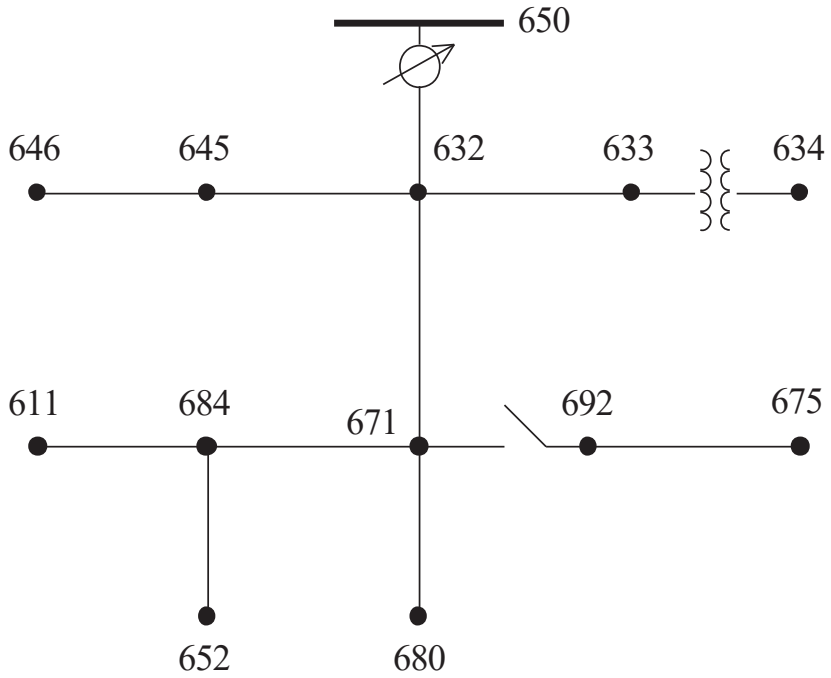


Figure 4. Topological schematic of the IEEE 13-bus feeder.

The complete nodal voltage and branch-current tables for the IEEE 13-bus feeder are provided in Appendix A (Tables A1 and A2) to keep the main results section concise. Entries marked with '—' indicate phases not present in the system. The minimum IEEE reference voltage is 0.9825 p.u. at node 652 (phase A); the minimum Open-RPS voltage is 0.9843 p.u. at the same node and phase. The maximum voltage is 1.0625 p.u. at node 631 (phase A). All values are within the 0.5% tolerance. The maximum voltage error is

recorded at node 652 (phase A) with 0.58%, below the tolerance of 2.0%. Voltages at nodes with regulators (631 and 632) practically coincide with the reference, confirming the correct implementation of the fixed-tap model. Figure 5 illustrates the spatial profile of phase A, showing the progressive drop from node 650 to the terminal branches.

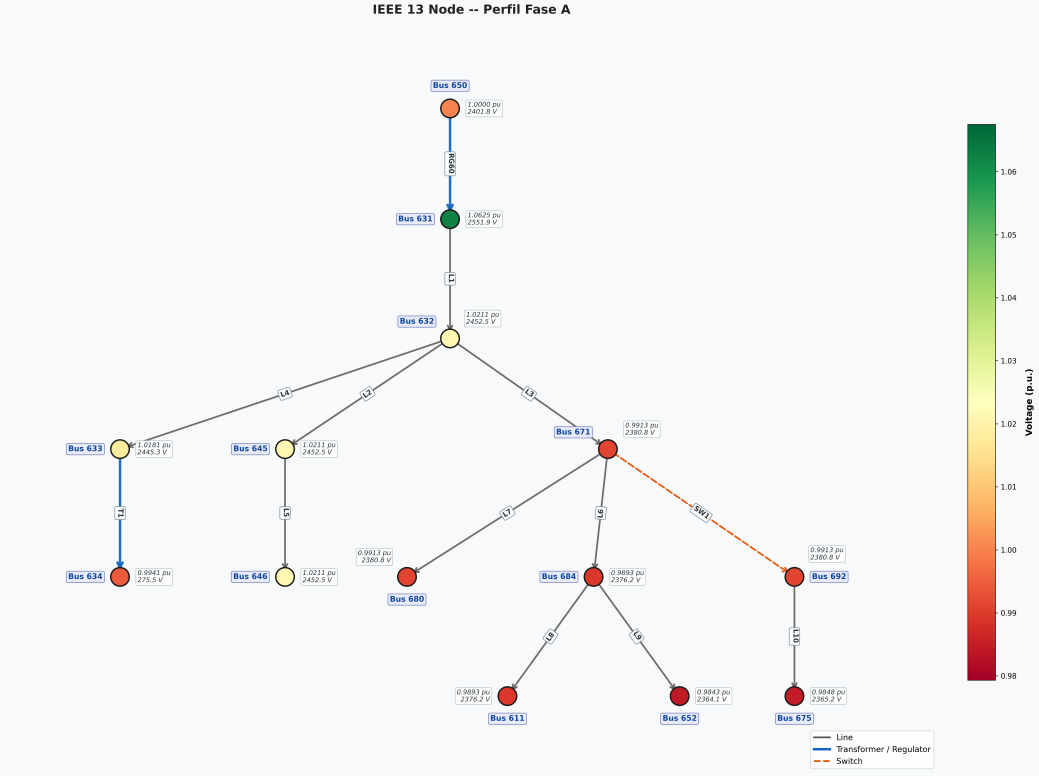


Figure 5. Voltage magnitude profile of phase A—IEEE 13-bus system (Open-RPS).

Figure 5 is read from left to right along the feeder path. The horizontal axis identifies the successive IEEE 13-bus nodes used in the plotted phase-A path, while the vertical axis reports voltage magnitude in per unit (p.u.). The curve therefore represents the spatial voltage profile, not the voltage angle; a downward trend indicates voltage drop along the feeder, and discontinuities or terminal points reflect changes in phase availability or feeder branches.

The full branch-current table (magnitude and angle for every phase, across all 12 branches of the feeder) is provided in Appendix A, Table A2. The currents from Open-RPS and GridLAB-D agree with the IEEE reference in magnitude and angle across all branches. The largest deviations are observed in the terminal single-phase branches (684–611 and 684–652), where the imbalance is most pronounced, but they remain within the tolerances of the case.

Table 5 summarizes the global validation metrics of the 13-bus system.

Table 5. Validation metrics—IEEE 13-bus system.

Parameter	IEEE 13 Node Feeder	Open-RPS	Error (%)
V_{max} (p.u.)	1.0690	1.0625	0.61
V_{min} (p.u.)	0.9900	0.9843	0.58
Total losses (kW)	80.53	80.53	0.00
Iterations	—	7	—
Convergence time (ms)	—	17.7	—

Table 5 confirms exact agreement in total losses (80.53 kW) and voltage errors below 0.61%, validating the joint implementation of the transformer, regulator, and capacitors.

3.4. Computational Performance

Table 6 shows the performance comparison between the platform implemented in MATLAB R2024 (compatible with later releases) and Python.

Table 6. Computational performance comparison: MATLAB (RPS Toolbox) vs. Open-RPS v1.0 (Intel Core i5-1235U, 8 GB RAM).

Feeder	Environment/Software	Tolerance	Iterations	Solution Time (s)	Total Time (s) †
IEEE 4-bus	MATLAB (RPS Toolbox)	$1 \cdot 10^{-5}$	19	–	0.2323
	Open-RPS v1.0	$1 \cdot 10^{-5}$	23	0.0032	0.1547
IEEE 13-bus	MATLAB (RPS Toolbox)	$1 \cdot 10^{-6}$	8	–	1.0314
	Open-RPS v1.0	$1 \cdot 10^{-6}$	7	0.0097	0.2328

† Note: The total time for Open-RPS includes the reading of data from the Excel file using openpyxl and Pandas [13], the network graph preprocessing, and the mathematical resolution. The solution time refers purely to the execution of the BFS solver loop.

3.5. IEEE 123-Bus System

The IEEE 123-bus feeder was added as the larger validation case. The model contains 132 nodes, 126 lines, 91 loads, 1 transformer, 4 voltage regulators, and 4 capacitor banks. Open-RPS converged in five iterations. The complete voltage-magnitude comparison included 179 non-zero phase values and produced a mean absolute error of 0.4857% and a maximum absolute error of 1.4500% at node 610, phase A. Node 610 is evaluated on the 0.48-kV secondary base of transformer XFM1. Table 7 summarizes these validation results.

Table 7. IEEE 123-bus validation results.

Metric	Open-RPS Result	Reference or Criterion
Nodes/lines	132/126	IEEE 123-bus feeder
Loads/transformer	91/1	IEEE 123-bus feeder
Voltage regulators/capacitor banks	4/4	IEEE 123-bus feeder
Convergence iterations	5	Converged
Minimum voltage magnitude	0.9769 p.u. (node 65, phase A)	Reference comparison
Maximum voltage magnitude	1.0437 p.u. (node 150r, phase B)	Reference comparison
Compared non-zero phase values	179	Complete magnitude comparison
Mean absolute error	0.4857%	Reference Table I
Maximum absolute error	1.4500% (node 610, phase A)	Reference Table I

Table 8 shows representative nodes: the minimum-voltage region, the regulated high-voltage region, the source-side node, and the 0.48-kV secondary node 610. The complete 179-value comparison remains available in the repository report.

Table 8. Representative voltage-magnitude comparison for the IEEE 123-bus feeder.

Node	Phase	IEEE Reference (p.u.)	Open-RPS (p.u.)	Error (%)
65	A	0.9858	0.9945	0.8789
149	B	1.0437	1.0437	0.0046
150	A	1.0000	1.0000	0.0000
610	A	0.9882	1.0025	1.4500

Appendix A (Tables A3 and A4) additionally presents a direct voltage and branch-current comparison between Open-RPS and GridLAB-D for a representative subset of IEEE 123-bus nodes and branches. The results show close agreement between both tools, and further improving this agreement is identified as a subject for future work.

4. Discussion

The results demonstrate that Open-RPS reproduces the IEEE reference values with high numerical fidelity for the three evaluated feeders. The maximum voltage magnitude error was 0.022% for the 4-bus system and 0.58% for the 13-bus system, both below their respective reference tolerances. For the IEEE 123-bus feeder, the complete comparison of 179 non-zero phase magnitudes produced a mean absolute error of 0.4857% and a maximum error of 1.4500%. These results support the implementation of the load models, transformers, regulators, capacitors, and radial BFS solver within the documented scope.

The small deviation in the minimum voltage of the 13-bus system (0.58% at node 652) is concentrated at the terminal buses of single-phase branches. This level of parity validates that the modular architecture preserved the physics of the electrical problem without introducing systematic errors during refactoring.

Compared with the tools reviewed in the introduction (Table 1), Open-RPS occupies the niche of open-source, auditable implementations specifically oriented towards the three-phase BFS method with explicit validation against IEEE benchmarks. It does not aim to replace OpenDSS or pandapower in large-scale studies but rather to offer a transparent computational base for teaching, reproducible research, and incremental algorithm development.

The current limitation of the software is its restriction to radial network topologies (without support for meshed networks).

5. Conclusions

Open-RPS v1.0 is an open, modular, and reproducible platform for three-phase power flow analysis in radial distribution networks, implemented in Python using the backward/forward sweep method without commercial license dependencies. Validation against the IEEE 4-bus, 13-bus, and 123-bus feeders confirms the numerical accuracy of the implementation within the tested scope. The comparison with GridLAB-D showed consistency.

From a computational performance perspective, Open-RPS achieves pure solution times of 3.2 ms and 9.7 ms for the 4-bus and 13-bus systems, respectively, compared to 232.3 ms and 1031.4 ms recorded with the equivalent implementation in MATLAB (RPS Toolbox). These reductions of approximately 33% and 77% are obtained without sacrificing numerical precision, which validates the advantages of the Python scientific ecosystem—specifically NumPy, SciPy, and NetworkX—for power distribution computing applications and confirms the suitability of Open-RPS for quasi-static time-series analyses where cumulative computational cost is critical.

The results as a whole validate that the modular architecture preserves the mathematical accuracy of the BFS method and position Open-RPS as an adequate computational base for teaching, reproducible research, and integration of distributed energy resources. As future work, four priority lines are identified: extending the solver to weakly meshed networks using nodal compensation methods; formulating three-phase unbalanced optimal power flow integrated with Python optimization environments; formally analyzing the convergence limits of the BFS method under high unbalance conditions and critical R/X ratios; and developing a voltage regulation strategy for PV nodes with distributed reactive compensation under high solar penetration.

https://doi.org/10.3390/engproc2026153022

Author Contributions: Conceptualization, A.M., S.O., D.V. and J.R.-F.; methodology, A.M., S.O., D.V. and J.R.-F.; software, A.M. and S.O.; validation, A.M., S.O. and D.V.; formal analysis, A.M., S.O. and J.R.-F.; investigation, A.M. and S.O.; resources, J.R.-F. and D.V.; data curation, A.M. and S.O.; writing original draft preparation, A.M. and S.O.; writing review and editing, A.M., S.O., D.V. and J.R.-F.; visualization, A.M. and S.O.; supervision, J.R.-F. and D.V.; 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 received no external funding.

Data Availability Statement: The source code, validation data, input files for the IEEE systems, automated tests, installation guide, reproduction scripts, and validation reports are publicly available at https://github.com/SantiagoOjeda13/Open-rps (accessed on 31 July 2026).

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.

Appendix A

This appendix provides the complete node-by-node voltage profile and branch-current results for the IEEE 13-bus system, summarized in Section 3 (IEEE 13-bus System). Table A1 lists the voltage magnitude and angle for every phase at every node, and Table A2 lists the branch currents for all 12 branches of the feeder, in both cases compared against the IEEE PES DSAS reference values and the GridLAB-D benchmark. Tables A3 and A4 present the corresponding voltage-magnitude and branch-current comparison for a representative subset of the IEEE 123-bus system, contrasting the IEEE/RPS Toolbox reference values, Open-RPS, and GridLAB-D.

Table A1. Nodal voltage profile in p.u.—IEEE 13-bus test system (base: 12.47 kV LL/4.16 kV LL).

Node	Source	V_a (p.u.) $\angle\theta_a$	V_b (p.u.) $\angle\theta_b$	V_c (p.u.) $\angle\theta_c$
650	IEEE 13 node test feeder.	1.0000 ∠0.00°	1.0000 ∠−120.00°	1.0000 ∠120.00°
	Open-RPS	1.0000 ∠0.00°	1.0000 ∠−120.00°	1.0000 ∠120.00°
	GridLAB-D	1.0000 ∠0.00°	1.0000 ∠−120.00°	1.0000 ∠120.00°
631	IEEE 13 node test feeder.	1.0625 ∠0.00°	1.0500 ∠−120.00°	1.0687 ∠120.00°
	Open-RPS	1.0625 ∠0.00°	1.0500 ∠−120.00°	1.0687 ∠120.00°
	GridLAB-D	1.0666 ∠0.00°	1.0526 ∠−120.00°	1.0738 ∠120.00°
632	IEEE 13 node test feeder.	1.0210 ∠−2.49°	1.0420 ∠−121.72°	1.0174 ∠117.83°
	Open-RPS	1.0211 ∠−2.50°	1.0421 ∠−121.71°	1.0169 ∠117.83°
	GridLAB-D	1.0336 ∠−2.22°	1.0371 ∠−121.62°	1.0290 ∠117.94°
633	IEEE 13 node test feeder.	1.0180 ∠−2.56°	1.0401 ∠−121.77°	1.0148 ∠117.82°
	Open-RPS	1.0181 ∠−2.56°	1.0402 ∠−121.75°	1.0143 ∠117.82°
	GridLAB-D	1.0306 ∠−2.28°	1.0352 ∠−121.67°	1.0264 ∠117.93°
634	IEEE 13 node test feeder.	0.9940 ∠−3.23°	1.0218 ∠−122.22°	0.9960 ∠117.34°
	Open-RPS	0.9941 ∠−3.24°	1.0218 ∠−122.21°	0.9954 ∠117.34°
	GridLAB-D	1.0070 ∠−2.94°	1.0167 ∠−122.13°	1.0078 ∠117.46°
645	IEEE 13 node test feeder.	—	1.0329 ∠−121.90°	1.0155 ∠117.86°
	Open-RPS	—	1.0329 ∠−121.89°	1.0149 ∠117.86°
	GridLAB-D	—	1.0230 ∠−121.80°	1.0324 ∠117.84°

Table A1. *Cont.*

Node	Source	V_a (p.u.) $\angle\theta_a$	V_b (p.u.) $\angle\theta_b$	V_c (p.u.) $\angle\theta_c$
646	IEEE 13 node test feeder.	—	1.0311 ∠−121.98°	1.0134 ∠117.90°
	Open-RPS	—	1.0312 ∠−121.96°	1.0129 ∠117.90°
	GridLAB-D	—	1.0183 ∠−121.87°	1.0336 ∠117.81°
671	IEEE 13 node test feeder.	0.9900 ∠−5.30°	1.0529 ∠−122.34°	0.9778 ∠116.02°
	Open-RPS	0.9913 ∠−5.31°	1.0512 ∠−122.32°	0.9740 ∠115.83°
	GridLAB-D	1.0133 ∠−5.17°	1.0508 ∠−121.83°	0.9794 ∠116.11°
675	IEEE 13 node test feeder.	0.9835 ∠−5.56°	1.0553 ∠−122.52°	0.9758 ∠116.03°
	Open-RPS	0.9848 ∠−5.56°	1.0536 ∠−122.50°	0.9721 ∠115.84°
	GridLAB-D	1.0070 ∠−5.42°	1.0531 ∠−122.01°	0.9774 ∠116.12°
680	IEEE 13 node test feeder.	0.9900 ∠−5.30°	1.0529 ∠−122.34°	0.9778 ∠116.02°
	Open-RPS	0.9913 ∠−5.31°	1.0512 ∠−122.32°	0.9740 ∠115.83°
	GridLAB-D	1.0133 ∠−5.17°	1.0508 ∠−121.83°	0.9794 ∠116.11°
684	IEEE 13 node test feeder.	0.9881 ∠−5.32°	—	0.9758 ∠115.92°
	Open-RPS	0.9893 ∠−5.33°	—	0.9720 ∠115.73°
	GridLAB-D	1.0113 ∠−5.19°	—	0.9773 ∠116.00°
692	IEEE 13 node test feeder.	0.9900 ∠−5.31°	1.0529 ∠−122.34°	0.9777 ∠116.02°
	Open-RPS	0.9913 ∠−5.31°	1.0512 ∠−122.32°	0.9740 ∠115.83°
	GridLAB-D	1.0133 ∠−5.17°	1.0508 ∠−121.83°	0.9794 ∠116.11°
611	IEEE 13 node test feeder.	—	—	0.9738 ∠115.78°
	Open-RPS	—	—	0.9700 ∠115.58°
	GridLAB-D	—	—	0.9753 ∠115.85°
652	IEEE 13 node test feeder.	0.9825 ∠−5.25°	—	—
	Open-RPS	**0.9843** ∠−5.17°	—	—
	GridLAB-D	1.0057 ∠−5.12°	—	—

Reference: IEEE PES DSAS — Table A1 (base: 12.47 kV LL / 4.16 kV LL). Minimum IEEE voltage: 0.9825 p.u. at node 652 (phase A). Minimum Open-RPS voltage: 0.9843 p.u. at node 652 (phase A). Maximum voltage: 1.0625 p.u. at node 631 (phase A). All values within the 0.5 % tolerance.

Table A2. Branch currents—IEEE 13-bus system.

Branch	Source	I_a (A) $\angle\theta$	I_b (A) $\angle\theta$	I_c (A) $\angle\theta$
650–631	IEEE 13 node test feeder.	558.40 ∠−28.58°	414.87 ∠−140.91°	586.60 ∠93.59°
	Open-RPS	558.52 ∠−28.61°	414.98 ∠−140.98°	588.99 ∠93.33°
	GridLAB-D	556.34 ∠−28.50°	413.76 ∠−140.81°	584.18 ∠93.71°
631–632	IEEE 13 node test feeder.	558.40 ∠−28.58°	414.87 ∠−140.91°	586.60 ∠93.59°
	Open-RPS	558.52 ∠−28.61°	414.98 ∠−140.98°	589.00 ∠93.33°
	GridLAB-D	556.34 ∠−28.50°	413.76 ∠−140.81°	584.18 ∠93.71°
632–633	IEEE 13 node test feeder.	81.33 ∠−37.74°	61.12 ∠−159.09°	62.71 ∠80.47°
	Open-RPS	81.32 ∠−37.75°	61.12 ∠−159.08°	62.74 ∠80.47°
	GridLAB-D	80.97 ∠−37.71°	60.96 ∠−159.08°	62.36 ∠80.50°
632–645	IEEE 13 node test feeder.	—	143.02 ∠−142.66°	65.21 ∠57.83°
	Open-RPS	—	143.02 ∠−142.66°	65.19 ∠57.82°
	GridLAB-D	—	143.03 ∠−142.58°	65.46 ∠57.88°
632–671	IEEE 13 node test feeder.	478.29 ∠−27.03°	215.12 ∠−134.66°	475.50 ∠99.90°
	Open-RPS	478.41 ∠−27.06°	215.18 ∠−134.80°	477.59 ∠99.56°
	GridLAB-D	476.60 ∠−26.94°	214.20 ∠−134.50°	473.34 ∠100.09°
633–634	IEEE 13 node test feeder.	704.83 ∠−37.74°	529.73 ∠−159.09°	543.45 ∠80.47°
	Open-RPS	704.76 ∠−37.75°	529.71 ∠−159.08°	543.74 ∠80.47°
	GridLAB-D	701.71 ∠−37.71°	528.34 ∠−159.08°	540.41 ∠80.50°

Table A2. *Cont.*

Branch	Source	I_a (A) ∠θ	I_b (A) ∠θ	I_c (A) ∠θ
645–646	IEEE 13 node test feeder.	—	65.21 ∠−122.18°	65.21 ∠57.82°
	Open-RPS	—	65.20 ∠−122.18°	65.20 ∠57.82°
	GridLAB-D	—	65.46 ∠−122.12°	65.46 ∠57.88°
671–684	IEEE 13 node test feeder.	63.07 ∠−39.12°	—	71.15 ∠121.62°
	Open-RPS	63.18 ∠−39.05°	—	71.13 ∠121.29°
	GridLAB-D	63.36 ∠−39.10°	—	71.17 ∠121.84°
671–692	IEEE 13 node test feeder.	229.11 ∠−18.18°	69.61 ∠−55.19°	178.38 ∠109.39°
	Open-RPS	228.78 ∠−18.14°	69.45 ∠−55.26°	178.94 ∠109.06°
	GridLAB-D	228.01 ∠−18.00°	69.82 ∠−55.05°	177.53 ∠109.72°
684–611	IEEE 13 node test feeder.	—	—	71.15 ∠121.61°
	Open-RPS	—	—	71.13 ∠121.29°
	GridLAB-D	—	—	71.17 ∠121.84°
684–652	IEEE 13 node test feeder.	63.07 ∠−39.12°	—	—
	Open-RPS	63.20 ∠−39.07°	—	—
	GridLAB-D	63.36 ∠−39.10°	—	—
692–675	IEEE 13 node test feeder.	205.33 ∠−5.15°	69.61 ∠−55.19°	124.07 ∠111.79°
	Open-RPS	205.06 ∠−5.09°	69.43 ∠−55.27°	124.60 ∠111.30°
	GridLAB-D	204.40 ∠−4.88°	69.82 ∠−55.05°	123.27 ∠112.27°

Reference: IEEE PES DSAS — Table A2. All Open-RPS and GridLAB-D values show agreement with the official reference in magnitude and angle.

Table A3. Voltage magnitude comparison—IEEE 123-bus test system (representative nodes).

Node	IEEE 123 Node Test Feeder			Open-RPS			GridLAB-D		
	V_a (p.u.)	V_b (p.u.)	V_c (p.u.)	V_a (p.u.)	V_b (p.u.)	V_c (p.u.)	V_a (p.u.)	V_b (p.u.)	V_c (p.u.)
150	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000
1	1.0313	1.0413	1.0349	1.0336	1.0404	1.0349	1.0318	1.0414	1.0348
7	1.0220	1.0395	1.0292	1.0259	1.0381	1.0292	1.0229	1.0398	1.0291
8	1.0160	1.0383	1.0254	1.0210	1.0365	1.0254	1.0171	1.0386	1.0253
13	1.0080	1.0361	1.0198	1.0146	1.0337	1.0198	1.0095	1.0365	1.0196
52	1.0020	1.0349	1.0165	1.0097	1.0316	1.0162	1.0039	1.0352	1.0164
57	0.9947	1.0307	1.0114	1.0031	1.0275	1.0103	0.9970	1.0311	1.0113
60	0.9882	1.0257	1.0053	0.9967	1.0239	1.0032	0.9911	1.0261	1.0054
67	1.0357	1.0312	1.0346	1.0442	1.0296	1.0327	1.0390	1.0315	1.0346
76	1.0361	1.0297	1.0350	1.0444	1.0286	1.0327	1.0394	1.0301	1.0351
83	1.0438	1.0376	1.0391	1.0505	1.0348	1.0370	1.0472	1.0377	1.0392
91	1.0340	1.0266	1.0376	1.0422	1.0256	1.0353	1.0374	1.0271	1.0376
97	1.0347	1.0307	1.0339	1.0432	1.0292	1.0319	1.0380	1.0310	1.0339
101	1.0339	1.0304	1.0333	1.0424	1.0290	1.0313	1.0373	1.0308	1.0333

Reference: IEEE PES DSAS 123-bus test feeder (RPS Toolbox/GridLAB-D benchmark). Entries shown correspond to a representative subset of nodes.

Table A4. Branch current magnitude comparison—IEEE 123-bus test system (representative branches).

Branch	IEEE 123 Node Test Feeder			Open-RPS			GridLAB-D		
	I_a (A)	I_b (A)	I_c (A)	I_a (A)	I_b (A)	I_c (A)	I_a (A)	I_b (A)	I_c (A)
150–150r	655.83	425.90	523.78	613.55	401.41	494.97	660.62	430.66	527.49
60–160	240.05	172.00	191.05	233.47	175.98	192.84	243.62	175.49	193.26
160–160r	240.05	172.00	191.05	233.47	175.98	192.84	243.62	175.49	193.26
160r–67	228.62	170.93	185.26	222.35	174.88	186.99	232.02	174.40	187.41
67–72	118.75	126.11	132.91	114.58	130.70	133.24	116.90	127.31	133.28
67–97	82.67	54.29	64.43	82.42	54.37	64.51	83.79	55.86	65.17

Reference: IEEE PES DSAS 123-bus test feeder (RPS Toolbox/GridLAB-D benchmark).

References

1. Kersting, W.H. *Distribution System Modeling and Analysis*, 3rd ed.; CRC Press: Boca Raton, FL, USA, 2012.
2. Taqiyuddin, S.; Suwarno; Nurdin, M.; Sianipar, G.H.M. Backward Forward Sweep Algorithm for Unbalanced Three-Phase Power Flow Analysis in Distribution Systems Containing Voltage Regulator. In Proceedings of the 2021 3rd International Conference on High Voltage Engineering and Power Systems (ICHVEPS), Bandung, Indonesia, 5–6 October 2021; pp. 1–6.
3. Kumar, A.; Kumar, A.; Mallipeddi, R.; Lee, D.-G. Adaptive Backward/Forward Sweep for Solving Power Flow of Islanded Microgrids. *Energies* **2022**, *15*, 9348. [CrossRef]
4. Zimmerman, R.D.; Murillo-Sánchez, C.E.; Thomas, R.J. MATPOWER: Steady-State Operations, Planning, and Analysis Tools for Power Systems Research and Education. *IEEE Trans. Power Syst.* **2011**, *26*, 12–19. [CrossRef]
5. Dugan, R.C.; McDermott, T.E. An Open Source Platform for Collaborating on Smart Grid Research. In Proceedings of the IEEE Power and Energy Society General Meeting, San Diego, CA, USA, 24–29 July 2011; pp. 1–7.
6. Chassin, D.P.; Fuller, J.C.; Djilali, N. GridLAB-D: An Agent-Based Simulation Framework for Smart Grids. *J. Appl. Math.* **2014**, *2014*, 492320. [CrossRef]
7. Thurner, L.; Scheidler, A.; Schäfer, F.; Menke, J.; Dollichon, J.; Wenderoth, F.; Meinecke, S.; Braun, M. pandapower: An Open Source Python Tool for Convenient Modeling, Analysis, and Optimization of Electric Power Systems. *IEEE Trans. Power Syst.* **2018**, *33*, 6510–6521. [CrossRef]
8. Didier. Radial Power Solver. MATLAB Central File Exchange. 2026. Available online: https://la.mathworks.com/matlabcentral/fileexchange/179944-radial-power-solver (accessed on 14 January 2026).
9. Hagberg, A.A.; Schult, D.A.; Swart, P.J. Exploring Network Structure, Dynamics, and Function using NetworkX. In Proceedings of the 7th Python in Science Conference, Pasadena, CA, USA, 19–24 August 2008; pp. 11–15.
10. Harris, C.R.; Millman, K.J.; van der Walt, S.J.; Gommers, R.; Virtanen, P.; Cournapeau, D.; Wieser, E.; Taylor, J.; Berg, S.; Smith, N.J.; et al. Array programming with NumPy. *Nature* **2020**, *585*, 357–362. [CrossRef] [PubMed]
11. Virtanen, P.; Gommers, R.; Oliphant, T.E.; Haberland, M.; Reddy, T.; Cournapeau, D.; Burovski, E.; Peterson, P.; Weckesser, W.; Bright, J.; et al. SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python. *Nat. Methods* **2020**, *17*, 261–272. [CrossRef] [PubMed]
12. IEEE PES Distribution System Analysis Subcommittee. Radial Test Feeders. Available online: https://cmte.ieee.org/pes-testfeeders/resources/ (accessed on 30 April 2026).
13. McKinney, W. Data Structures for Statistical Computing in Python. In Proceedings of the 9th Python in Science Conference, Austin, TX, USA, 28 June–3 July 2010; pp. 56–61.

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.