

Article

Analysis and Comparison of Chebyshev–Halley Multipoint Methods for Power Flow Calculation in Monopolar Direct-Current Networks

Sebastián Salazar-Méndez ¹, José Daniel Pico-Díaz ¹ and Oscar Danilo Montoya ^{1,2,*}

¹ Grupo de Compatibilidad e Interferencia Electromagnética (GCEM), Facultad de Ingeniería, Universidad Distrital Francisco José de Caldas, Bogotá 110231, Colombia; sesalazarm@udistrital.edu.co (S.S.-M.); jdpicod@udistrital.edu.co (J.D.P.-D.)

² Department of Automatics, Electrical Engineering and Electronic Technology, Universidad Politécnica de Cartagena, 30202 Cartagena, Spain

* Correspondence: odmontoyag@udistrital.edu.co

Abstract

The increasing penetration of direct-current (DC) technologies in power transmission and distribution systems necessitates efficient and robust tools for steady-state analysis. This paper presents a comparative evaluation of the Chebyshev–Halley (CH) family of multipoint iterative methods against the classical Newton–Raphson (NR) method for power flow calculation in monopolar DC networks. Both methods were implemented in MATLAB and tested on four radial test systems of increasing complexity (10, 21, 33, and 69 nodes) under three distinct initialization scenarios: optimal (flat start), adverse ($V^{(0)} = 0.5$ p.u.), and random ($V^{(0)} \sim \mathcal{U}[0.8, 1.2]$ p.u.). Performance was assessed using key metrics including iteration count, CPU time, solution accuracy, and convergence failure rate. The results demonstrate that the cubic convergence of CH consistently reduces the number of iterations by one when compared to NR across all systems. However, this reduction does not translate into computational savings, as CH exhibits median CPU times 1.36 to 2.44 times higher than those of NR, given its higher cost per iteration, which involves solving two additional linear systems. Under adverse starting conditions, both methods converge for the 10-, 21-, and 33-node systems, but CH fails on the 69-node network due to severe Jacobian ill-conditioning, from which NR recovers through an implicit regularization mechanism. Under random initializations, both methods show high failure rates, reaching 100% in the 69-node network. It is concluded that, while CH offers superior convergence order and final accuracy, NR remains more computationally efficient for small- to medium-scale networks under flat-start conditions. The CH family is best justified in high-precision applications or larger networks where the iteration reduction may offset its per-step overhead. Future work should focus on extending CH to meshed and multi-source DC networks, developing quasi-Newton variants to reduce its computational cost, and designing hybrid NR-CH strategies that combine global robustness with local cubic convergence.



Academic Editors: Lijuan Zha, Jinliang Liu and Jian Liu

Received: 22 April 2026

Revised: 15 July 2026

Accepted: 15 July 2026

Published: 19 July 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 \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

Keywords: DC power flow; Chebyshev–Halley methods; Newton–Raphson; constant power loads; monopolar DC networks

1. Introduction

The growing global demand for energy, coupled with the need to integrate renewable generation sources located in remote areas, has driven the development and application

of high-voltage direct-current (HVDC) transmission technologies [1,2]. This technology presents itself as a viable solution and, in many cases, the only technical option for the efficient transport of large blocks of power over long distances, especially in submarine or underground links exceeding 50 km, where alternating-current (AC) transmission faces significant technical and economic limitations [3]. Furthermore, HVDC facilitates the interconnection of asynchronous electrical systems and offers precise control of the power flow, contributing to the overall stability and reliability of the power system [4].

Concurrently, at the distribution level, low-voltage direct-current (LVDC) microgrids have emerged as a promising alternative for the efficient integration of distributed energy resources, such as photovoltaic (PV) solar panels, batteries, and electric vehicles (EVs), which inherently operate in DC [5]. These systems offer significant advantages over their AC counterparts, including the elimination of concerns related to reactive power, frequency, and synchronization, as well as a reduction in the number of conversion stages, which translates into lower energy losses and a greater overall efficiency [6,7]. Field studies have demonstrated that an LVDC distribution system can outperform an equivalent AC system in efficiency, consolidating its potential for growth in the current energy landscape [8].

The practical motivation for developing robust and efficient power flow solvers for DC networks has grown substantially in recent years. Data centers powered by DC buses demand fast steady-state solvers for real-time energy management [6], while fast-charging infrastructures for EVs impose stringent convergence requirements due to the highly nonlinear behavior of power converters [9]. Furthermore, flexible DC distribution networks integrating PV generation and battery storage operate under continuous load variability, making solver robustness a critical design criterion [5]. In this context, the CH-D variant proposed in this work addresses a concrete engineering need: providing a solver that maintains cubic convergence accuracy while extending the convergence basin under the remote or uncertain initializations that characterize these practical scenarios.

The steady-state analysis of both HVDC and LVDC systems is based on the study of power flow, an essential computational tool that determines nodal voltages and active power flows in the lines [10]. The equations governing the power flow in a DC network are inherently nonlinear, primarily due to the presence of constant power loads (CPLs), which model the behavior of many electronic converters and devices connected to the grid [11]. CPLs are characterized by maintaining a constant output power regardless of the voltage at their terminals, which introduces a negative incremental impedance that can lead to voltage instabilities and challenges in the convergence of numerical methods [9]. Therefore, solving the resulting system of nonlinear equations requires the use of robust and efficient iterative numerical methods [12].

Traditionally, the Newton–Raphson method (NR) method has been the industry standard for solving power flow problems, both in AC and DC, given its quadratic convergence near the solution [13]. Its application in DC systems has been widely validated, demonstrating high effectiveness and accuracy in various network configurations [14]. However, as electrical systems grow in scale and complexity, with thousands of nodes and a high penetration of CPLs, the performance of NR may be compromised. Its convergence strongly depends on the choice of an adequate initial guess and on the conditioning of the Jacobian matrix, which must be computed and inverted (or its equivalent, solving a linear system) at each iteration. This computational cost can be prohibitive in large-scale systems, and the lack of global convergence guarantees remains a significant limitation [15,16].

In the search for more efficient and robust alternatives, research on numerical methods has turned towards multipoint iterative schemes. These methods, unlike single-point methods such as NR, use additional evaluations of the function and, in some cases, its derivatives at intermediate points within the same iteration, in order to achieve higher

orders of convergence without a proportional increase in computational cost [17]. According to the Kung-Traub conjecture, a memoryless multipoint method that consumes $n + 1$ functional evaluations per iteration can achieve a maximum convergence order of 2^n , which opens the possibility of doubling the convergence order of NR (which is 2) to orders 4, 6, 8, or higher, with a similar computational effort [13]. This characteristic makes these tools particularly attractive for problems where function evaluation is expensive, as is the case with the power flow in electrical networks.

Within the wide range of multipoint methods, the Chebyshev–Halley (CH) family stands out for its balance between simplicity and efficiency. This parametric family of methods, which includes the classical variants of the Chebyshev, Halley, and super-Halley techniques, typically achieves a cubic convergence order by incorporating second-order information of the function, albeit without explicitly computing second derivatives [18]. In its formulation for multivariable systems, this translates into solving two linear systems with the same Jacobian matrix per iteration, which represents a moderate additional computational cost compared to NR [19]. Recent research has shown that these methods can ‘dominate’ in terms of lower numerical error and computation time compared to NR in load flow problems [20]. Furthermore, in the specific context of DC networks, it has been observed that multipoint schemes can avoid the repeated inversion of the Jacobian matrix, dramatically improving computational efficiency without compromising robustness [21].

Despite these theoretical and demonstrated advantages in other fields, the systematic application and comparison of the CH family for solving the power flow in monopolar DC networks is an area that has not yet been explored in depth. Most studies have focused on improving classical methods or proposing ad hoc techniques for specific topologies [22,23]. Therefore, there is a research opportunity that involves quantitatively comparing the advantages of these higher-order multipoint methods against the well-established NR in a controlled and representative environment.

As a key contribution, this work introduces a variant called Chebyshev–Halley with adaptive damping (CH-D), which consists of scaling the computed correction whenever its norm exceeds a threshold $\delta = 0.5$, using an adaptive damping factor $\alpha = \min(1, \delta / \|\Delta \mathbf{V}^{(n)}\|_\infty)$. This mechanism prevents the iterate from leaving the convergence basin under unfavorable initializations. When $\|\Delta \mathbf{V}\|_\infty \leq \delta$, one has $\alpha = 1$ and CH-D behaves identically to standard CH, fully preserving its cubic convergence rate. To the best of the authors’ knowledge, this stabilized variant has not been previously reported for power flow analysis in monopolar DC networks. Combined with a symmetric Jacobian regularization ($\text{rcond} < 10^{-14}$, $+10^{-12}\mathbf{I}$) applied identically to both NR and CH-D, this formulation provides a methodologically fair comparison between the two approaches.

In light of the above, the main objective of this research paper is to evaluate the effectiveness of memoryless multipoint methods of the CH family against the NR method in calculating the power flow of monopolar DC networks. To this effect, both algorithms will be implemented in the MATLAB programming environment and tested on 10, 21, 33, and 69-node systems, considering different initial conditions (optimal, adverse, and random). Performance will be measured through key metrics such as (i) the number of iterations until convergence, (ii) CPU time, (iii) solution accuracy, and (iv) numerical robustness.

The expected results seek not only to determine which method is superior in absolute terms, but also to characterize the specific scenarios where the CH family offers significant advantages, thus providing a practical guide for engineers and researchers in analyzing DC systems. Furthermore, this study aims to establish a foundation for future research on advanced numerical methods applied to modern power systems within the context of the energy transition, thereby contributing to the development of more efficient and robust tools for the steady-state analysis of DC networks.

The remainder of this article is organized as follows. Section 2 presents the mathematical formulation of the power flow problem in DC networks. Section 3 outlines the methodology used for the comparison. Section 4 presents the test systems analyzed. Section 5 examines the obtained results. Section 6 discusses the implications of said results, and, finally, Section 7 presents the conclusions of this research and proposals for future work.

2. Calculating the DC Power Flow in Monopolar Networks

Power flow calculation serves as a fundamental tool in the analysis and operation of DC electrical networks. Although methods such as NR and Gauss–Seidel have been widely adopted due to their robustness and ease of implementation, their convergence rates impose limitations when working with larger-scale networks [24]. In this context, our work proposes the application of the CH multipoint method, whose cubic convergence order allows reducing the number of required iterations compared to classical approaches. Validation was carried out on four test networks (10, 21, 33, and 69 nodes) [25], with the aim of demonstrating the feasibility of the method as an efficient alternative for the analysis of small- and medium-scale DC distribution networks.

2.1. Mathematical Formulation of the Power Flow Problem

When working with DC networks, the analysis is considerably simplified from the voltage perspective, since the voltage does not exhibit angular phase shifts. Consequently, the only power component involved in the model is active power [7]. In general terms, the equation for the DC power flow at a demand node k is as follows:

$$\mathbf{P}_k = \mathbf{V}_k \sum_{m=1}^N \mathbf{G}_{km} \mathbf{V}_m \quad (1)$$

where V_k and V_m are the voltage magnitudes at the corresponding nodes, while G_{km} denotes the elements of the nodal conductance matrix G , whose structure reflects the interconnection topology of the network [7]. This matrix allows establishing the relationship between the nodal voltage vector and the injected current vector in the system:

$$\begin{pmatrix} i_1 \\ \vdots \\ i_k \\ \vdots \\ i_n \end{pmatrix} = \begin{bmatrix} G_{11} & \cdots & G_{1k} & \cdots & G_{1n} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ G_{k1} & \cdots & G_{kk} & \cdots & G_{kn} \\ \vdots & \ddots & \vdots & \ddots & \vdots \\ G_{n1} & \cdots & G_{nk} & \cdots & G_{nn} \end{bmatrix} \begin{pmatrix} v_1 \\ \vdots \\ v_k \\ \vdots \\ v_n \end{pmatrix} \quad (2)$$

The central focus of this work is the implementation of two iterative numerical methods with different convergence orders for solving the power flow equations defined above. These methods are NR and CH, which are discussed in the following sections.

2.2. Newton–Raphson Method

NR is an iterative procedure based on local linearization via the Jacobian matrix. At each step, the vector of unknowns is corrected by solving the linear system defined by the Jacobian evaluated at the current iteration and the function vector, subsequently applying this correction to the approximate solution. NR exhibits quadratic convergence in the vicinity of the solution. However, its performance is highly dependent on the conditioning of the Jacobian matrix and on the initial guess [26]. In DC systems, it is widely employed as a benchmark due to its local robustness and speed, having been adapted in numerous

studies on DC power flows. From the general equation described in the previous section, the system for this method can be formulated as follows:

$$F_k(\mathbf{V}_d) = \mathbf{V}_k \sum_{m=1}^N \mathbf{G}_{km} \mathbf{V}_m - \mathbf{P}_k = 0. \quad (3)$$

Equation (3) expresses the active power balance at node k . Physically, it establishes that the power delivered by said node to the network must equal the power demanded P_k at the same point. The term $V_k \sum_m G_{km} V_m$ represents the power injected by node k into the rest of the network, and it can be interpreted as the product of the voltage at node k and the total current leaving it. At the branch level, the generalized Ohm law states that the current flowing through the line connecting nodes k and m is proportional to the voltage difference between them, $I_{km} = g_{km}(V_k - V_m)$, where g_{km} is the branch conductance. Summing these branch currents over all neighbors of node k and incorporating the sign convention of the nodal conductance matrix G yields the aggregated nodal expression $V_k \sum_m G_{km} V_m$ used in Equation (3).

When $F_k = 0$, the node is in equilibrium, indicating that the incoming power equals the outgoing one. The objective of iterative methods is precisely to determine the voltage vector $\mathbf{V}_d$ that simultaneously satisfies this condition for all nodes in the system.

Furthermore, it is necessary to express this equation in vector form, since, when implementing the algorithm in MATLAB [26], a matrix representation facilitates both programming and numerical evaluation:

$$\mathbf{F}(\mathbf{V}_d) = \text{diag}(\mathbf{V}_d) (\mathbf{G}_{dd} \mathbf{V}_d + \mathbf{G}_{dg} \mathbf{V}_g) + \mathbf{P}_{ref} = 0 \quad (4)$$

Equation (4) includes the term $\mathbf{P}_{ref}$, which corresponds to the vector of demanded powers defined here as $P_{ref} = -P_d$, i.e., as a negative injection consistent with the sign convention of Equation (3). When operated together with the other terms, it establishes the balance, since the equation equals zero only when the power injected at each node equals the demanded power.

Following the logic of the method, the Jacobian matrix is defined to represent the local linearization of the residual function $\mathbf{F}(\mathbf{V})$ at the current point. Thus, it indicates how each balance equation changes when each nodal voltage is perturbed. Its elements are obtained by partially differentiating each F_k with respect to each voltage $\mathbf{V}_m$. This matrix is divided into two main elements: the diagonal element $\mathbf{J}_{kk}$ and the off-diagonal element $\mathbf{J}_{km}$. The former accounts for the sensitivity of the node, while the latter incorporates the influence of neighboring nodes on one another [26]. The following expression groups these two elements in matrix form:

$$\mathbf{J} = \text{diag}(\mathbf{G}_{dd} \mathbf{V}_d + \mathbf{G}_{dg} \mathbf{V}_g) + \text{diag}(\mathbf{V}_d) \mathbf{G}_{dd} \quad (5)$$

Once $\mathbf{J}$ has been constructed and evaluated together with $\mathbf{F}$ at the current point $\mathbf{V}^{(k)}$, the linear system is formulated as follows:

$$\mathbf{J}(\mathbf{V}^{(k)}) \Delta \mathbf{V}^{(k)} = -\mathbf{F}(\mathbf{V}^{(k)}) \quad (6)$$

The system in Equation (6) answers the following question: how much must the current voltage vector be corrected to nullify the residual? The geometric insight is that, if $\mathbf{F}$ were linear, this correction would lead directly to the exact solution in a single step. Since $\mathbf{F}$ is nonlinear, the correction is only a local approximation valid near the current point, so the process must be repeated iteratively.

In practice, the vector $\Delta \mathbf{V}^{(k)}$ is not obtained by explicitly inverting $\mathbf{J}$, which would be computationally expensive, but rather by solving the linear system through factorization:

$$\Delta \mathbf{V}^{(k)} = -[\mathbf{J}(\mathbf{V}^{(k)})]^{-1} \mathbf{F}(\mathbf{V}^{(k)}). \quad (7)$$

Finally, the calculated correction is applied to the current voltage vector in order to obtain a new estimate that is closer to the true solution:

$$\mathbf{V}^{(k+1)} = \mathbf{V}^{(k)} + \Delta \mathbf{V}^{(k)}. \quad (8)$$

The complete process of evaluating $\mathbf{F}$, constructing $\mathbf{J}$, solving the linear system, and updating $\mathbf{V}$ is repeated until the norm of the correction falls below the established tolerance.

2.3. Chebyshev–Halley Method

The CH method constitutes a parametric family of multipoint methods that encompasses, among others, the Chebyshev, Halley, and super-Halley variants. These variants can achieve cubic convergence orders (3) or higher in their optimal formulations. This is accomplished by incorporating additional second-order corrections or auxiliary evaluations of the system functions, which reduces the number of iterations required to converge to the solution, albeit at the cost of more computations per iteration [27].

The initial formulation of the problem follows the same approach as the NR method, but with the addition of a correction term that elevates the convergence order to three, without requiring the explicit calculation of second-order derivatives.

Prior to solving any linear system, the numerical conditioning of the Jacobian is assessed. In networks with highly heterogeneous admittances, such as those containing very short branches with resistances on the order of $10^{-4} \Omega$, the Jacobian matrix $\mathbf{J}$ may become nearly singular at machine precision, causing lower-upper (LU) factorization to amplify round-off errors across all subsequent solves. To prevent this, a Tikhonov regularization is conditionally applied:

$$\mathbf{J} \leftarrow \mathbf{J} + \epsilon_r \mathbf{I}, \quad \text{if } \text{rcond}(\mathbf{J}) < \tau_r, \quad (9)$$

where $\epsilon_r = 10^{-12}$ is the regularization perturbation, $\tau_r = 10^{-14}$ is the conditioning threshold, and $\mathbf{I}$ is the identity matrix of appropriate dimension. The function $\text{rcond}(\cdot)$ returns an estimate of the reciprocal condition number in the 1-norm. Since the three linear systems of the method, corresponding to $\mathbf{h}_n$, $\mathbf{L}_n$, and $\mathbf{s}_n$ in Equations (10), (13) and (16), share the same LU factorization of $\mathbf{J}$, the regularization is applied once per iteration and acts simultaneously on all three systems. When $\text{rcond}(\mathbf{J}) \geq \tau_r$, Equation (9) is not activated, and the method operates on the original Jacobian without modification.

With this numerical safeguard in place, the CH iteration proceeds as follows:

$$\mathbf{J} \cdot \mathbf{h}_n = \mathbf{F}(\mathbf{V}^{(n)}) \Rightarrow \mathbf{h}_n = [\mathbf{F}'(\mathbf{V}^{(n)})]^{-1} \mathbf{F}(\mathbf{V}^{(n)}) \quad (10)$$

Equation (10) evaluates the residual $\mathbf{F}$ and the Jacobian $\mathbf{J}$ at the current point $\mathbf{V}^{(n)}$, and it solves the linear system to obtain the vector $\mathbf{h}_n$, which represents the direction and magnitude of the correction that NR would provide at that iteration. It is worth noting that $\mathbf{h}_n$ is not applied directly as the final correction, as occurs in the NR method. Instead, it serves as a starting point for constructing a more accurate correction that accounts for the curvature of the function.

Instead of advancing directly to the solution, the CH method first takes a partial step of magnitude $\beta = 2/3$ in the NR direction, obtaining the intermediate point $\mathbf{y}_n$ [28]. This point is neither the solution nor a complete iteration: it is an auxiliary point used exclusively to explore how the Jacobian changes along the correction direction:

$$\mathbf{y}_n = \mathbf{V}^{(n)} - \beta \mathbf{h}_n \quad \text{with} \quad \beta = \frac{2}{3} \quad (11)$$

The choice of $\beta = 2/3$ is not arbitrary; it is the value that ensures a cubic convergence order with the best balance between accuracy and computational cost.

Next, a second evaluation of the Jacobian is performed at the intermediate point $\mathbf{y}_n$, using exactly the same formula as that for $\mathbf{J}$, but substituting $\mathbf{V}_d$ with $\mathbf{y}_n$:

$$\mathbf{J}_y = \mathbf{F}'(\mathbf{y}_n) = \text{diag}(\mathbf{G}_{dd}\mathbf{y}_n + \mathbf{G}_{dg}\mathbf{V}_g) + \text{diag}(\mathbf{y}_n)\mathbf{G}_{dd} \quad (12)$$

Equation (12) is the key element that allows the CH method to detect the curvature of $\mathbf{F}$. If $\mathbf{J}_y \approx \mathbf{J}$, the function is nearly linear in that region, and the NR correction is already sufficiently accurate. If $\mathbf{J}_y$ differs significantly from $\mathbf{J}$, a substantial curvature is present, and the additional correction of the CH method makes a meaningful difference.

To relate both Jacobians, a matrix linear system is formulated to obtain $\mathbf{L}_n$, which indicates how much the slope of $\mathbf{F}$ has changed between the current point $\mathbf{V}^{(n)}$ and the intermediate point $\mathbf{y}_n$. If $\mathbf{L}_n = \mathbf{I}$, the two Jacobians are identical and there is no curvature. The more $\mathbf{L}_n$ deviates from the identity matrix, the greater the local nonlinearity of $\mathbf{F}$:

$$\mathbf{J} \cdot \mathbf{L}_n = \mathbf{J}_y \Rightarrow \mathbf{L}_n = \left[\mathbf{F}'(\mathbf{V}^{(n)}) \right]^{-1} \mathbf{F}'(\mathbf{y}_n) = \mathbf{J}^{-1} \mathbf{J}_y \quad (13)$$

Using the operator $\mathbf{L}_n$ defined in Equation (13), the curvature operator $\mathbf{M}_n$ is defined, which directly quantifies the deviation from linearity. If the function were perfectly linear, $\mathbf{J} = \mathbf{J}_y$, then $\mathbf{L}_n = \mathbf{I}$ and $\mathbf{M}_n = \mathbf{0}$, which would cause the entire additional correction of CH to be zero, reducing the method exactly to NR. Conversely, a considerable magnitude of $\mathbf{M}_n$ indicates that the curvature is relevant and that the additional CH correction will significantly improve the accuracy of the step:

$$\mathbf{M}_n = \mathbf{I} - \mathbf{L}_n \quad (14)$$

As a primary input for calculating the additional correction made by CH method, the auxiliary correction vector $\mathbf{g}_n$ is constructed. This vector combines information on how much should be corrected (given by $\mathbf{h}_n$) according to the curvature that exists in that direction (given by $\mathbf{M}_n$):

$$\mathbf{g}_n = \mathbf{M}_n \cdot \mathbf{h}_n \quad (15)$$

The following linear system defines the additional correction characteristic of CH. It is solved to obtain $\mathbf{s}_n$, i.e., the vector that concentrates all the curvature information captured in the previous steps. The operator $(\mathbf{I} - \frac{3}{2}\mathbf{M}_n)$ acts as an amplifier of the correction: it takes the curvature measured by $\mathbf{M}_n$ and weights it appropriately so that, when added to the NR direction, it elevates the convergence order from two to three [28]. Notably, if $\mathbf{M}_n \approx \mathbf{0}$, this system yields $\mathbf{s}_n \approx \mathbf{0}$, and CH is reduced to NR, confirming that the additional correction only acts when significant curvature is present:

$$\left(\mathbf{I} - \frac{\lambda}{\beta} \mathbf{M}_n \right) \mathbf{s}_n = \mathbf{g}_n \quad \text{with} \quad \lambda = 1, \beta = \frac{2}{3} \quad (16)$$

$$\left(\mathbf{I} - \frac{3}{2}\mathbf{M}_n\right)\mathbf{s}_n = \mathbf{g}_n \quad (17)$$

The following equation defines the final correction of the CH method, which must incorporate two clearly differentiated components:

$$\Delta\mathbf{V}^{(n)} = -\mathbf{h}_n - \frac{1}{2\beta}\mathbf{s}_n = -\mathbf{h}_n - \frac{3}{4}\mathbf{s}_n \quad (18)$$

Once the raw correction $\Delta\mathbf{V}^{(n)}$ has been computed via Equation (18), an adaptive damping factor is applied before the update step. When the iterate lies far from the solution, as occurs under adverse or random initializations, the norm of $\Delta\mathbf{V}^{(n)}$ can be excessively large, displacing the iterate beyond the convergence basin of the method. To prevent this, the correction is scaled as follows:

$$\alpha = \min\left(1, \frac{\delta}{\|\Delta\mathbf{V}^{(n)}\|_\infty}\right), \quad \delta = 0.5 \quad (19)$$

$$\Delta\mathbf{V}^{(n)} \leftarrow \alpha \cdot \Delta\mathbf{V}^{(n)} \quad (20)$$

When $\|\Delta\mathbf{V}^{(n)}\|_\infty \leq \delta$, it follows that $\alpha = 1$, and Equations (19) and (20) leave the correction unchanged, so the method behaves identically to the standard CH, and its cubic convergence rate is fully preserved. The damping is activated only when the correction is sufficiently large to risk leaving the convergence basin; once the iterate enters the region of local cubic convergence, $\alpha = 1$ in all subsequent iterations, and no accuracy is sacrificed. The overhead introduced by this operation is $\mathcal{O}(n)$ per iteration—one scalar division and one vector scaling—which is negligible relative to the $\mathcal{O}(n^3)$ LU factorization already performed.

Finally, the calculated correction is applied to the current voltage vector in order to obtain an estimate that is closer to the true solution, analogously to what was done via the NR method in Equation (8).

3. Comparison Methodology

Building upon the above-presented mathematical formulation, this section describes the systematic procedure employed to compare the performance of the NR and CH methods in solving the power flow problem for monopolar DC networks. The methodology encompasses the computational implementation, test cases, initial conditions, evaluation metrics, and statistical analysis of the results.

3.1. Computational Implementation

Both algorithms were implemented in MATLAB R2025b, leveraging its capabilities for handling sparse matrices and efficiently solving linear systems through LU factorization. A convergence tolerance of $\varepsilon = 10^{-6}$ (measured as the infinity norm of the voltage correction) was adopted, along with a predefined maximum number of iterations: $t_{\max} = 100$. The nodal conductance matrix G , the power vector P , and the slack node identification were defined identically for both methods, ensuring a fair and unbiased comparison.

3.1.1. Classical NR Approach

The NR algorithm implements the classical local linearization scheme [13]. Based on the residual function defined in (4) and the Jacobian in (5), the procedure is summarized in the following steps (see Algorithm 1):

Algorithm 1 Newton–Raphson for the DC power flow

Require: Conductance matrix G , power vector P , slack voltage V_{slack} , tolerance ε , maximum iterations t_{max}

Ensure: Voltage vector $\mathbf{V}$ at load nodes

- 1: **Define** $\mathcal{D}$ as the set of load nodes (all except slack)
- 2: **Initialize** $\mathbf{V}^{(0)} = 1.0$ p.u. for $i \in \mathcal{D}$
- 3: $t \leftarrow 0$
- 4: **while** $t < t_{\text{max}}$ **do**
- 5: Compute $\mathbf{G}\mathbf{v} = G \mathbf{V}^{(t)}$
- 6: Compute $\mathbf{F}(\mathbf{V}^{(t)}) = \text{diag}(\mathbf{V}^{(t)}) (\mathbf{G}_{dd} \mathbf{V}^{(t)} + \mathbf{G}_{dg} V_{\text{slack}}) + \mathbf{P}_{\text{ref}}$
- 7: Construct the Jacobian matrix $\mathbf{J}(\mathbf{V}^{(t)})$ with the following elements:

$$\mathbf{J}_{km} = \begin{cases} \mathbf{G}_{kk} \mathbf{V}_k + \sum_j \mathbf{G}_{kj} V_j & \text{if } k = m \\ \mathbf{G}_{km} V_k & \text{if } k \neq m \end{cases}$$

- 8: Solve $\mathbf{J} \Delta \mathbf{V} = -\mathbf{F}$ using LU factorization
- 9: Update $\mathbf{V}^{(t+1)} = \mathbf{V}^{(t)} + \Delta \mathbf{V}$
- 10: **if** $\|\Delta \mathbf{V}\|_{\infty} < \varepsilon$ **then**
- 11: **break**
- 12: **end if**
- 13: $t \leftarrow t + 1$
- 14: **end while**
- 15: **return** $\mathbf{V}^{(t)}$

This implementation follows the principles described in [14], where the effectiveness of the method is demonstrated.

3.1.2. Chebyshev–Halley Multipoint Method (Super-Halley)

The CH method belongs to the multipoint family that achieves cubic convergence by incorporating curvature information without explicitly computing second-order derivatives [19]. The super-Halley variant is employed, with parameters $\lambda = 1$ and $\beta = 2/3$ [29]. The procedure can be outlined as follows (see Algorithm 2):

Algorithm 2 Chebyshev–Halley for the DC power flow

Require: $G, P, V_{\text{slack}}, \varepsilon, t_{\text{max}}, \lambda = 1, \beta = 2/3$

Ensure: Voltage vector $\mathbf{V}$ at load nodes

- 1: **Define** $\mathcal{D}$ as the set of load nodes
- 2: **Initialize** $\mathbf{V}^{(0)} = 1.0$ p.u. for $i \in \mathcal{D}$
- 3: $n \leftarrow 0$
- 4: **while** $n < t_{\text{max}}$ **do**
- 5: Compute $\mathbf{G}\mathbf{v} = G \mathbf{V}^{(n)}$ and the residual $\mathbf{F}$ as in Algorithm 1
- 6: Construct $\mathbf{J}(\mathbf{V}^{(n)})$ (same expression as in Newton)
- 7: **if** $\text{rcond}(\mathbf{J}) < 10^{-14}$ **then**
- 8: $\mathbf{J} \leftarrow \mathbf{J} + 10^{-12} \mathbf{I}$
- 9: **end if**
- 10: Solve $\mathbf{J} \mathbf{h}_n = \mathbf{F}$ ▷ Newton step
- 11: Define intermediate point $\mathbf{y}_n = \mathbf{V}^{(n)} - \beta \mathbf{h}_n$
- 12: Compute $\mathbf{G}\mathbf{y}_n = G \mathbf{y}_n$ and construct $\mathbf{J}_y = \mathbf{J}(\mathbf{y}_n)$
- 13: Solve $\mathbf{J} \mathbf{L}_n = \mathbf{J}_y$
- 14: $\mathbf{M}_n = \mathbf{I} - \mathbf{L}_n$
- 15: $\mathbf{g}_n = \mathbf{M}_n \mathbf{h}_n$
- 16: Solve $(\mathbf{I} - \frac{\lambda}{\beta} \mathbf{M}_n) \mathbf{s}_n = \mathbf{g}_n$
- 17: Final correction $\Delta \mathbf{V}^{(n)} = -\mathbf{h}_n - \frac{1}{2\beta} \mathbf{s}_n$
- 18: Update $\mathbf{V}^{(n+1)} = \mathbf{V}^{(n)} + \Delta \mathbf{V}^{(n)}$
- 19: **if** $\|\Delta \mathbf{V}^{(n)}\|_{\infty} < \varepsilon$ **then**
- 20: **break**
- 21: **end if**
- 22: $n \leftarrow n + 1$
- 23: **end while**
- 24: **return** $\mathbf{V}^{(n)}$

3.1.3. CH-D: Chebyshev–Halley with Adaptive Damping

The standard CH formulation lacks global stabilization mechanisms, which limits its convergence under remote or ill-conditioned initializations. To address this, the CH-D variant proposed in this work incorporates two complementary stabilization strategies that are symmetrically applied to both methods:

- Adaptive damping. After computing the full correction $\Delta \mathbf{V}^{(n)}$ via Equation (18), the step is scaled as

$$\Delta \mathbf{V}^{(n)} \leftarrow \alpha \cdot \Delta \mathbf{V}^{(n)}, \quad \alpha = \min\left(1, \frac{\delta}{\|\Delta \mathbf{V}^{(n)}\|_\infty}\right), \quad \delta = 0.5.$$

When $\|\Delta \mathbf{V}^{(n)}\|_\infty \leq 0.5$, the factor $\alpha = 1$ and CH-D preserves its full cubic convergence rate. The damping is only activated when the corrections are explosively large, redirecting the iterate towards the convergence basin without altering the cubic convergence rate once the method enters it.

- Jacobian regularization. If $\text{rcond}(\mathbf{J}) < 10^{-14}$, a perturbation $+10^{-12}\mathbf{I}$ is added to $\mathbf{J}$ before factorization. This regularization is applied identically to NR and CH-D in every iteration, ensuring methodological symmetry between the two methods compared.

The overhead introduced by the damping operation is $\mathcal{O}(n)$ per iteration (one scalar division and one vector scaling), which is negligible compared to the $\mathcal{O}(n^3)$ LU factorization already performed. Under flat-start conditions, the damping factor satisfies $\alpha = 1$ in all iterations for all four test systems, so CH-D produces results identical to the standard CH, and the comparison tables under optimal initialization are not affected.

Notably, at each iteration, two linear systems are solved using the same Jacobian matrix $\mathbf{J}$, which allows reusing the LU factorization, thereby reducing the additional computational cost [20,30]. Such schemes have proven particularly effective in problems exhibiting strong nonlinearities [31].

3.2. Initial Conditions and Performance Metrics

To comprehensively assess the robustness of both numerical methods, three distinct types of initial conditions are defined:

- Optimal conditions: $V_i^{(0)} = 1.0$ p.u. for all load nodes. This represents an ideal starting point, close to the expected solution, and serves as a baseline for the best-case convergence behavior.
- Adverse conditions: $V_i^{(0)} = 0.5$ p.u., a value deliberately far from the expected solution. This scenario simulates cold-start situations or significantly erroneous voltage estimates, thereby testing the global convergence capabilities of each method.
- Random conditions: 100 independent replicates are generated with initial voltage values uniformly distributed in the $[0.8, 1.2]$ p.u. interval. This allows for statistical evaluations of method performance under random variations and provides insights into the typical convergence behavior.

For each combination of test systems, numerical methods, and initial conditions, the following key metrics are recorded:

- The number of iterations required to achieve convergence, defined as $\|\Delta \mathbf{V}^{(k)}\|_\infty < \varepsilon$. This metric serves as a direct indicator of the convergence speed of each method.
- The Final residual norm $\|\mathbf{F}(\mathbf{V}^{(k)})\|_\infty$, which verifies whether the obtained solution satisfies the power balance equations within the prescribed tolerance.

- The iteration-wise convergence rate, i.e., the evolution of $\|\mathbf{F}\|_\infty$ and $\|\Delta\mathbf{V}\|_\infty$ across iterations, visualized through convergence plots to enable a direct comparison of the convergence behavior of both methods.
- The total CPU time measured in milliseconds using MATLAB's `tic` and `toc` functions. A total of 1000 repetitions are performed for each scenario to obtain robust statistical metrics, including the minimum, median, mean, maximum, and standard deviation. Additionally, the computational cost per iteration is analyzed, considering that CH solves two linear systems with the same Jacobian matrix $\mathbf{J}$ per iteration, whereas NR solves only one.
- The nodal voltage profile, obtained from the final voltage vector $\mathbf{V}^*$, to verify the physical consistency of the solution and compare the results of both methods for each test network.
- The convergence failure rate, i.e., the percentage of executions that fail to converge within $t_{\max}$ iterations, a fundamental metric for evaluating the robustness of each method under adverse and random initial conditions.

3.3. Experimental Procedure and Statistical Analysis

The experiments are automated using MATLAB scripts that systematically iterate through all test systems (10, 21, 33, and 69 nodes) and initial conditions. For each scenario, 1000 repetitions of the corresponding method are executed, recording CPU times and verifying convergence. From the collected data, the following elements are constructed:

- Comparative tables summarizing the key metrics (number of iterations, residual norm, CPU time, and failure rate) for each system and initial condition.
- Convergence plots showing the evolution of $\|\mathbf{F}\|_\infty$ and $\|\Delta\mathbf{V}\|_\infty$ as a function of the iteration number, allowing for a visual assessment of each method's convergence rate.
- Nodal voltage profiles for each test network, comparing the results of both methods and verifying their equivalence within the established tolerance.
- Histograms and boxplots of CPU times to analyze the distribution and dispersion of computation times across repetitions.

The comparative analysis focuses on four key dimensions:

- **Computational efficiency:** The trade-off between the number of iterations and the total execution time, considering that CH solves two linear systems per iteration while NR solves one. A critical evaluation is conducted to determine whether the reduction in the iteration count justifies the additional cost.
- **Accuracy:** A comparison of the final residual norms and computed voltage profiles, verifying whether both methods achieve equivalent solutions within the established tolerance.
- **Robustness:** A comparison of failure rates under adverse and random initial conditions, characterizing the applicability of the CH family for monopolar DC networks of varying scales.
- **Scalability:** An analysis of method behavior as network size increases (from 10 to 69 nodes), examining how the CPU time, iteration count, and convergence rate evolve with problem dimension.

3.4. Results Validation

The validity of the implementations is verified through three complementary strategies:

- **Literature comparison:** Voltage profiles and residual norms are compared against reference solutions reported in the literature for the selected test systems [7,22,23].

- Internal consistency: The results obtained from both methods must be identical within the numerical tolerance, confirming that they converge to the same solution.
- Sensitivity analysis: A controlled variation of parameters (tolerance, initial conditions) is performed to observe the convergence behavior of each method under different settings.

This methodological framework ensures a fair, reproducible, and rigorous comparison between the NR and CH methods, providing quantitative insights to determine under which contexts the CH family offers significant advantages over the classical NR approach.

4. Test Systems

To evaluate the methodology developed in the previous section, four radial DC networks were employed, each fed by a single source. To guide the analysis gradually and didactically, 10- and 21-bus networks were first considered. The parameters of both networks were normalized to operate the system on a per-unit basis, adopting a power base of 1 MW. The topology of these systems is presented in Figures 1 and 2.

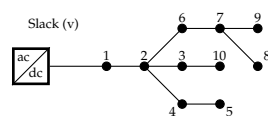


Figure 1. 10-bus network. Node 1 corresponds to the slack node (voltage source).

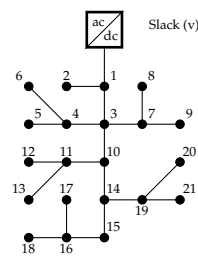


Figure 2. 21-bus network. Node 1 correspondes to the slack node (voltage source).

To enable a more exhaustive analysis and greater complexity in the comparative study, 33- and 69-bus networks are also considered, adopting base values of 1 MW and 12.66 kV. The corresponding topological layouts are shown in Figures 3 and 4.

Conversely, the electrical parameters and load data corresponding to each test network are summarized in the following tables. For each network, the tables provide the resistance values of the lines connecting the nodes, along with the active power demand at each node.

As shown in Table 1, the 10-bus network is characterized by short line resistances ranging from 0.0013 to 0.0050 Ω and nodal demands between 0.5 and 2.0 p.u., making it the most electrically compact system in this study.

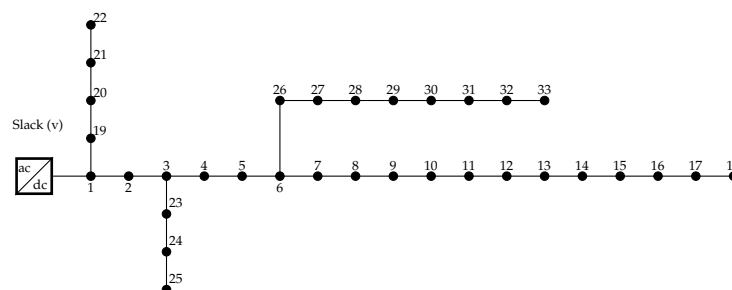


Figure 3. 33-bus network. Node 1 corresponds to the slack node (voltage source).

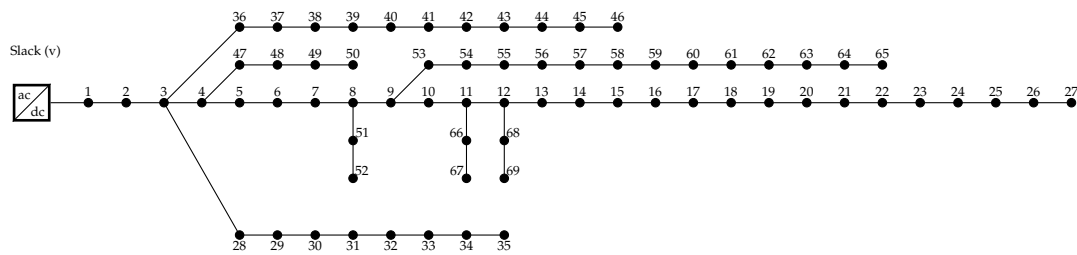


Figure 4. 69-bus network. Node 1 corresponds to the slack node (voltage source).

Table 1. Main parameters for the 10-bus network.

Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (p.u.)
1	2	0.0050	0
2	3	0.0015	0.8
2	4	0.0020	1.3
4	5	0.0018	0.5
2	6	0.0023	2.0
6	7	0.0017	0
7	8	0.0021	0
7	9	0.0013	0.7
3	10	0.0015	1.25

Table 2 describes the 21-bus network, which presents a more heterogeneous demand profile and a branched topology that results in a monotonically decreasing voltage profile, with all nodal voltages below the slack reference.

Table 2. Main parameters for the 21-bus network.

Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (p.u.)	Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (p.u.)
1	2	0.0053	0.7	11	12	0.0079	0.68
1	3	0.0054	0.0	11	13	0.0078	0.10
3	4	0.0054	0.36	10	14	0.0083	0.00
4	5	0.0063	0.04	14	15	0.0065	0.22
4	6	0.0051	0.36	15	16	0.0064	0.23
3	7	0.0037	0.00	16	17	0.0074	0.43
7	8	0.0079	0.32	16	18	0.0081	0.34
7	9	0.0072	0.80	14	19	0.0078	0.09
3	10	0.0053	0.00	19	20	0.0084	0.21
10	11	0.0038	0.45	19	21	0.0082	0.00

The 33-bus network, detailed in Table 3, introduces a wider range of line resistances (0.0922 to 1.7114 Ω) and higher power demands, representing a more complex radial distribution system with greater voltage drop along its main feeder.

Table 3. Main parameters for the 33-bus network.

Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (kW)	Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (kW)
1	2	0.0922	100	17	18	0.7320	90
2	3	0.4930	90	2	19	0.1640	90
3	4	0.3660	120	19	20	1.5042	90
4	5	0.3811	60	20	21	0.4095	90
5	6	0.8190	60	21	22	0.7089	90
6	7	0.1872	200	3	23	0.4512	90
7	8	1.7114	200	23	24	0.8980	420

Table 3. Cont.

Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (kW)	Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (kW)
8	9	1.0300	60	24	25	0.8900	420
9	10	1.0400	60	6	26	0.2030	60
10	11	0.1966	45	26	27	0.2842	60
11	12	0.3744	60	27	28	1.0590	60
12	13	1.4680	60	28	29	0.8042	120
13	14	0.5416	120	29	30	0.5075	200
14	15	0.5910	60	30	31	0.9744	150
15	16	0.7463	60	31	32	0.3105	210
16	17	1.2890	60	32	33	0.3410	60

Finally, Table 4 presents the largest system considered in this study. The 69-node network is characterized by a highly heterogeneous impedance profile, with resistances spanning from 0.0005 Ω in the head-end branches to values exceeding 1.5 Ω in the laterals, and includes a notably large concentrated load at node 61 (1244 kW).

Table 4. Main parameters for the 69-bus network.

Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (kW)	Node <i>i</i>	Node <i>j</i>	R_{ij} (Ω)	P_j (kW)
1	2	0.0005	0	3	36	0.0044	26
2	3	0.0005	0	36	37	0.0640	26
3	4	0.0015	0	37	38	0.1053	0
4	5	0.0215	0	38	39	0.0304	24
5	6	0.3660	2.6	39	40	0.0018	24
6	7	0.3810	40.4	40	41	0.7283	102
7	8	0.0922	75	41	42	0.3100	0
8	9	0.0493	30	42	43	0.0410	6
9	10	0.8190	28	43	44	0.0092	0
10	11	0.1872	145	44	45	0.1089	39.22
11	12	0.7114	145	45	46	0.0009	39.22
12	13	1.0300	8	4	47	0.0034	0
13	14	1.0440	8	47	48	0.0851	79
14	15	1.0580	0	48	49	0.2898	384.7
15	16	0.1966	45	49	50	0.0822	384.7
16	17	0.3744	60	8	51	0.0928	40.5
17	18	0.0047	60	51	52	0.3319	3.6
18	19	0.3276	0	9	53	0.1740	4.35
19	20	0.2106	1	53	54	0.2030	26.4
20	21	0.3416	114	54	55	0.2842	24
21	22	0.0140	5	55	56	0.2813	0
22	23	0.1591	0	56	57	1.5900	0
23	24	0.3463	28	57	58	0.7837	0
24	25	0.7488	0	58	59	0.3042	100
25	26	0.3089	14	59	60	0.3861	0
26	27	0.1732	14	60	61	0.5075	1244
3	28	0.0044	26	61	62	0.0974	32
28	29	0.0640	26	62	63	0.1450	0
29	30	0.3978	0	63	64	0.7105	227
30	31	0.0702	0	64	65	1.0410	59
31	32	0.3510	0	65	66	0.2012	18
32	33	0.8390	10	66	67	0.0047	18
33	34	1.7080	14	67	68	0.7394	28
34	35	1.4740	4	68	69	0.0047	28

5. Simulation Results

The computational implementation of both algorithms was carried out in MATLAB R2025b on a workstation equipped with an Intel Core i5-8265U processor (1.6 GHz), 12.0 GB of RAM, and a 64-bit version of Windows 10. The simulation parameters were kept identical for both methods in order to ensure a fair comparison: a convergence tolerance of $\varepsilon = 10^{-6}$, a maximum number of iterations $t_{\max} = 100$, a flat-start initialization $V_i^{(0)} = 1.0$ p.u. for all load nodes, and 1000 repetitions per scenario to obtain robust CPU time statistics.

5.1. 10-Bus Test System

Table 5 summarizes the performance metrics of both methods on the 10-bus radial DC network. Both methods converge to identical solutions, thereby confirming the internal consistency of the implementations. CH-D requires one fewer iteration than NR (two vs. three), reflecting its cubic convergence order. However, this reduction does not translate into shorter CPU times: the median CPU time for CH-D (0.0505 ms) is approximately 2.6 times higher than that of NR (0.0195 ms), given that each CH-D iteration solves two linear systems using the same Jacobian matrix [19].

Table 5. Comparative results—10-node system ($\varepsilon = 10^{-6}$, 1000 repetitions, flat start).

Metric	Newton–Raphson	Chebyshev–Halley-D
Iterations	3	2
$\ \mathbf{F}(\mathbf{V}^*)\ _{\infty}$ (p.u.)	8.44×10^{-6}	8.44×10^{-6}
$\ \Delta\mathbf{V}^*\ _{\infty}$ (p.u.)	1.55×10^{-7}	1.55×10^{-7}
$t_{\min}$ (ms)	0.0151	0.0441
t_{median} (ms)	0.0195	0.0505
t_{mean} (ms)	0.0247	0.0738
$t_{\max}$ (ms)	0.1569	0.4085
σ_t (ms)	0.0118	0.1913
Cost per iteration (ms)	0.0080	0.0253
Total losses (kW)	64.4750	64.4750
Convergence failures (%)	0	0

Table 6 details the evolution of the residual and the correction per iteration. In the first iteration, both methods yield comparable residuals (1.30 p.u.). In the second iteration, CH-D reduces the residual to 8.44×10^{-6} p.u., at which point $\|\Delta\mathbf{V}\|_{\infty}$ falls below the prescribed tolerance $\varepsilon = 10^{-6}$, whereas NR requires a third iteration. The condition number of the Jacobian remains stable around 142, indicating the favorable numerical sensitivity of this network. The adaptive damping factor α remains equal to 1 in both iterations under flat-start conditions, confirming that the CH-D correction step is identical to that of the standard CH when the initialization is well-conditioned.

Table 6. Iteration-wise details—10-bus system.

Iter.	Newton–Raphson		Chebyshev–Halley-D		$\kappa(\mathbf{J})$
	$\ \mathbf{F}\ _{\infty}$	$\ \Delta\mathbf{V}\ _{\infty}$	$\ \mathbf{F}\ _{\infty}$	$\ \Delta\mathbf{V}\ _{\infty}$	
1	1.30×10^0	1.99×10^{-2}	1.30×10^0	2.03×10^{-2}	140.27
2	2.32×10^{-2}	3.94×10^{-4}	8.44×10^{-6}	1.55×10^{-7}	142.82
3	8.44×10^{-6}	1.55×10^{-7}	—	—	142.87

Figure 5 shows two comparative graphs corresponding to the 10-bus test case: (a) the nodal voltage profile and (b) the convergence curves of the residual $\|\mathbf{F}\|_{\infty}$. The voltage

profiles obtained by both methods are identical, exhibiting a minimum voltage at bus 9 ($V_9 = 0.9797$ p.u.).

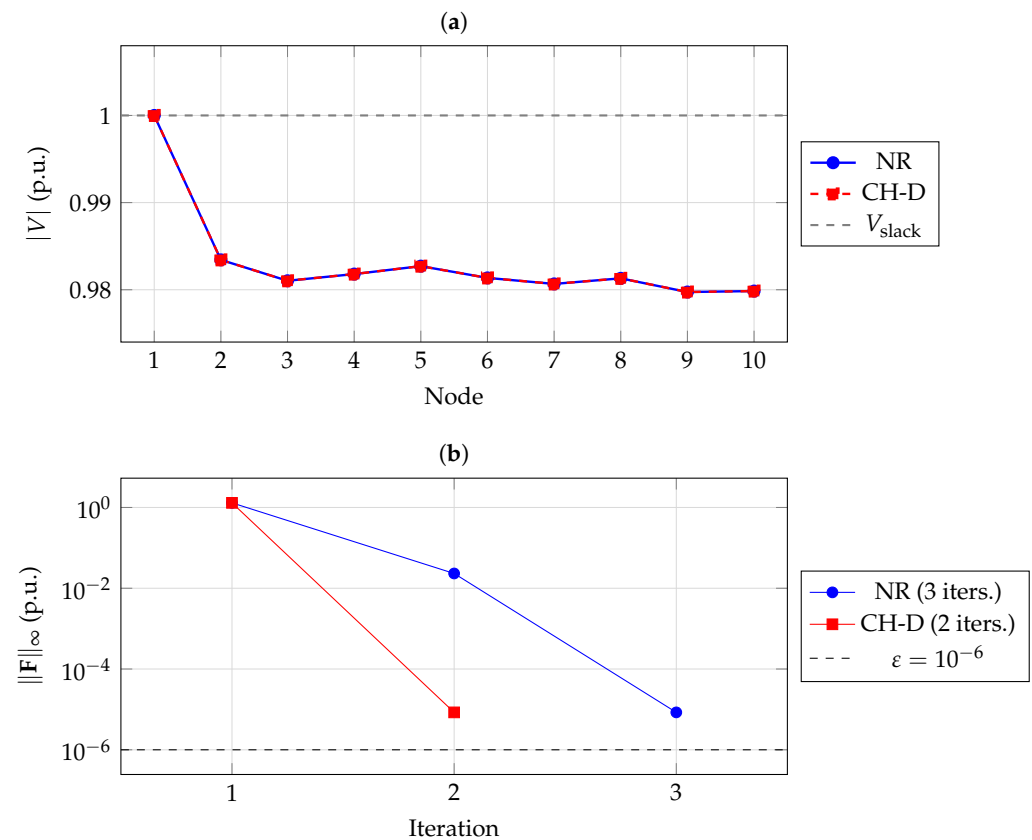


Figure 5. 10-bus network: (a) nodal voltage profile and (b) convergence curve of the residual $\|F\|_{\infty}$. The profiles from the NR and CH-D methods coincide perfectly.

5.2. 21-Bus Network

Table 7 summarizes the results obtained for the 21-bus radial DC network. With the CH-D variant (adaptive damping, $\delta = 0.5$; regularization $\text{rcond} < 10^{-14}$, $+10^{-12}\mathbf{I}$), both CH-D and NR converge in three iterations. The median computation time for CH-D (0.1660 ms) is 3.00 times greater than that for NR (0.0554 ms). The cost per iteration of CH-D (0.0553 ms) is 2.99 times that of NR (0.0185 ms). The Jacobian condition number, approximately 277, as shown in Table 8, reveals that this network is numerically more sensitive than the 10-bar network, although the stability of both methods is not affected.

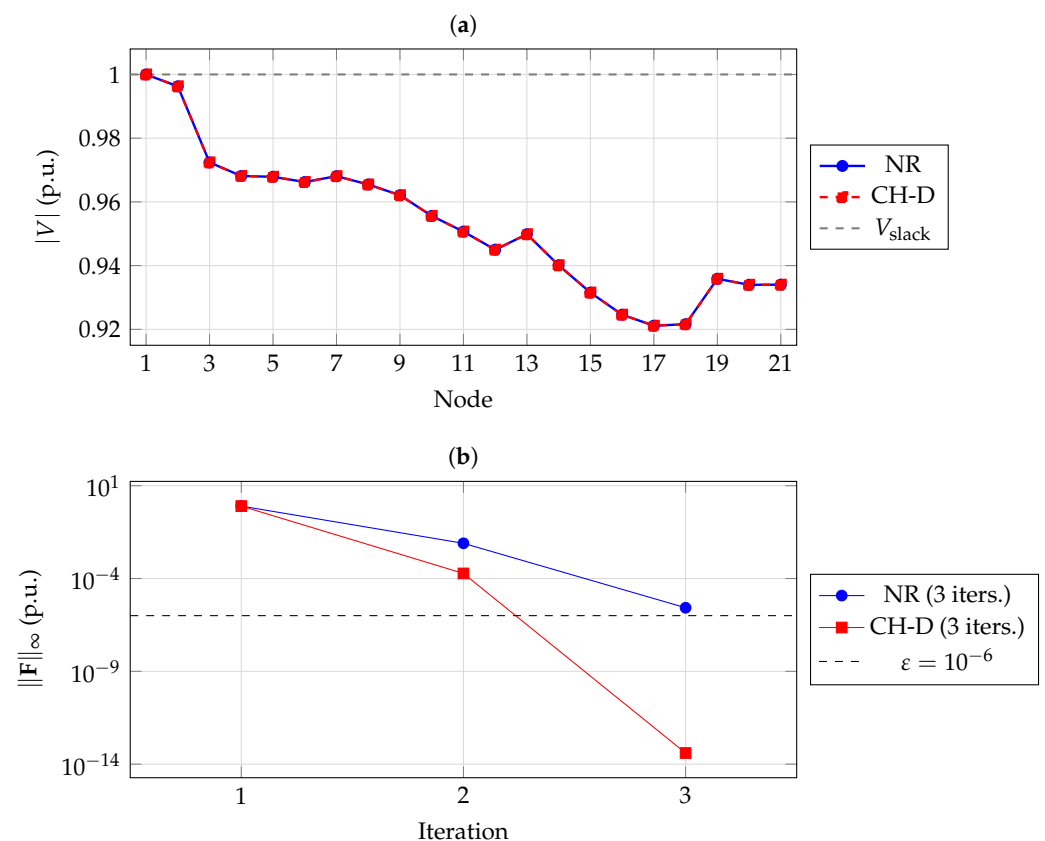
Table 7. Comparison of results for the 21-bus network ($\epsilon = 10^{-6}$, 1000 repetitions, flat start).

Metric	Newton–Raphson	Chebyshev–Halley-D
Iterations	3	3
$\ F(V^*)\ _{\infty}$ (p.u.)	2.67×10^{-6}	3.97×10^{-14}
$\ \Delta V^*\ _{\infty}$ (p.u.)	9.94×10^{-8}	2.59×10^{-16}
$t_{\min}$ (ms)	0.0502	0.1467
t_{median} (ms)	0.0554	0.1660
t_{mean} (ms)	0.0656	0.1801
$t_{\max}$ (ms)	2.5236	0.6696
σ_t (ms)	0.0809	0.0532
Cost per iteration (ms)	0.0185	0.0553
Total losses (kW)	276.0341	276.0341
Convergence failures (%)	0	0

Table 8. Iteration details—21-node network.

Iter.	Newton–Raphson		Chebyshev–Halley-D		$\kappa(J)$
	$\ F\ _{\infty}$	$\ \Delta V\ _{\infty}$	$\ F\ _{\infty}$	$\ \Delta V\ _{\infty}$	
1	8.00×10^{-1}	1.16×10^{-2}	8.00×10^{-1}	7.88×10^{-2}	250.17
2	7.90×10^{-3}	3.25×10^{-4}	1.88×10^{-4}	2.57×10^{-5}	276.96
3	2.67×10^{-6}	9.94×10^{-8}	3.97×10^{-14}	2.59×10^{-16}	276.97

The voltage profile of this network (see Figure 6) displays a monotonically decreasing pattern, with all nodal voltages below 1.0 p.u., consistent with a purely load network configuration in which all nodal injections are negative. The lowest voltage is observed at node 17 ($V_{17} = 0.9211$ p.u.), while the highest load-node voltage occurs at node 2 ($V_2 = 0.9963$ p.u.).

**Figure 6.** 21-bus network: (a) nodal voltage profile, and (b) convergence curve of the residual $\|F\|_{\infty}$.

5.3. 33-Node Network

Table 9 presents the results for the 33-node radial DC network. In this system, NR requires four iterations, while CH-D converges in three, maintaining the one-iteration reduction observed in the previous cases. The median time of CH-D (0.2141 ms) is approximately 1.92 times higher than that of NR (0.1118 ms), and the cost per iteration of the former (0.0714 ms) exceeds that of the latter (0.0279 ms) by a factor of 2.56, which is greater than in the smaller systems, reflecting the higher cost of operating with Jacobian matrices of dimension 32×32 . The iteration-wise residual evolution, detailed in Table 10, reveals that the cubic convergence of CH-D is most clearly observed in this network: between iterations 2 and 3, the residual drops eight orders of magnitude, from 2.80×10^{-5} p.u. to 3.54×10^{-13} p.u., while NR requires one additional step to reach 2.89×10^{-10} p.u. The Jacobian condition number stabilizes around 2075 from the second iteration onward,

notably higher than in the 10- and 21-bus systems—this is consistent with the greater electrical complexity of this network. In all cases, both methods produce identical total losses (135.2581 kW) and zero convergence failures, confirming the robustness of both implementations under flat-start conditions.

Table 9. Comparative results—33-bus network ($\varepsilon = 10^{-6}$, 1000 repetitions, flat start).

Metric	Newton–Raphson	Chebyshev–Halley-D
Iterations	4	3
$\ \mathbf{F}(\mathbf{V}^*)\ _{\infty}$ (p.u.)	2.89×10^{-10}	3.54×10^{-13}
$\ \Delta \mathbf{V}^*\ _{\infty}$ (p.u.)	1.13×10^{-10}	5.92×10^{-15}
$t_{\min}$ (ms)	0.0998	0.2222
t_{median} (ms)	0.1118	0.2463
t_{mean} (ms)	0.1207	0.2675
$t_{\max}$ (ms)	2.9341	6.0518
σ_t (ms)	0.1058	0.2414
Cost per iteration (ms)	0.0279	0.0821
Total losses (kW)	135.2581	135.2581
Convergence failures (%)	0	0

Table 10. Iteration details—33-bus network.

Iter.	Newton–Raphson		Chebyshev–Halley-D		$\kappa(\mathbf{J})$
	$\ \mathbf{F}\ _{\infty}$	$\ \Delta \mathbf{V}\ _{\infty}$	$\ \mathbf{F}\ _{\infty}$	$\ \Delta \mathbf{V}\ _{\infty}$	
1	4.20×10^{-1}	6.26×10^{-2}	4.20×10^{-1}	6.61×10^{-2}	1862.85
2	1.04×10^{-2}	3.48×10^{-3}	2.80×10^{-5}	1.10×10^{-5}	2062.73
3	2.80×10^{-5}	1.10×10^{-5}	3.54×10^{-13}	5.92×10^{-15}	2075.38
4	2.89×10^{-10}	1.13×10^{-10}	—	—	2075.42

The voltage profile shown in Figure 7 exhibits a progressive drop along the main feeder (nodes 1–18), with the minimum voltage occurring at node 18 ($V_{18} = 0.9339$ p.u.). The visible voltage jump between nodes 18 and 19 corresponds to the beginning of the secondary branch, which originates from nodes closer to the substation with lower accumulated voltage drops. The condition number of the Jacobian, approximately 2075, is significantly higher than in the 10- and 21-bus systems, reflecting the greater electrical complexity of this network.

5.4. 69-Node Network

Table 11 presents the results for the 69-bus radial DC network, the largest-scale system evaluated. The iteration pattern remains consistent with the 33-bus case (NR: 4, CH-D: 3). However, the absolute computation times increase substantially: the median time for NR is 0.4652 ms and 1.1325 ms for CH-D, a ratio of 2.44. The per-iteration cost of CH-D (0.3775 ms) is 3.24 times that of NR (0.1163 ms), the highest ratio observed among the four systems, reflecting the increased cost of operating with matrices of dimension 68×68 . The Jacobian condition number reaches values on the order of 6.25×10^5 , as can be seen in Table 12, notably higher than in the previous cases, which indicates a greater impedance contrast and a deeper network structure.

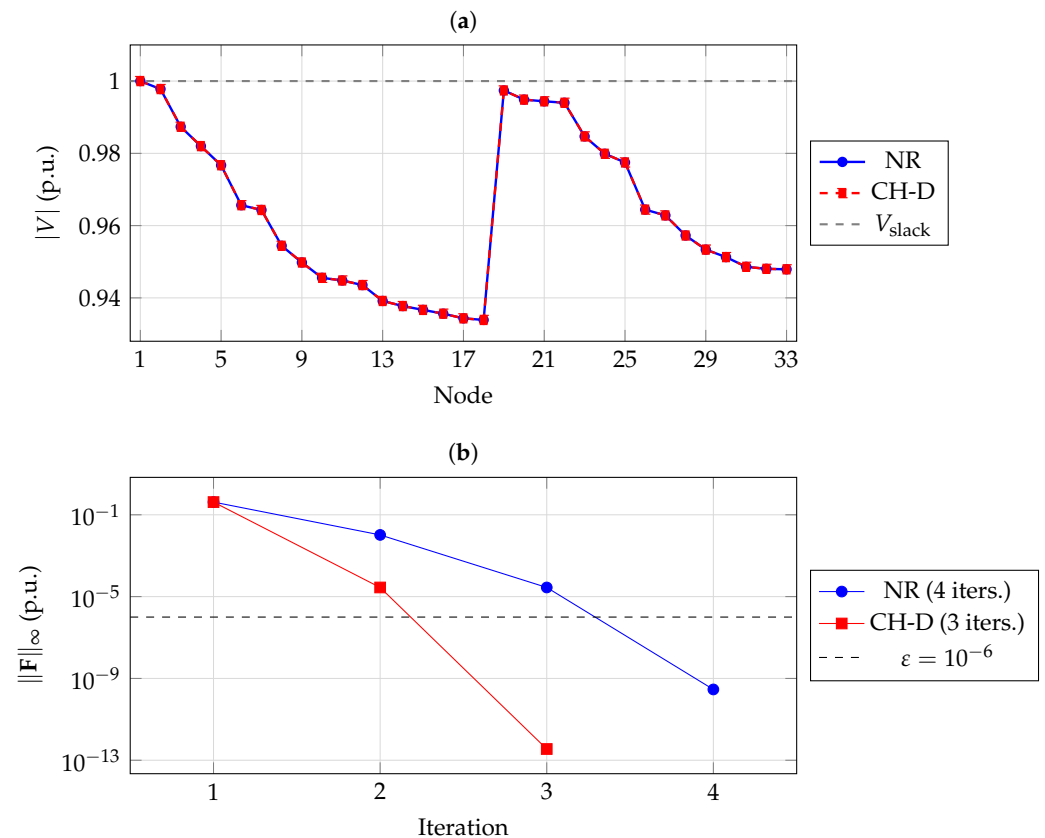


Figure 7. 33-node network: (a) nodal voltage profile and (b) convergence curve of the residual $\|F\|_{\infty}$. The voltage drop is more pronounced along the main branch (nodes 1–18), with the minimum at node 18 ($V_{18} = 0.9339$ p.u.). The steeper slope of CH reflects its cubic convergence.

Table 11. Comparison of results—69-bus network ($\epsilon = 10^{-6}$, 1000 repetitions, flat start).

Metric	Newton–Raphson	Chebyshev–Halley–D
Iterations	4	3
$\ F(V^*)\ _{\infty}$ (p.u.)	1.06×10^{-8}	5.82×10^{-11}
$\ \Delta V^*\ _{\infty}$ (p.u.)	6.26×10^{-10}	1.84×10^{-13}
$t_{\min}$ (ms)	0.3743	0.8872
t_{median} (ms)	0.4652	1.0668
t_{mean} (ms)	0.4979	1.1419
$t_{\max}$ (ms)	3.3754	5.2322
σ_t (ms)	0.1612	0.2709
Cost per iteration (ms)	0.1163	0.3556
Total losses (kW)	153.8475	153.8475
Convergence failures (%)	0	0

Table 12. Iteration details—69-bus network.

Iter.	Newton–Raphson		Chebyshev–Halley–D		$\kappa(J)$
	$\ F\ _{\infty}$	$\ \Delta V\ _{\infty}$	$\ F\ _{\infty}$	$\ \Delta V\ _{\infty}$	
1	1.24×10^0	6.78×10^{-2}	1.24×10^0	7.25×10^{-2}	6.25×10^5
2	7.98×10^{-2}	4.74×10^{-3}	4.07×10^{-4}	2.40×10^{-5}	6.25×10^5
3	4.07×10^{-4}	2.40×10^{-5}	5.82×10^{-11}	1.84×10^{-13}	6.25×10^5
4	1.06×10^{-8}	6.26×10^{-10}	—	—	6.25×10^5

The voltage profile of the 69-bus feeder (see Figure 8) exhibits the largest variation among the four systems studied, with the minimum voltage occurring at nodes 68 and 69

($V_{69} = 0.9274$ p.u.). The most severe drop occurs at nodes 60–65, associated with the largest load magnitude in the system (node 61, 1244 kW of active power).

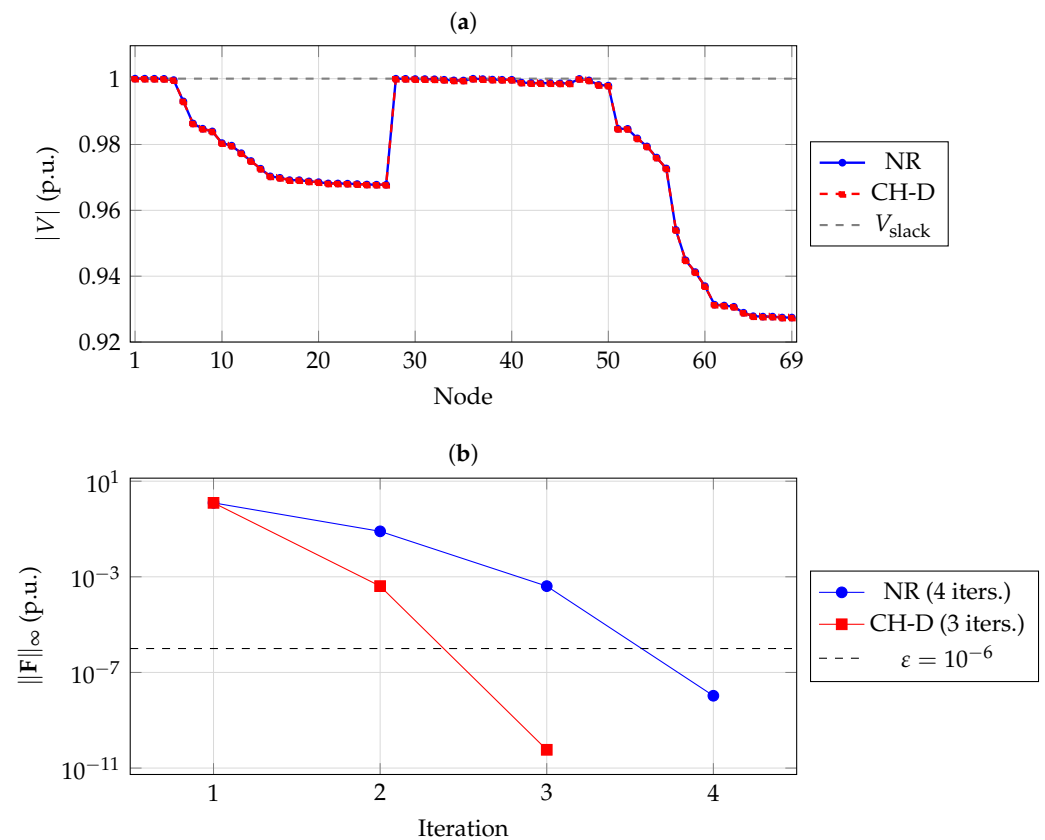


Figure 8. 69-node network: (a) nodal voltage profile and (b) convergence curve of the residual $\|F\|_{\infty}$. The most severe drop occurs at nodes 60–69, associated with the 1244 kW load at node 61. The steeper slope of CH-D between iterations 2 and 3 confirms its cubic convergence order.

5.5. Global Comparative Analysis

Table 13 consolidates the key metrics for the four evaluated systems. As the network size increases from 10 to 69 nodes, the number of iterations remains stable for both methods (NR: 3–4, CH-D: 2–3), confirming that convergence behavior is not significantly affected by network size under flat-start conditions. However, the absolute CPU times increase substantially: the median time for NR rises from 0.0195 ms (10 nodes) to 0.4652 ms (69 nodes), a factor of 23.8; for CH-D, it grows from 0.0467 ms to 1.1325 ms, a factor of 24.3. The similarity between these factors indicates that the scalability of both methods is comparable within the analyzed range of sizes. Furthermore, the CH-D/NR time ratio remains between 1.36 and 2.44 for the total median, but the cost-per-iteration ratio increases from 2.0 (10 nodes) to 3.24 (69 nodes), revealing that the relative overhead of the additional CH-D iteration grows with the network size. It is noteworthy that, for the 21-node system, the introduction of uniform Jacobian regularization ($\text{rcond} < 10^{-14}$, $+10^{-12}\mathbf{I}$) causes CH-D to require one additional iteration compared to the unregularized formulation (three vs. two), as the regularization slightly modifies the curvature operator $\mathbf{M}_n$. Nevertheless, CH-D achieves a final residual norm of 3.97×10^{-14} p.u., three orders of magnitude below NR's 2.67×10^{-6} p.u., confirming its superior accuracy.

Figure 9 illustrates the evolution of the median CPU time and the cost per iteration as a function of the network size. Both metrics exhibit superlinear growth when moving from 33 to 69 nodes, which is consistent with the quadratic complexity of constructing and factorizing the dense Jacobian matrix. The gap between CH-D and NR regarding the

cost per iteration widens with size, while the ratio of total medians remains approximately constant, suggesting that CH-D's advantage in the iteration count partially compensates for its higher per-iteration cost at larger scales.

Table 13. Global comparative summary of the four systems ($\varepsilon = 10^{-6}$, flat start).

System	Iterations		Median CPU (ms)		Cost/iter (ms)		Total Losses
	NR	CH-D	NR	CH-D	NR	CH-D	
10-node	3	2	0.0195	0.0467	0.0080	0.0162	64.475 kW
21-node	3	3	0.0554	0.1660	0.0185	0.0553	276.034 kW
33-node	4	3	0.1118	0.2141	0.0279	0.0714	135.2581 kW
69-node	4	3	0.4652	1.0668	0.1163	0.3556	153.8475 kW

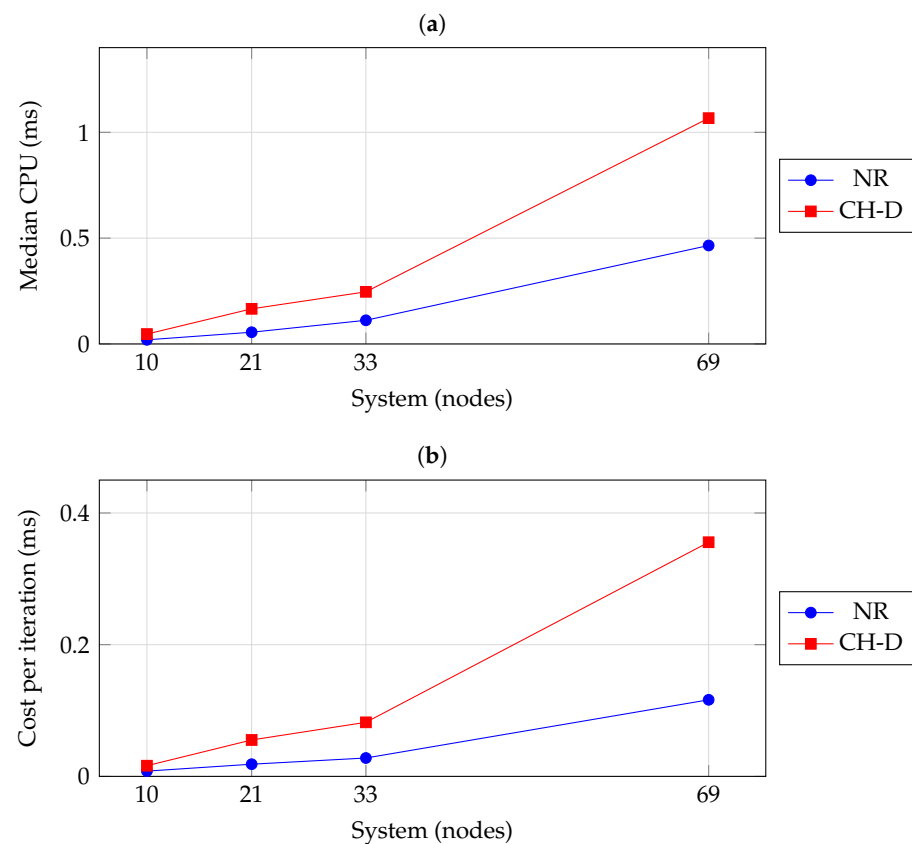


Figure 9. Scalability of NR and CH-D: (a) median CPU time and (b) cost per iteration as a function of network size. The gap in the cost per iteration between CH-D and NR widens as the system size increases.

In all scenarios evaluated under flat-start conditions, neither method exhibited convergence failures, confirming the robustness of both implementations for this type of initialization in small- to medium-scale radial DC networks. Furthermore, the calculated power losses and final residual norms are identical between methods in all cases, as reported individually in Tables 5, 7, 9 and 11, confirming that both implementations converge to the same physical operating point.

Theoretical Analysis of the Computational Cost

The observed growth in the cost-per-iteration ratio (from 2.0 at 10 nodes to 3.24 at 69 nodes) follows directly from the algorithmic structure of each method. NR requires one LU factorization ($\mathcal{O}(n^3)$) and one forward-backward substitution ($\mathcal{O}(n^2)$) per iteration. CH-D

reuses the same factorization but performs three substitutions: one each for $\mathbf{h}_n$, $\mathbf{L}_n$, and $\mathbf{s}_n$ —Equations (10), (13) and (16)—, plus the $\mathcal{O}(n^2)$ construction of $\mathbf{J}_y$.

For a small n , the shared $\mathcal{O}(n^3)$ factorization dominates and the CH-D/NR cost ratio approaches 2.0. As n increases, the additional $\mathcal{O}(n^2)$ operations become relatively more significant, explaining the rise to 3.24 at 69 nodes. This trend suggests that the ratio would stabilize near 3.0 for larger networks when sparse LU factorization is fully exploited [13], which is consistent with the superlinear growth observed in Figure 9.

5.6. Robustness Analysis Under Adverse and Random Initial Conditions

To evaluate the robustness of both methods, two additional initialization scenarios were applied to each system: an adverse start ($V_i^{(0)} = 0.5$ p.u.) with 1000 repetitions for benchmarking, and a random start ($V_i^{(0)} \sim \mathcal{U}[0.8, 1.2]$ p.u.) with 100 independent replicates using a fixed seed `rng(42)`.

Table 14 consolidates the metrics for adverse conditions across the four systems. In all cases where convergence was achieved, both methods reached the same solution as under the optimal condition, confirming the uniqueness of the operating point. The increase in the number of iterations relative to the flat start reflects that the point $V^{(0)} = 0.5$ p.u. lies outside the local convergence radius of both methods, requiring an initial phase of global descent before entering the superlinear convergence region. The cost per iteration remains practically constant between initial conditions for a given system, confirming that this parameter depends solely on the network size and not on the starting point.

Table 14. Comparison of optimal vs. adverse conditions for the four systems ($\varepsilon = 10^{-6}$, 1000 repetitions). All CH values correspond to the proposed CH-D adaptive damping variant.

Network	Metric	Newton–Raphson		Chebyshev–Halley-D	
		$V^{(0)} = 1.0$	$V^{(0)} = 0.5$	$V^{(0)} = 1.0$	$V^{(0)} = 0.5$
10-node	Iterations	3	12	2	3
	t_{median} (ms)	0.0195	0.0781	0.0467	0.0793
	t_{mean} (ms)	0.0247	0.0920	0.0503	0.1008
	Cost/iter (ms)	0.0080	0.0065	0.0162	0.0264
	Converged	Yes	Yes	Yes	Yes
21-node	Iterations	3	45	3	5
	t_{median} (ms)	0.0554	1.1717	0.1660	0.3170
	t_{mean} (ms)	0.0656	1.3700	0.1801	0.3712
	Cost/iter (ms)	0.0185	0.0260	0.0553	0.0634
	Converged	Yes	Yes	Yes	Yes
33-node	Iterations	4	43	3	4
	t_{median} (ms)	0.1118	1.1678	0.2141	0.3657
	t_{mean} (ms)	0.1207	1.2374	0.2421	0.4121
	Cost/iter (ms)	0.0279	0.0272	0.0821	0.0914
	Converged	Yes	Yes	Yes	Yes
69-node	Iterations	4	50	3	100 (no conv.)
	t_{median} (ms)	0.4652	6.0259	1.1325	38.7077
	t_{mean} (ms)	0.4979	6.3099	1.2005	39.0525
	Cost/iter (ms)	0.1163	0.1205	0.3775	0.3871
	Converged	Yes	Yes	Yes	No

For the 10-node network, Table 15 details the residual evolution per iteration under adverse conditions. In NR, the residual grows from 5.0×10^1 to 2.7×10^5 p.u. in the second iteration before decreasing monotonically, a typical behavior of Newton’s method outside the quadratic convergence radius. CH-D achieves the tolerance in only three iterations

compared to NR's 12, confirming that the adaptive damping mechanism ($\alpha_1 = 0.0272$) successfully limits the first correction to $\|\Delta \mathbf{V}\|_\infty = 0.5$ p.u., after which the method recovers cubic convergence and reaches the tolerance in two additional steps, as reported in Table 14.

Table 15. Iteration details under adverse condition—10-bus network.

Iter.	Newton–Raphson		Chebyshev–Halley-D		
	$\ \mathbf{F}\ _\infty$	$\ \Delta \mathbf{V}\ _\infty$	$\ \mathbf{F}\ _\infty$	$\ \Delta \mathbf{V}\ _\infty$	α
1	5.00×10^1	3.68×10^1	5.00×10^1	5.00×10^{-1}	0.0272
2	2.75×10^5	1.84×10^1	1.36×10^0	1.90×10^{-2}	1.0000
3	6.87×10^4	9.20×10^0	7.55×10^{-6}	1.14×10^{-7}	1.0000
4	1.72×10^4	4.59×10^0	—	—	—
5	4.28×10^3	2.28×10^0	—	—	—
6	1.06×10^3	1.12×10^0	—	—	—
7	2.52×10^2	5.12×10^{-1}	—	—	—
8	5.29×10^1	1.86×10^{-1}	—	—	—
9	6.90×10^0	3.32×10^{-2}	—	—	—
10	2.17×10^{-1}	1.13×10^{-3}	—	—	—
11	2.46×10^{-4}	1.30×10^{-6}	—	—	—
12	3.23×10^{-10}	1.73×10^{-12}	—	—	—

Table 16 summarizes the results for the random condition across the four systems. For the 10-node network, NR exhibits a failure rate of 76% and CH-D reports 41%. The 21- and 33-bus networks reveal a markedly different convergence behavior between NR and CH-D: in the 21-bus system, NR fails in 90% of the replicates while CH-D converges in 57 out of 100 (43% failure rate), a substantial improvement attributable to the adaptive damping mechanism that prevents divergence under moderately perturbed initializations; in the 33-bus network, NR reaches a 97% failure rate while CH-D achieves convergence in 13 out of 100 replicates (87% failure rate), with a mean of 44.46 iterations among the successful cases—compared to 74.33 for NR. The 69-node network represents the most severe case: no replicate converges with either method (100% failure rate), demonstrating that the intrinsic ill-conditioning of the Jacobian of this network—originating from head-end admittances on the order of 3×10^5 p.u.—places the problem outside the convergence radius of both methods when the initialization deviates from the nominal profile.

Table 16. Global robustness summary under random conditions ($V_i^{(0)} \sim \mathcal{U}[0.8, 1.2]$, 100 replicates, rng(42)).

System	Conv. / 100		Failure Rate (%)		Mean Iter. (Conv.)	
	NR	CH	NR	CH	NR	CH
10-node	24	59	76.0	41.0	9.46	8.07
21-node	10	57	90.0	43.0	15.30	22.35
33-node	3	13	97.0	87.0	74.33	44.46
69-node	0	0	100.0	100.0	—	—

Figure 10 illustrates two key aspects of robustness for the 10-bus network: (a) convergence under adverse conditions and (b) the dispersion of voltage profiles from the converged replicates under random conditions. The initial increase of the residual in both methods before the monotonic descent phase confirms the behavior outside the local convergence radius. The converged replicates all reach the same reference profile, confirming the uniqueness of the solution.

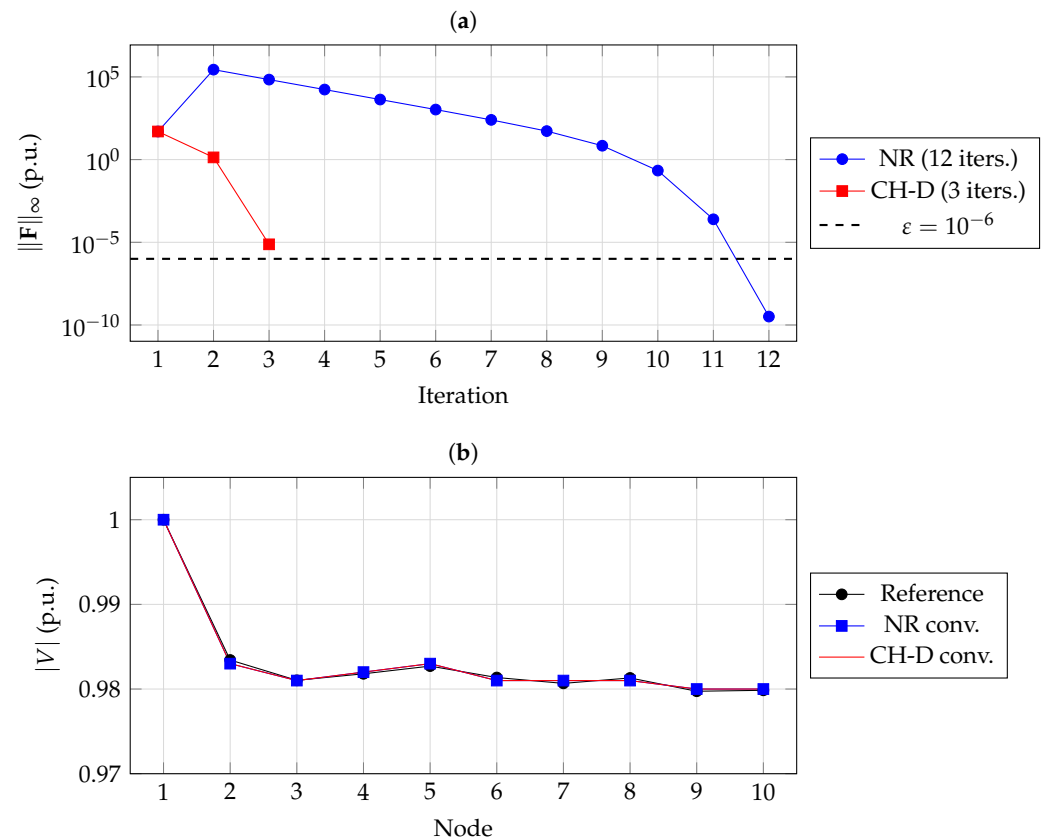


Figure 10. 10-node network—robustness. (a) Residual convergence under adverse conditions ($V^{(0)} = 0.5$ p.u.) for NR (12 iterations) and CH-D (3 iterations). The adaptive damping factor $\alpha_1 = 0.0272$ limits the first correction to $\|\Delta V\|_{\infty} = 0.5$ p.u. and allows recovering cubic convergence from iteration 2 onward. (b) Dispersion of voltage profiles from converged replicates under random conditions ($\mathcal{U}[0.8, 1.2]$ p.u.). The 24 converged NR replicates and 59 converged CH-D replicates all attain the same reference profile ($V^{(0)} = 1.0$ p.u., black line).

Figure 11 presents the equivalent results for the 21-bus network. Under adverse conditions, NR required 45 iterations while CH-D converged in only five. The iteration-wise damping factors confirm the stabilization mechanism: $\alpha_1 \approx 0$ at the first iteration limits the correction to $\|\Delta V\|_{\infty} = 0.5$ p.u., $\alpha_2 = 0.2095$ further stabilizes the second step, and from iteration 3 onward $\alpha = 1$, recovering the full cubic convergence rate and reaching a final residual of 8.40×10^{-14} p.u. Under random conditions, 10 out of 100 replicates converged with NR and 57 with CH-D, with means of 15.30 and 22.35 iterations, respectively, demonstrating that CH-D substantially extends the convergence basin under random initialization for this network.

Figure 12 presents the results for the 33-bus network. Under adverse conditions, NR required 43 iterations, while CH-D converged in only four. The iteration-wise damping factors confirm the stabilization mechanism: $\alpha_1 \approx 0$ at the first iteration limits the correction to $\|\Delta V\|_{\infty} = 0.5$ p.u. despite the extreme Jacobian ill-conditioning ($\kappa(J) \approx 1.36 \times 10^{15}$), and, from iteration 2 onward, $\alpha = 1$, recovering the full cubic convergence rate. The residual drops from 8.32 p.u. at iteration 2 to 2.59×10^{-3} p.u. at iteration 3, and to 3.54×10^{-13} p.u. at iteration 4, which is consistent with the cubic convergence order of the CH-D method. Under random conditions, convergence becomes extremely sensitive to initialization: NR only converges in three out of 100 replicates (97% failure rate), while CH-D achieves convergence in 13 out of 100 replicates (87% failure rate), with means of 74.33 and 44.46 iterations, respectively, confirming the robustness advantage of the proposed variant.

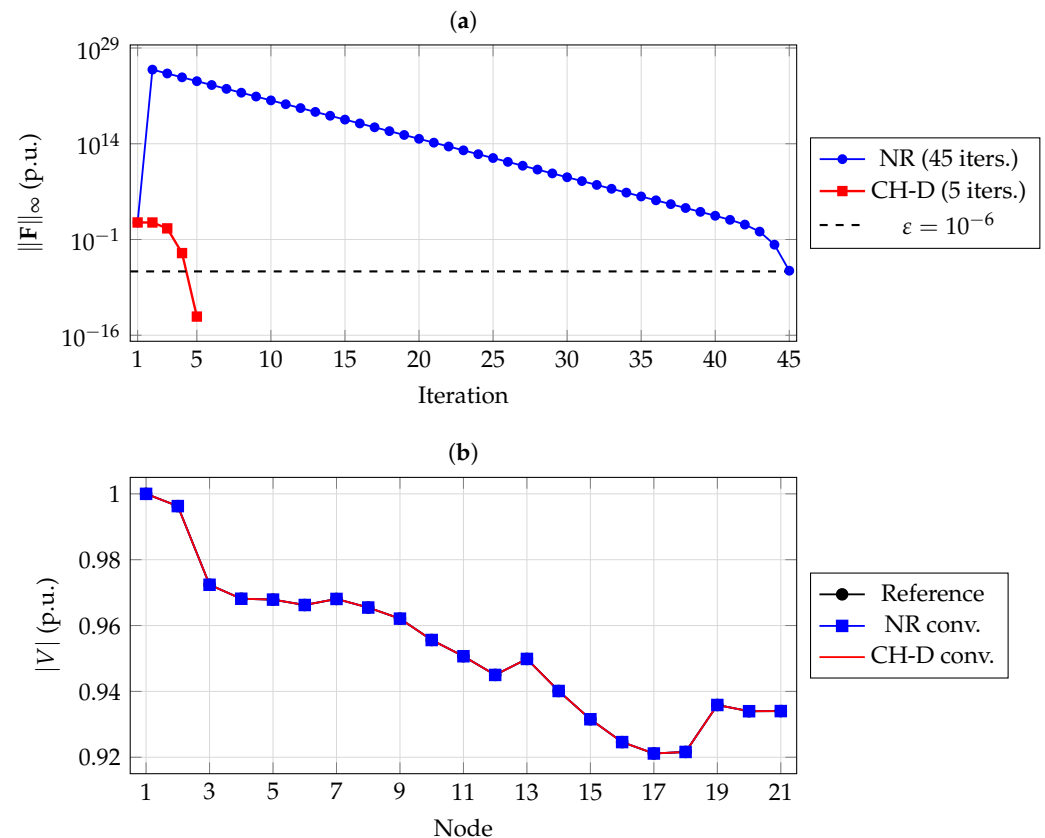


Figure 11. 21-node network—robustness. (a) Residual convergence under adverse conditions ($V^{(0)} = 0.5$ p.u.) for NR (45 iterations) and the proposed CH-D variant (five iterations). In the first iteration the damping factor $\alpha_1 \approx 0$ controls the explosive correction ($\|\Delta V\|_{\infty} = 0.5$ p.u. after damping). In the second iteration, $\alpha_2 = 0.2095$ further stabilizes the step, and, from iteration 3 onward, $\alpha = 1$ and cubic convergence is fully recovered, reaching a final residual of 8.40×10^{-14} p.u. (b) Dispersion of voltage profiles from converged replicates under random conditions ($\mathcal{U}[0.8, 1.2]$ p.u.). The ten converged NR replicates and the 57 converged CH-D replicates all attain the same reference profile ($V^{(0)} = 1.0$ p.u., black line).

Numerically speaking, the 69-node network is the most severe case in this study. Under adverse conditions, NR converges in 50 iterations ($t_{\text{med}} = 6.03$ ms), while CH-D exhausts the limit of 100 iterations without reaching the tolerance. Under random conditions, neither method converges in any of the 100 replicates. In this regard, Table 17 summarizes the key indicators, and Table 18 details the residual evolution of NR under adverse conditions. It is worth noting that the adverse and random scenarios fail—or succeed—through different mechanisms, as shown by the Jacobian condition number at the first iteration (Table 17): $\kappa(J) \approx 3.75 \times 10^{15}$ under the adverse start versus $\kappa(J) \approx 7.23 \times 10^5$ under random initialization, ten orders of magnitude better conditioned. The adverse case is a numerical problem: the uniform offset $V^{(0)} = 0.5$ p.u. preserves relative voltage differences between nodes, so Jacobian regularization stabilizes the first step and NR converges monotonically thereafter. The random case is a structural one: the Jacobian is never ill-conditioned, so regularization has nothing to correct, and failure instead comes from uncorrelated node-to-node perturbations narrowing Newton’s attraction basin near the concentrated load at node 61.

The CH-D variant with $\delta = 0.5$ does not converge under adverse initialization in the 69-node network, since the Jacobian ill-conditioning $\kappa(J) \approx 3.75 \times 10^{15}$ generates correction steps that exceed the damping threshold. A reduced damping parameter ($\delta = 0.01$) was

tested, but it also failed to converge within 100 iterations, confirming that the adverse 69-node case lies outside the stabilization capacity of the current CH-D formulation.

The head branches 1–2 and 2–3 have $R = 0.0005 \Omega$, which, with a base impedance $Z_b = 160.28 \Omega$, produces admittances as high as $y \approx 3.2 \times 10^5$ p.u. These dominate the diagonal of $\mathbf{G}$ and generate $\kappa(\mathbf{J}) \approx 3.75 \times 10^{15}$ at the first iteration with $V^{(0)} = 0.5$ p.u., a value that exceeds the inverse of machine epsilon ($\epsilon_{\text{mach}} \approx 2.2 \times 10^{-16}$): the matrix is effectively singular at machine precision from this starting point.

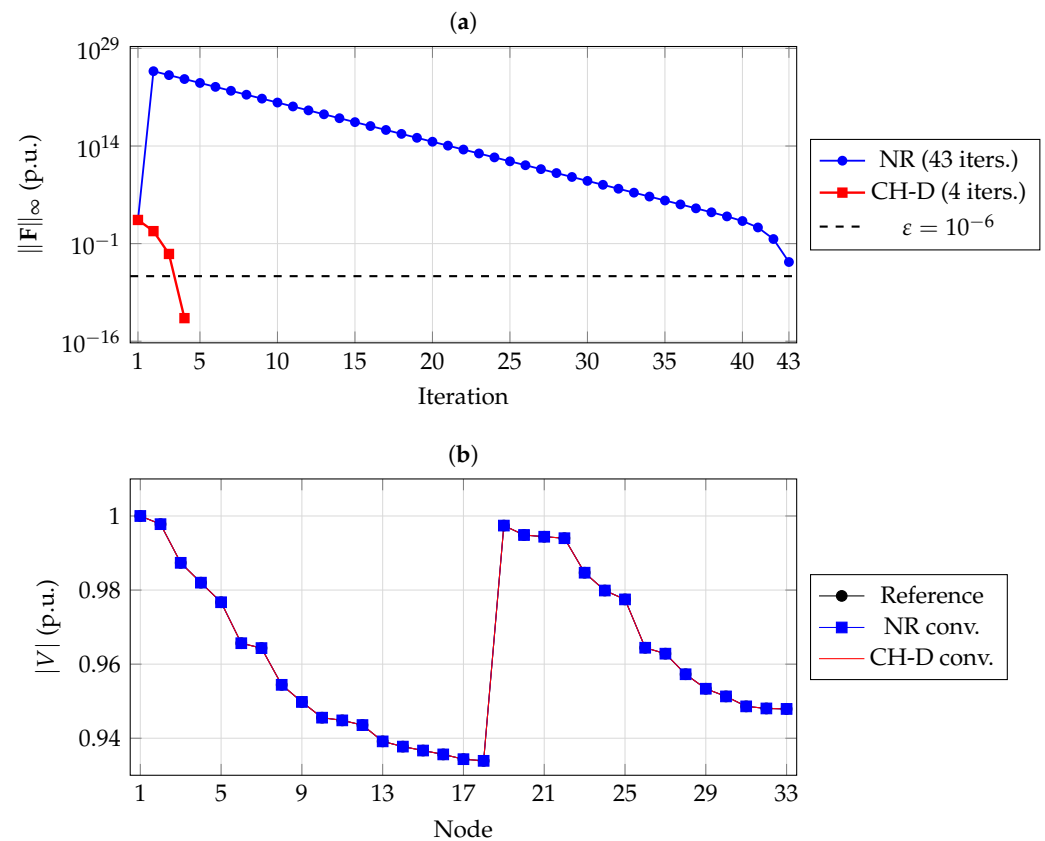


Figure 12. 33-node network—robustness. (a) Residual convergence under adverse conditions ($V^{(0)} = 0.5$ p.u.) for NR (43 iterations) and the proposed CH-D variant (four iterations). In the first iteration, the damping factor $\alpha_1 \approx 0$ limits the correction to $\|\Delta \mathbf{V}\|_\infty = 0.5$ p.u., preventing the iterate from leaving the convergence basin despite the extreme ill-conditioning ($\kappa(\mathbf{J}) \approx 1.36 \times 10^{15}$). From iteration 2 onward, $\alpha = 1$, and cubic convergence is fully recovered, reaching a final residual of 3.54×10^{-13} p.u. in only four iterations. (b) Dispersion of voltage profiles from converged replicates under random conditions ($\mathcal{U}[0.8, 1.2]$ p.u.). The three converged NR replicates and the 13 converged CH-D replicates all attain the same reference profile ($V^{(0)} = 1.0$ p.u., black line).

Table 17. Comparison: NR vs. CH-D in the 69-bus network (✓ conv.; × no conv. in 100 iters.).

Metric	Adverse ($V_0 = 0.5$ p.u.)		Random ($\mathcal{U}[0.8, 1.2]$)	
	NR	CH	NR	CH
Converged	✓	×	×	×
Iterations	50	100 (no)	100 (no)	100 (no)
t_{med} (ms)	6.03	38.71	36.37	62.97
Cost/iter (ms)	0.1205	0.3871	—	—
Failure rate (%)	0	100	100	100
$\kappa(\mathbf{J})$ iter. 1	3.75×10^{15}	3.75×10^{15}	$\sim 7.23 \times 10^5$	$\sim 7.23 \times 10^5$

Table 18. NR history—Adverse conditions ($V_0 = 0.5$ p.u.), representative iterations for the 69-bus feeder.

Iter.	$\ F(V)\ _\infty$	$\ \Delta V\ _\infty$	$\kappa(J)$
1	8.014×10^4	1.187×10^{13}	3.749×10^{15}
2	4.520×10^{31}	5.937×10^{12}	7.229×10^5
10	6.897×10^{26}	2.319×10^{10}	7.229×10^5
20	6.577×10^{20}	2.265×10^7	7.229×10^5
30	6.273×10^{14}	2.212×10^4	7.229×10^5
40	5.982×10^8	2.160×10^1	7.193×10^5
48	8.602×10^2	3.465×10^{-3}	6.252×10^5
49	2.284×10^0	1.083×10^{-5}	6.255×10^5
50	1.627×10^{-5}	1.001×10^{-10}	6.255×10^5

Despite the initial ill-conditioning, NR manages to converge in 50 iterations thanks to Jacobian regularization (threshold $\text{rcond} < 10^{-14}$ with a $+10^{-12}\mathbf{I}$ perturbation), which dampens the first step and brings the iterate to a region where $\kappa(J) \approx 7.23 \times 10^5$ and the convergence is monotonic. CH-D, in contrast, cannot benefit from this regularization in the same way: its first step requires solving three linear systems with the same singular J , which amplifies the error up to $\|\Delta V\|_\infty \approx 1.19 \times 10^{13}$ p.u., displacing the iterate to a region of no return and causing non-convergence within the maximum of 100 iterations.

For NR, the problem is not the extreme ill-conditioning observed under adverse conditions: with $V^{(0)} \sim \mathcal{U}[0.8, 1.2]$ p.u., $\kappa(J)$ at the first iteration is on the order of 7.23×10^5 , which is numerically manageable. The cause of failure is structural: the concentrated load at node 61 ($P_{61} = 1244$ kW, the largest in the system) severely narrows the Newton attraction basin, so that perturbations of only ± 0.2 p.u. in $V^{(0)}$ are sufficient to place the initial iterate outside it. The value of $\kappa(J)$ remains practically constant between iterations 2 and 100 ($\approx 7.23 \times 10^5$), confirming that the method orbits the same region without approaching the solution. For CH-D, the cause of failure is different and more severe: its third-order operator introduces two additional matrix solves per iteration (increasing the mean time by approximately 26.6 ms compared to NR) without reducing the residual—since starting from a point outside the convergence basin causes the curvature step to amplify the perturbation rather than correct it. Consequently, CH-D proves more unstable than NR under random conditions in this network, with a 100% failure rate compared to the same 100% for NR, albeit with significantly longer computation times ($t_{\text{med,CH}} = 62.97$ ms vs. $t_{\text{med,NR}} = 36.37$ ms).

To ensure the convergence of both methods in the 69-bus network under remote initializations, a hybrid strategy is recommended: first execute damped NR steps until the quadratic convergence basin is entered, and only then activate the curvature step of CH-D to exploit its higher convergence order. Alternatively, the use of an informed initial estimate (e.g., the solution of a linearized power flow) can eliminate the remote start problem for both methods.

6. Discussion and Limitations

The results reveal five concrete limitations that bound the scope of this work. First, both methods are formulated exclusively for strictly radial DC networks with a single slack node. In practice, most real HVDC and LVDC systems operate with meshed topologies or multiple sources, which places the majority of real-world applications outside the scope of the current formulation.

Second, the standard CH formulation lacks global stabilization mechanisms, a limitation that motivated the development of the CH-D variant proposed in this work. CH-D

incorporates an adaptive damping factor $\alpha = \min(1, \delta / \|\Delta \mathbf{V}\|_\infty)$ with $\delta = 0.5$, combined with uniform Jacobian regularization ($\text{rcond} < 10^{-14}$, $+10^{-12}\mathbf{I}$) applied symmetrically to both NR and CH-D, thereby addressing the methodological asymmetry identified during our review. Under this configuration, CH-D successfully converges under adverse conditions for the 10-, 21-, and 33-bus systems, requiring only three, five, and four iterations, respectively, compared to the 12, 45, and 43 iterations required by NR. However, in the 69-bus system under adverse initialization ($V^{(0)} = 0.5$ p.u.), the Jacobian ill-conditioning ($\kappa(\mathbf{J}) \approx 3.75 \times 10^{15}$) generates correction steps that exceed the stabilization capacity of the damping with $\delta = 0.5$, causing CH-D to fail even with regularization. A reduced parameter ($\delta = 0.01$) was also tested, but did not achieve convergence within 100 iterations, confirming that the adverse 69-node case presents a structural challenge beyond the reach of the current CH-D formulation. In the 69-bus feeder under adverse initialization ($V^{(0)} = 0.5$ p.u.), even with CH-D, the presence of head-end branches with $R = 0.0005 \Omega$ raises $\kappa(\mathbf{J})$ to 3.75×10^{15} , effectively rendering the Jacobian singular at machine precision. CH-D amplifies this error across three linear solves per iteration, producing corrections of $\|\Delta \mathbf{V}\|_\infty \approx 10^{13}$ p.u. that displace the iterate beyond any recoverable region. It should be noted that this asymmetry does not reflect a general superiority of NR, but rather a specific engineering mechanism absent from the current CH-D implementation at $\delta = 0.5$, one that vanishes in well-conditioned systems.

Third, sensitivity to initialization is not exclusive to either method: both are local convergence methods whose effectiveness critically depends on the quality of the initial guess. In the absence of a robust initialization strategy, neither method ensures convergence under moderate perturbations of the nominal profile, which limits their direct applicability in operational scenarios with high uncertainty in the initial system state.

Fourth, this study evaluates only the super-Halley variant ($\lambda = 1$, $\beta = 2/3$). Other members of the CH family, such as Chebyshev ($\alpha = 0$) and Halley ($\alpha = 1/2$), may exhibit different behaviors with respect to ill-conditioning and robustness under remote initializations. A systematic comparison across variants remains an open question that could identify more suitable parameter configurations for specific network types.

Finally, this study is restricted to networks of up to 69 nodes with constant power loads. Modern DC distribution systems incorporate loads with significant temporal variability, associated with PV converters, batteries, and EVs, whose impact on the convergence of both methods has not been evaluated and represents a relevant limitation for real-world applications.

7. Conclusions and Future Work

This study presents a systematic and quantitative comparison between the NR method and the CH multipoint family for solving the power flow problem in monopolar radial DC networks, evaluated on four test systems of increasing complexity under three initialization scenarios.

Regarding the order of convergence, under flat-start conditions, CH-D requires the same or fewer iterations than NR. Specifically, CH-D converges in two iterations for the 10-bus network and in three for the 21-, 33-, and 69-bus feeders, compared to the three and four iterations required by NR for the smaller and larger systems, respectively. For the 21-node system, the introduction of uniform Jacobian regularization caused CH-D to require one additional iteration compared to the unregularized formulation (three vs. two), as discussed in Section 5. This advantage was further confirmed in the final residual norms: in larger-scale systems, CH achieved residuals between two and three orders of magnitude lower than those obtained by NR, even though both methods satisfy the prescribed tolerance.

Regarding computational cost, the reduction in the number of iterations does not translate into an advantage in CPU time. The median computation time of CH is between 1.36 and 2.44 times higher than that of NR, and the cost-per-iteration ratio increases from 2.0 to 3.24 when moving from ten to 69 nodes, driven by the need to solve two linear systems per iteration and by the increasing cost of LU factorization. For small- and medium-scale networks under well-conditioned initialization, NR remains the most computationally efficient option.

In terms of robustness, under adverse initialization ($V^{(0)} = 0.5$ p.u.), both methods converge for the 10-, 21-, and 33-bus systems, with CH requiring approximately 60% of the iterations needed by NR. However, in the 69-node feeder, the severe ill-conditioning of the Jacobian ($\kappa(J) \approx 3.75 \times 10^{15}$) prevents CH from converging, whereas NR recovers through an implicit regularization mechanism. Under random initialization, both methods exhibit high failure rates across all systems, reaching 100% in the 69-bus network, which confirms that their effectiveness critically depends on the quality of the initial guess.

In summary, the CH family offers a reproducible advantage in terms of convergence order and final solution accuracy, but this advantage becomes computationally relevant only in applications requiring high precision or where the additional per-iteration cost can be reduced through algorithmic improvements. The CH-D variant introduced in this work demonstrates that adaptive damping with $\alpha = \min(1, 0.5/\|\Delta V\|_\infty)$ substantially extends the convergence basin of the CH family, improving robustness under adverse initialization. For routine power flow analysis in small- and medium-scale radial DC networks under flat-start conditions, NR remains the most computationally efficient option. However, CH-D is recommended when initializations deviate significantly from the nominal profile and high solution accuracy is required.

The results and limitations identified in this study motivate several research directions. First, an extension to meshed DC networks with multiple sources would be interesting. This work is restricted to strictly radial topologies with a single supply source. Therefore, extending the CH formulation to meshed configurations while addressing the challenges involved constitutes a direct and necessary continuation of this work.

Second, the high failure rates observed under random initialization, as well as the complete failure in the 69-node network under adverse conditions, demonstrate that the convergence of both methods critically depends on the quality of the initial guess. Strategies based on linearized power flow solutions, graph-theoretic estimates, or optimization-based initializations represent a parallel path for improving performance.

Third, a scalability analysis for large-scale networks is recommended. The cost-per-iteration ratio between CH and NR increases from 2.0 to 3.24 in the range of ten to 69 nodes, suggesting a trend that may become more pronounced in systems with hundreds or thousands of nodes. A dedicated study on larger networks, exploiting the sparse structure of the Jacobian, would help to determine whether the iteration reduction of CH compensates for its higher per-step cost at truly large scales.

Finally, future work should evaluate the robustness and performance of both methods under more demanding and realistic operational scenarios. This includes heavy load conditions, post-fault restoration processes, parametric uncertainties regarding line resistances and load models, and measurement errors in nodal power injections, all of which are common in the actual operation of DC distribution networks and may significantly affect the conditioning of both methods' Jacobian matrix and convergence basin. Additionally, since loads associated with PV converters, batteries, and EV in modern DC distribution systems exhibit significant temporal variability, assessing both methods under variable or probabilistically sampled load profiles would provide a more realistic evaluation of the applicability of the CH-D variant in active DC networks.

Author Contributions: Conceptualization, methodology, software, and writing—review and editing: S.S.-M., J.D.P.-D., and O.D.M. All authors have read and agreed to the published version of the manuscript.

Funding: The authors acknowledge the support provided by Thematic Network 723RT0150, i.e., Red para la integración a gran escala de energías renovables en sistemas eléctricos (RIBIERSE-CYTED), funded through the 2022 call for thematic networks of the CYTED (Ibero-American Program of Science and Technology for Development).

Data Availability Statement: The original contributions presented in this study are included in the article. Further inquiries may be directed to the corresponding author.

Acknowledgments: The authors acknowledge the use of AI-based tools, including OpenAI’s ChatGPT, version 5.1, which supported the refinement of the manuscript’s language, structure, and overall clarity. These tools were utilized exclusively to improve the presentation and readability of the original ideas, formulations, and numerical simulations provided by the authors. Importantly, the AI tools did not contribute to the development of the scientific content, to the formulation, or to the validation of the results, for which the authors assume full responsibility.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wang, W.; Li, G.; Guo, J. Large-Scale Renewable Energy Transmission by HVDC: Challenges and Proposals. *Engineering* **2022**, *19*, 252–267. <https://doi.org/10.1016/j.eng.2022.04.017>.
2. Cavus, M. Advancing Power Systems with Renewable Energy and Intelligent Technologies: A Comprehensive Review on Grid Transformation and Integration. *Electronics* **2025**, *14*, 1159. <https://doi.org/10.3390/electronics14061159>.
3. Pietribiasi, D.; Yaman, B.; Wu, J. A review of HVDC cable systems configurations for long distance submarine power transmission. In *Proceedings of the 2025 25th Conference of the Electric Power Supply Industry (CEPSI)*; IEEE: New York, NY, USA, 2025; pp. 356–361. <https://doi.org/10.1109/cepsi66359.2025.11403663>.
4. Zhou, Y.; Liu, Y.; Huang, S.; Guo, C.; Cao, Y.; Li, Y. A review of HVDC-based transmission congestion alleviation strategies for modern power systems. *Electr. Power Syst. Res.* **2026**, *255*, 112748. <https://doi.org/10.1016/j.epsr.2026.112748>.
5. Meng, Q.; He, Y.; Hussain, S.; Lu, J.; Guerrero, J.M. Design and optimization of solar photovoltaic microgrids with adaptive storage control for residential standalone systems. *Energy Rep.* **2025**, *14*, 989–997. <https://doi.org/10.1016/j.egyrs.2025.06.048>.
6. Starke, M.; Tolbert, L.M.; Ozpineci, B. AC vs. DC distribution: A loss comparison. In *Proceedings of the 2008 IEEE/PES Transmission and Distribution Conference and Exposition*; IEEE: New York, NY, USA, 2008; pp. 1–7. <https://doi.org/10.1109/tdc.2008.4517256>.
7. Montoya, O.D.; Gil-González, W.; Garces, A. Numerical methods for power flow analysis in DC networks: State of the art, methods and challenges. *Int. J. Electr. Power Energy Syst.* **2020**, *123*, 106299. <https://doi.org/10.1016/j.jepes.2020.106299>.
8. Han, J.; Oh, Y.S.; Gwon, G.H.; Kim, D.U.; Noh, C.H.; Jung, T.H.; Lee, S.J.; Kim, C.H. Modeling and Analysis of a Low-Voltage DC Distribution System. *Resources* **2015**, *4*, 713–735. <https://doi.org/10.3390/resources4030713>.
9. Singh, S.; Gautam, A.R.; Fulwani, D. Constant power loads and their effects in DC distributed power systems: A review. *Renew. Sustain. Energy Rev.* **2017**, *72*, 407–421. <https://doi.org/10.1016/j.rser.2017.01.027>.
10. Schipper, J.; Bathini, V.; Mukhedkar, R.; Watson, N.; Pirnia, M.; Nair, N.K.; Watson, J. Steady-state power system analysis revisited for hybrid AC–DC grids. *Renew. Sustain. Energy Rev.* **2025**, *215*, 115564. <https://doi.org/10.1016/j.rser.2025.115564>.
11. Liu, Z.; Su, M.; Sun, Y.; Yuan, W.; Han, H.; Feng, J. Existence and Stability of Equilibrium of DC Microgrid With Constant Power Loads. *IEEE Trans. Power Syst.* **2018**, *33*, 6999–7010. <https://doi.org/10.1109/tpwrs.2018.2849974>.
12. Simpson-Porco, J.W.; Dorfler, F.; Bullo, F. On Resistive Networks of Constant-Power Devices. *IEEE Trans. Circuits Syst. II Express Briefs* **2015**, *62*, 811–815. <https://doi.org/10.1109/tcsii.2015.2433537>.
13. Petković, M.S.; Neta, B.; Petković, L.D.; Džunić, J. Multipoint methods for solving nonlinear equations: A survey. *Appl. Math. Comput.* **2014**, *226*, 635–660. <https://doi.org/10.1016/j.amc.2013.10.072>.
14. Bhuyan, S.; Hazarika, S.; Bardalai, A. Power flow analysis on IEEE 57 bus system using MATLAB. *Int. J. Eng. Res. Technol. (IJERT)* **2014**, *3*, 1161–1171.
15. Afolabi, O.A.; Ali, W.H.; Cofie, P.; Fuller, J.; Obiomon, P.; Kolawole, E.S. Analysis of the Load Flow Problem in Power System Planning Studies. *Energy Power Eng.* **2015**, *7*, 509–523. <https://doi.org/10.4236/epe.2015.710048>.
16. Taheri, S.; Kekatos, V. Power Flow Solvers for Direct Current Networks. *IEEE Trans. Smart Grid* **2020**, *11*, 634–643. <https://doi.org/10.1109/tsg.2019.2927455>.

17. Abdullah, S.; Choubey, N.; Dara, S. An efficient two-point iterative method with memory for solving non-linear equations and its dynamics. *J. Appl. Math. Comput.* **2023**, *70*, 285–315. <https://doi.org/10.1007/s12190-023-01953-w>.
18. Hernández, M.A.; Salanova, M.A. A family of chebyshev-halley type methods. *Int. J. Comput. Math.* **1993**, *47*, 59–63. <https://doi.org/10.1080/00207169308804162>.
19. Nedzhibov, G.H. A family of multi-point iterative methods for solving systems of nonlinear equations. *J. Comput. Appl. Math.* **2008**, *222*, 244–250. <https://doi.org/10.1016/j.cam.2007.10.054>.
20. Zafar, F.; Cordero, A.; Maryam, H.; Torregrosa, J.R. A Novel Higher-Order Numerical Scheme for System of Nonlinear Load Flow Equations. *Algorithms* **2024**, *17*, 86. <https://doi.org/10.3390/a17020086>.
21. Montoya, O.D.; Gil-González, W.; Garcés, A. A successive approximations method for power flow analysis in bipolar DC networks with asymmetric constant power terminals. *Electr. Power Syst. Res.* **2022**, *211*, 108264. <https://doi.org/10.1016/j.epsr.2022.108264>.
22. Grisales-Noreña, L.F.; Montoya, O.D.; Gil-González, W.J.; Perea-Moreno, A.J.; Perea-Moreno, M.A. A Comparative Study on Power Flow Methods for Direct-Current Networks Considering Processing Time and Numerical Convergence Errors. *Electronics* **2020**, *9*, 2062. <https://doi.org/10.3390/electronics9122062>.
23. Augugliaro, A.; Dusonchet, L.; Ippolito, M.; Sanseverino, E. An efficient iterative method for load-flow solution in radial distribution networks. In *Proceedings of the 2001 IEEE Porto Power Tech Proceedings (Cat. No.01EX502)*; IEEE: New York, NY, USA, 2001; Volume 3, p. 6. <https://doi.org/10.1109/ptc.2001.964878>.
24. Montoya, O.D.; Gil-González, W.; Orozco-Henao, C. On the convergence of the power flow methods for DC networks with mesh and radial structures. *Electr. Power Syst. Res.* **2021**, *191*, 106881. <https://doi.org/10.1016/j.epsr.2020.106881>.
25. Grisales-Noreña, L.; Morales-Duran, J.; Velez-Garcia, S.; Montoya, O.D.; Gil-González, W. Power flow methods used in AC distribution networks: An analysis of convergence and processing times in radial and meshed grid configurations. *Results Eng.* **2023**, *17*, 100915. <https://doi.org/10.1016/j.rineng.2023.100915>.
26. Montoya, O.D.; Pulgarín Rivera, J.D.; Grisales-Noreña, L.F.; Gil-González, W.; Andrade-Rengifo, F. A Family of Newton and Quasi-Newton Methods for Power Flow Analysis in Bipolar Direct Current Networks with Constant Power Loads. *Math. Comput. Appl.* **2025**, *30*, 50. <https://doi.org/10.3390/mca30030050>.
27. Tostado-Véliz, M.; Kamel, S.; Jurado, F.; Ruiz-Rodriguez, F.J. On the Applicability of Two Families of Cubic Techniques for Power Flow Analysis. *Energies* **2021**, *14*, 4108. <https://doi.org/10.3390/en14144108>.
28. Gutiérrez, J.; Hernández, M. A family of Chebyshev-Halley type methods in Banach spaces. *Bull. Aust. Math. Soc.* **1997**, *55*, 113–130. <https://doi.org/10.1017/s0004972700030586>.
29. Gutiérrez, J.M.; Galilea, V. Two Dynamic Remarks on the Chebyshev–Halley Family of Iterative Methods for Solving Nonlinear Equations. *Axioms* **2023**, *12*, 1114. <https://doi.org/10.3390/axioms12121114>.
30. Behl, R.; Martínez, E.; Cevallos, F.; Alarcón, D. A Higher Order Chebyshev-Halley-Type Family of Iterative Methods for Multiple Roots. *Mathematics* **2019**, *7*, 339. <https://doi.org/10.3390/math7040339>.
31. Villafuerte, R.; Villafuerte, R.A.; Medina, J.; Mejía, E. Aplicación de un Método Iterativo de dos Pasos para el Cálculo de Flujos de Potencia. *Inf. Tecnológica* **2017**, *28*, 189–198. <https://doi.org/10.4067/s0718-07642017000100019>.

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.