Two efficient and reliable Power-Flow methods with Seventh order of convergence

Marcos Tostado-Véliz, Salah Kamel, and Francisco Jurado, *Senior Member, IEEE*

Abstract*— *Developing efficient Power-Flow solution (PF) techniques competitive with the conventional Newton-Raphson (NR), will be necessary in the future power system paradigm. In this paper, two seventh order PF solution methods based on two classic cubic nonlinear solvers are developed. The developed methods are more efficient than the standard NR. Up to ten realistic distribution and transmission systems are considered for validating the developed methods. Three scenarios including base load, maximum load level and enforcement of generators' reactive limits are considered. Results show that the developed PF solution methods enable computational savings higher than 20% and 30% with respect to standard NR and the cubic methods, respectively.

***Index Terms*— Power-Flow analysis, High order methods, Computational efficiency, Cubic nonlinear solvers, Realistic power systems.**

I. INTRODUCTION

A. *Motivation*

PF is widely considered as the most important computational tool in Power System Analysis [1]. It is frequently used for planning, operation and control of power systems [2] along as main ingredient of optimization problems and security tools [3], or as initial point for dynamic simulations. Basically, PF is a nonlinear problem which relates the bus voltages to the power generation and consumption. As main result of PF, steady state bus voltages are obtained, nevertheless, other related results as total power losses can be also calculated.

As nonlinear problem, PF has to be solved using some available iterative method. As it is well known, NR is frequently considered as the standard nonlinear method, and it is also typically used in industry tools.

Recently, PF has attracted a huge attention especially for solving realistic large-scale networks (see e.g. [4]). This kind of systems will be presumably relevant in a future power system paradigm [5]. Consequently, it is especially important developing efficient and reliable PF methods, which are competitive with the most conventional methodology, i.e. NR. This paper aims at providing a step in this direction.

B. *Literature Review*

Other PF solution methodologies like decoupled techniques [6] or inexact and dishonest Newton-like methods [7] have been explored for alleviating the computational burden of NR. Although quite efficient, these techniques present some important drawbacks, especially in stiff problems (heavy load or high R/X ratio). In such cases, the mentioned methods normally require a huge amount of iterations for converging. It is worth mentioning the limitations of the decoupled methods for incorporating alternative PF formulations such as [8]-[10].

Although the PF problem is strongly nonlinear, several linear formulations have been studied with the aim of obtaining direct solutions [11]. However, this kind of approaches are limited to some online tools, due to the accuracy of the results may be compromised.

A plethora of robust approaches based on the optimal-multiplier paradigm [9], have been studied during decades for solving ill-conditioned cases. This kind of approaches aims to modify the Newton's increment vector by solving an optimization problem raised from the second order expansion of the Taylor series. As result, the unknowns evolve on the direction of reducing the residuals. These methods are normally not competitive with NR due to the convergence rate is eminently lower. In addition, this kind of techniques may be trapped in a local minimum, where the optimal multiplier slowly tends to zero and the algorithm does not evolve. More recently, several PF techniques have been studied for solution of realistic large-scale ill-conditioned systems. One can be referred to the works conducted by Milano [13], [14], Pourbagher and Derakhshandeh [15] or the authors [4], [16], by just an example. The main issue of these techniques is their lower convergence rate. Most of them achieve higher robustness properties by intentionally deteriorating the convergence properties of the Newton's method [4], [13], [14], [16]. Thus, the Newton's increment vector is calculated in a safer way, however more iterations are normally required. In addition, some robust solvers present a high computational burden. For example, the solvers in [4] and [13], require several factorizations each iteration, while the methods in [14] and [16] have to compute various function evaluations. Regarding the Regularization methods [15], it usually requires an extra matrix-matrix product, which is very heavy especially in large-

M. Tostado-Véliz and F. Jurado are with Department of Electrical Engineering, University of Jaén, 23700 EPS Linares, Jaén, Spain (e-mail: mtostado@ujaen.es and fjurado@ujaen.es). S. Kamel is with the Department of Electrical Engineering, Faculty of Engineering, Aswan University, 81542 Aswan, Egypt (email: skamel@aswu.edu.eg).

scale systems. In addition, this calculation leads to factorize a dense matrix, complexing the factorization. These issues have limited the application of these methodologies for just a few idiosyncratic cases.

C. *Contributions & Paper Organization*

In PF analysis, the factorization of Jacobian matrix is by far the heaviest calculation [14]. Quadratic NR can be consequently considered an efficient technique, since, it only involves a LU decomposition along with other cheap computations each iteration.

On the other hand, a huge variety of the so-called high order Newton-like methods have been also proposed for solving scalar nonlinear functions or systems of nonlinear equations (interested readers can find recently surveys about this topic in [17], [18] and references therein). The High order Newton-like methods are based on the Jacobian matrix or the fist Fréchet derivative calculation of the function evaluated, however, unlike NR, they present higher order convergence (i.e. higher than two). Due to this feature, this kind of methodologies usually converge faster than NR. It is worth mentioning that, unlike other PF solvers, alternative formulations or variables, such as [8]-[10], could be easily adapted for any High order Newton-like method.

Despite these interesting features, the High order Newton-like methods have not been profusely studied for PF analysis yet due to, generally, they cannot be considered efficient as more than a LU decomposition is involved each iteration. Hence, higher convergence features are not normally enough for offset the overall computational cost and, typically, NR is still preferred. This is the main reason why some high order approaches such as [19] or [20] are, a priori, few attractive for industry tools. This paper aims to address this issue by proposing two modifications of the methods in [19], [20] so that the order of convergence is increased to seven. Thus, although our proposals still require two LU factorizations each iteration, they achieve important computational savings due to their outstanding convergence features.

Several realistic radial distribution and transmission systems are considered for validating the developed methods. In addition, several scenarios enabling the generator's reactive limits and increasing the loading level close to the collapse point are also covered.

Remainder of this paper is organized as follows. Section II outlines the PF problem and briefly describes its solution using the standard NR. Third order Newton-like methods [15, 16] are briefly described in Section III. Section IV introduces the developed PF methods with seventh order of convergence and its superior efficiency is also checked. Developed PF methods are tested and compared in several realistic systems in Section V. Finally, Section VI concludes the paper.

II. BACKGROUND

The PF in polar coordinates can be established as a set of n nonlinear equations as follows [21]:

$$\mathbf{g}(\boldsymbol{x})\begin{cases}\mathbf{g}_{\boldsymbol{P}}, & \forall i \in [\mathcal{N}_l, \mathcal{N}_g] \\ \mathbf{g}_{\boldsymbol{Q}}, & \forall i \in \mathcal{N}_l\end{cases} \tag{1}$$

$$g_{P_i} = P_i^{sch} - \sum |V_i|\,|V_j||Y_{ij}|\cos(\theta_{ij} - \delta_i + \delta_j) \tag{2}$$

$$g_{Q_i} = Q_i^{sch} - \sum |V_i|\,|V_j||Y_{ij}|\sin(\theta_{ij} - \delta_i + \delta_j) \tag{3}$$

$$\boldsymbol{x} = [\boldsymbol{\delta_g}, \boldsymbol{\delta_l}, \boldsymbol{V_g}]^T \tag{4}$$

where, $\mathcal{N}_l$ and $\mathcal{N}_g$ are the sets of PQ and PV buses, respectively. $P_i^{sch} \in \mathbb{R}$ and $Q_i^{sch} \in \mathbb{R}$ are the active and reactive power injected at i^{th} bus, respectively. $V_i\angle\delta_i \in \mathbb{C}$ is the complex voltage at i^{th} bus. $Y_{ij}\angle\theta_{ij} \in \mathbb{C}$ is the ij^{th} element of the admittance matrix, $\boldsymbol{\delta_g} \in \mathbb{R}^{n_g}$ is the vector of voltage angles at PV buses, $\boldsymbol{\delta_l} \in \mathbb{R}^{n_l}$ is the vector of voltage angles at PQ buses and $\boldsymbol{V_l} \in \mathbb{R}^{n_l}$ is the vector of voltage angles at PQ buses, $n_l \in \mathbb{N}$ and $n_g \in \mathbb{N}$ are the total number of PQ buses and PV buses, respectively.

As commented before, NR is considered as the most standard method for solving PF problem. For a generic k^{th} iteration, NR for solving the set of equations (1) proceeds as follows:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - [\mathbf{g}'(\boldsymbol{x}^{(k)})]^{-1}\mathbf{g}(\boldsymbol{x}^{(k)}) \tag{5}$$

where, $\mathbf{g}'(\boldsymbol{x}) \in \mathbb{R}^{n\times n}$ is the Jacobian matrix formed by the first derivatives of (1) with respect to (4).

III. CLASSIC CUBIC METHODS FOR PF ANALYSIS

During this section, application of two classic third order Newton-like methods [19], [20], is briefly described.

A. *Weerakoon's Third Order Method*

In [19], Weerakoon and Fernando developed a third order Newton-like method, derived from the approximation of the integral arisen from the Newton's theorem by the Trapezoidal rule. For a generic k^{th} iteration, this method solves the set of equations (1) as follows:

$$\begin{cases}\boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} - [\mathbf{g}'(\boldsymbol{x}^{(k)})]^{-1}\mathbf{g}(\boldsymbol{x}^{(k)}) \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - 2[\mathbf{g}'(\boldsymbol{x}^{(k)}) + \mathbf{g}'(\boldsymbol{y}^{(k)})]^{-1}\mathbf{g}(\boldsymbol{x}^{(k)})\end{cases} \tag{6}$$

The method (6) requires two Jacobian evaluations, two LU decompositions and a function evaluation each iteration.

B. *Özban's Third Order Method*

In [20], Özban proposed a modification of (6) which proceeds for a generic k^{th} iteration, as follows:

$$\begin{cases}\boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} - [\mathbf{g}'(\boldsymbol{x}^{(k)})]^{-1}\mathbf{g}(\boldsymbol{x}^{(k)}) \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \left[\mathbf{g}'\left(\frac{\boldsymbol{x}^{(k)}+\boldsymbol{y}^{(k)}}{2}\right)\right]^{-1}\mathbf{g}(\boldsymbol{x}^{(k)})\end{cases} \tag{7}$$

The method (7) requires two Jacobian evaluations, two LU decompositions and a function evaluation each iteration. However, it is slightly more efficient than (6) since one matrix sum is avoided and replaced by a vector sum.

IV. DEVELOPED PF METHODS

A. *Conceptual Idea of the Developed Methods*

As commented, the mappings (6) and (7) are not competitive with NR since their third order of convergence is not able to offset the computational burden brought by the two LU decompositions required each iteration (see Section IV.C). One way to make the studied cubic techniques competitive, could be increasing their order of convergence to be high enough for compensating their higher computational burden. Basically, there are two ways for increasing the order of convergence of a nonlinear solver. On the one hand, one can evaluate the Jacobian matrix in several points. On the other hand, rather than the Jacobian, one can evaluate the function in several points [22]. It is worth noting that the former approach is used for the mappings (6) and (7). In Section IV.C, it is demonstrated that these techniques are not competitive with NR since increasing the order of convergence by evaluating the Jacobian in several points bring more LU factorizations, specifically, for a third order method, 2 LU factorizations are required, thus, the total number of these calculations would increase with the order of the method. Instead, one can recur to evaluate the function in several points, which supposes a cheaper computation (a LU factorization has a computational cost of $\mathcal{O}(n^3)$ while a function evaluation supposes a cost of $\mathcal{O}(n)$). This is the approach used for developing our proposals as follows:

- It is worth noting that, in the mappings (6) and (7), the point $\boldsymbol{y}$ is calculated using NR. Instead, we propose to use the cubic method [22]. Thus, $\mathbf{g}$ is evaluated twice at $\boldsymbol{x}$ and the point $\boldsymbol{y}$ brings third order of convergence.
- In the mappings (6) and (7), the unknowns vector is updated using the Jacobian information at $\boldsymbol{x}$ and $\boldsymbol{y}$. Alternatively, we propose to take advantage from this information to increase the order of convergence. Thus, an extra step is added which calculates an intermediate point namely $\boldsymbol{w}$. Evaluating the function at the new point and the information of the Jacobian at $\boldsymbol{x}$ and $\boldsymbol{y}$, the unknowns vector is updated.

By these modifications, no extra factorizations are required, however, since seventh order of convergence is achieved (see Appendixes A and B), the developed methods are more competitive than NR (see Section IV.C).

B. Developed 7th Order Methods

The main contribution of this paper is developing two efficient modifications of iterative methods (6) and (7) with seventh order of convergence. Thus, for a generic k^{th} iteration, the developed methods solve the set of equations (1) as follows:

$$\begin{cases} \boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} - \left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{x}^{(k)}\right) \\ \boldsymbol{z}^{(k)} = \boldsymbol{y}^{(k)} - \left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{y}^{(k)}\right) \\ \boldsymbol{w}^{(k)} = \boldsymbol{y}^{(k)} - 2\left[\mathbf{g}'\left(\boldsymbol{y}^{(k)}\right) + \mathbf{g}'\left(\boldsymbol{z}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{y}^{(k)}\right) \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{w}^{(k)} - 2\left[\mathbf{g}'\left(\boldsymbol{y}^{(k)}\right) + \mathbf{g}'\left(\boldsymbol{z}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{w}^{(k)}\right) \end{cases} \tag{8}$$

$$\begin{cases} \boldsymbol{y}^{(k)} = \boldsymbol{x}^{(k)} - \left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{x}^{(k)}\right) \\ \boldsymbol{z}^{(k)} = \boldsymbol{y}^{(k)} - \left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{y}^{(k)}\right) \\ \boldsymbol{w}^{(k)} = \boldsymbol{y}^{(k)} - \left[\mathbf{g}'\left(\frac{\boldsymbol{y}^{(k)}+\boldsymbol{z}^{(k)}}{2}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{y}^{(k)}\right) \\ \boldsymbol{x}^{(k+1)} = \boldsymbol{w}^{(k)} - \left[\mathbf{g}'\left(\frac{\boldsymbol{y}^{(k)}+\boldsymbol{z}^{(k)}}{2}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{w}^{(k)}\right) \end{cases} \tag{9}$$

The developed methods require two LU decompositions and three functions evaluations. Nevertheless, differences between (8) and (9) are more notable since (8) requires three Jacobian evaluations, while (9) just requires two. On the other hand, the proof of seventh order of convergence of (8) and (9) are included in Appendix A and B, respectively. This Section is concluded by summarizing the main steps of the developed methods in Algorithm 1 using pseudocode. In this paper, $\|\mathbf{g}\|_\infty$ has been considered as convergence criterion and $\varepsilon \in \mathbb{R}^+$ the convergence tolerance.

Algorithm 1: main steps of the developed methods

```
Let $\boldsymbol{x}^{(0)}$ and $\varepsilon$
Initialize $k \leftarrow 0$
while $\|\mathbf{g}(\boldsymbol{x}^{(k)})\|_\infty > \varepsilon$ do
    $\boldsymbol{x}^{(k+1)} \leftarrow$ Solve (8) # or (9)
    $k \leftarrow k + 1$
    if $\|\mathbf{g}(\boldsymbol{x}^{(k)})\|_\infty \leq \varepsilon$ then
        break # Convergence
    end if
end do
return solution $\boldsymbol{x}^{(k)}$
```

C. Comparison with the Standard NR and the Cubic Methods (6) and (7)

In order to compare the developed methods with the standard NR and the classic cubic methods (6) and (7), let us consider the following efficiency index [23].

$$FEI = p^{1/CO} \tag{10}$$

where, $p \in \mathbb{R}^+$ is the order of convergence and CO is the total cost of an iteration, which can be estimated as follows:

$$CO = O(\mathbf{g}) + O(\mathbf{g}') + O(n, q) \tag{11}$$

where, $O(\mathbf{g})$, $O(\mathbf{g}')$ and $O(n, q)$ represent the total computational cost of a function evaluation, a Jacobian evaluation and the solution of q linear systems using the same LU decomposition, respectively. In this regard, let us introduce the following theorem for estimating the cost of $O(n, q)$ [24].

Theorem 1. *The total computational cost of solving q linear systems of n equations with the same LU decomposition, is:*

$$O(n, q) = \frac{1}{3}n^3 + qn^2 - \frac{1}{3}n \tag{12}$$

Making use of the Theorem 1, the following theorems estimate the value of (10) for the developed methods.

Theorem 2. *For the developed iterative method (8), we have: $FEI = 7^{1/\frac{2}{3}n^3+7n^2+\frac{7}{3}n}$*

Proof. Iterative algorithm (8) requires three functions evaluations, three Jacobians evaluations along with the solution of four linear systems using two LU decompositions, thus we have $CO = 3n + 3n^2 + 2\left(\frac{1}{3}n^3 + 2n^2 - \frac{1}{3}n\right) = \frac{2}{3}n^3 + 7n^2 + \frac{7}{3}n$ □

Theorem 3. *For the developed iterative method (9), we have:* $FEI = 7^{1/\frac{2}{3}n^3+6n^2+\frac{7}{3}n}$

Proof. Iterative algorithm (9) requires three functions evaluations, two Jacobians evaluations along with the solution of four linear systems using two LU decompositions, thus we have $CO = 3n + 2n^2 + 2\left(\frac{1}{3}n^3 + 2n^2 - \frac{1}{3}n\right) = \frac{2}{3}n^3 + 6n^2 + \frac{7}{3}n$ □

Fig. 1 plots the value of efficiency index (10) for the standard NR and methods (6)-(9). As it can be seen, while cubic methods (6) and (7) are less efficient than the NR, the developed seventh order methods are clearly the more efficient. It can be also noted that (9) is slightly more efficient than (8) as a Jacobian evaluation is avoided.

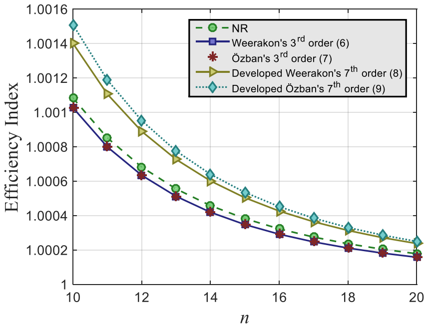

Fig 1. Efficiency index of different iterative methods

D. *Multiple Solutions*

The PF equations have two solutions typically called high (stable) and low (unstable) voltage solutions. The developed mappings as NR and the cubic methods (6) and (7) may converge either to the high or low voltage values. Reachability of each solution depends on the selected initial guess $\boldsymbol{x}^{(0)}$. Thus, one can define two convergence regions as the sets of starting points which the developed techniques converge either to the high or low voltage solution. To illustrate that, let us refer to the well-known P-V curve of the i^{th} bus of a generic system as in Fig. 2. In this figure, the high and low voltage solutions (V_i^{high} and V_i^{low}, respectively) are represented for a specified loading level (λ^{act}). As observed, two convergence regions exist in the vicinity of both solutions. If the selected starting guess lies within the convergence region of the high voltage value, our methods would converge to V_i^{high}, while they would converge to V_i^{low} if $\boldsymbol{x}^{(0)}$ lies within the convergence region of the low voltage solution. However, only one solution at a time can be calculated using the developed methods. Thus, depending on $\boldsymbol{x}^{(0)}$, the mappings (8) and (9) (as NR and the mappings (6) and (7)), will converge to either the high or low voltage solution. Nevertheless, one can recur to further PF analysis tools like the Continuation Power Flow [25], for calculating all the solutions of the system. The developed mappings can be straightforward integrated within the Continuation Power Flow at corrector stage, thus, they can be used for simultaneously calculating the high and low voltage solutions.

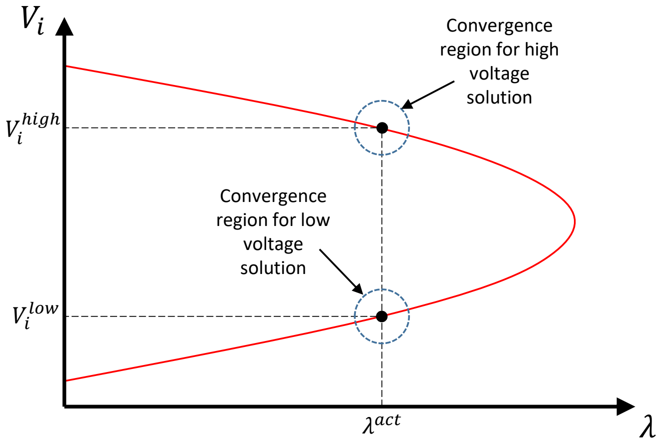

Fig. 2 This figure represents the P-V curve of a generic system with the high and low voltage solutions for a specified loading level, along the convergence regions associated with each solution

V. NUMERICAL RESULTS

A. *Studied Systems*

Several realistic distribution and transmission systems are available in the Matpower's database [26]-[30]. Table I collects the studied systems along with their identifying number. For the sake of completeness, their main characteristics are also included.

TABLE I
STUDIED SYSTEMS

Syst. No.	Name	Buses	Branches	Generators	Load MW	Load MVar	n
#1	*case69*†	69	68	1	3.8	2.7	136
#2	*case141*†	141	140	1	11.9	7.4	280
#3	*case1354pegase*	1354	1991	260	73059.7	13401.4	2447
#4	*case_ACTIVSg2000*	2000	3206	544	67109.2	19014.3	3607
#5	*case2737sop*	2737	3506	399	11267.2	3953.2	5280
#6	*case2746wop*	2746	3514	514	18962.1	5534.5	5141
#7	*case2746wp*	2746	3514	520	24873	7146.5	5127
#8	*case2869pegase*	2869	4582	510	132437.3	29007.8	5227
#9	*case3120sp*	3120	3693	505	21181.5	8723.2	5991
#10	*case9241pegase*	9241	16049	1445	312354.1	73581.6	17036

† Radial distribution system

B. Tested Scenarios

Along with the base load scenario (Scenario #1), the systems collected in Table I have been also tested in the following situations:

- Scenario #2 Limit Load: this scenario considers a loading level near to the collapse point. This situation is demanding for PF solution techniques since typically the degree of ill conditioning is high and, consequently, slow convergence behaviour or even divergence may be experimented. Limit versions of the studied systems can be downloaded from [31].
- Scenario #3 Considering generators' reactive limits: in this scenario, the default strategy implemented in Matpower has been considered for handling the generators' reactive limits. In this strategy, the PV buses connected with the violated generators, are switched to PQ buses with injected reactive powers equal to the limits of these generators (see Matpower's manual for details [32]). Only one modification has been included. In Matpower's default code, a flat start is considered for starting each PF solution, in this paper, the last found solution has been used for initializing the following PF calculation. Due to this starting point is normally closer to the solution than the flat start, better and reliable results are normally achieved. It is worth noting that, since the radial distribution systems do not involve any PV bus, solving the PF in this scenario essentially yields the same results obtained for the Scenario #1. Therefore, these systems have not been included in this scenario.

In all simulations, a flat start has been considered and $\varepsilon = 10^{-6}$ has been imposed as convergence criterion. On the other hand, all simulations have been done under Windows 10 on a 3.4 GHz Intel Core i7-8750H CPU 2.2 GHz personal laptop (16.00 GB RAM). It is worth mentioning that the reported results have been obtained as the mean value of 5000 simulations, in order to avoid the influence of other computational activities.

C. Tested PF Methods

Along standard NR in polar coordinates, the Weerakoon's cubic method (3OW), the Özban's cubic technique (3OÖ), the developed methods (8) and (9) (7OW and 7OÖ, respectively), and the Implicit Continuous Newton's paradigm using the Backward Euler method (BEM) [14] have been tested. In addition, the Forward-Backward Sweep method (FBS) [33] has been considered for solving the radial distribution systems. For FBS, an iteration is considered when the algorithm walks the whole network in backward and forward directions.

D. Radial Distribution Systems

Tables II and III present the total number of iterations and LU factorizations computed by the tested PF methods in the two considered radial distribution systems, for the three studied Scenarios. It is worth noting that, in the case of the radial distribution systems, the Scenarios #1 and #3 are analogue. As observed, the lower iteration counter is achieved at the higher convergence rate. Thus, the developed methods required less iterations than the other techniques. Superiority of the developed solvers is more notorious for the Scenario #2. Here, those methods with linear convergence (BEM and FBS), required many iterations for converging. NR needed 9 and 10 iterations for solving the Systems #1 and #2, respectively, while the cubic techniques required 6 iterations for solving both systems. On the other hand, our methods just needed 4 iterations. In all cases, 7OW and 7OÖ computed less LU factorizations than NR, BEM, 3OW and 3OÖ. FBS does not need factorizations since it is carried out by computing multiple scalar operations.

TABLE II
TOTAL ITERATIONS AND LU DECOMPOSITIONS IN RADIAL SYSTEMS FOR THE SCENARIOS #1 AND #3

	System #1		System #2	
	# Iterations	# LU Fact.	# Iterations	# LU Fact.
NR	3	3	3	3
BEM	10	10	9	9
FBS	4	0	3	0
3OW	2	4	2	4
3OÖ	2	4	2	4
7OW	1	2	1	2
7OÖ	1	2	1	2

TABLE III
TOTAL ITERATIONS AND LU DECOMPOSITIONS IN RADIAL SYSTEMS FOR THE SCENARIO #2

	System #1		System #2	
	# Iterations	# LU Fact.	# Iterations	# LU Fact.
NR	9	9	10	10
BEM	13	13	14	14
FBS	269	0	336	0
3OW	6	12	6	12
3OÖ	6	12	6	12
7OW	4	8	4	8
7OÖ	4	8	4	8

If one considers the factorization of the Jacobian matrix as the heaviest part of a PF calculation, the developed techniques should be more efficient than the other tested techniques (see Section IV.C). This is confirmed by observing the solution times reported in Tables IV and V, where one can observe that the computational burden of a PF solver is quite proportional to the total amount of LU factorizations required. In the case of FBS, this technique does not involve LU factorizations, however, this kind of algorithms are generally less competitive due to it computes multiple loops, which are less efficient than vector and matrix calculations [34].

TABLE IV
SOLUTION TIMES [ms] IN RADIAL SYSTEMS FOR THE SCENARIOS #1 AND #3

	System #1	System #2
NR	3.6	4.5
BEM	17.6	18.0
FBS	14.0	15.1
3OW	4.0	5.3
3OÖ	4.0	5.2
7OW	3.5	4.2
7OÖ	3.1	3.8

TABLE V
SOLUTION TIMES [ms] IN RADIAL SYSTEMS FOR THE SCENARIO #2

	System #1	System #2
NR	8.0	11.7
BEM	18.5	19.1
FBS	57.3	118.1
3OW	9.1	13.2
3OÖ	9.0	12.8
7OW	7.6	11.2
7OÖ	7.4	10.4

E. Large Transmission Systems

Tables VI-VIII report the total iterations in the Systems #3-10 for the Scenarios #1-3, respectively. For the Scenario #1, 7OW and 7OÖ are able to save various iterations in comparison with NR and BEM. In comparison with the tested cubic techniques, our methods were able to save a fewer iteration. The developed 7OW and 7OÖ were even more competitive for the Scenarios #2 and #3. In the Scenario #2, the developed methods were able to save up to 8, 11 and 3 iterations in comparison with NR, BEM and the cubic methods, respectively. In the Scenario #3, 7OW and 7OÖ employed up to 9, 47 and 4 fewer iterations than NR, BEM and the third order methods, respectively. These results allow to the developed techniques to save a considerable number of LU factorizations, as observed in Tables IX-XI, were the total LU decompositions for the Scenarios considered in Tables VI-VIII, respectively, are presented. Here, one can observe that our solvers required less LU decompositions than the other tested techniques.

TABLE VI
TOTAL ITERATIONS IN TRANSMISSION SYSTEMS FOR THE SCENARIO #1

	System #							
	3	4	5	6	7	8	9	10
NR	5	5	5	6	6	5	5	6
BEM	13	12	13	13	13	13	13	13
3OW	3	3	3	3	3	3	3	3
3OÖ	3	3	3	3	3	3	3	3
7OW	2	2	2	2	2	2	2	2
7OÖ	2	2	2	2	2	2	2	2

TABLE VII
TOTAL ITERATIONS IN TRANSMISSION SYSTEMS FOR THE SCENARIO #2

	System #							
	3	4	5	6	7	8	9	10
NR	11	9	11	11	12	9	12	10
BEM	15	14	15	15	15	14	15	15
3OW	7	6	7	7	7	6	7	6
3OÖ	7	6	7	7	7	6	7	6
7OW	4	4	4	4	4	4	4	4
7OÖ	4	4	4	4	4	4	4	4

TABLE VIII
TOTAL ITERATIONS IN TRANSMISSION SYSTEMS FOR THE SCENARIO #3

	System #							
	3	4	5	6	7	8	9	10
NR	10	13	11	16	15	12	13	13
BEM	34	42	38	53	50	43	53	45
3OW	7	8	6	10	9	8	10	8
3OÖ	7	8	6	10	9	8	10	8
7OW	4	5	5	7	6	5	6	5
7OÖ	4	5	5	7	6	5	6	5

TABLE IX
TOTAL LU FACTORIZATIONS IN TRANSMISSION SYSTEMS FOR THE SCENARIO #1

	System #							
	3	4	5	6	7	8	9	10
NR	5	5	5	6	6	5	5	6
BEM	13	12	13	13	13	13	13	13
3OW	6	6	6	6	6	6	6	6
3OÖ	6	6	6	6	6	6	6	6
7OW	4	4	4	4	4	4	4	4
7OÖ	4	4	4	4	4	4	4	4

TABLE X
TOTAL LU FACTORIZATIONS IN TRANSMISSION SYSTEMS FOR THE SCENARIO #2

	System #							
	3	4	5	6	7	8	9	10
NR	11	9	11	11	12	9	12	10
BEM	15	14	15	15	15	14	15	15
3OW	14	12	14	14	14	12	14	12
3OÖ	14	12	14	14	14	12	14	12
7OW	8	8	8	8	8	8	8	8
7OÖ	8	8	8	8	8	8	8	8

TABLE XI
TOTAL LU FACTORIZATIONS IN TRANSMISSION SYSTEMS FOR THE SCENARIO #3

	System #							
	3	4	5	6	7	8	9	10
NR	10	13	11	16	15	12	13	13
BEM	34	42	38	53	50	43	53	45
3OW	14	16	12	20	18	16	20	16
3OÖ	14	16	12	20	18	16	20	16
7OW	8	10	10	14	12	10	12	10
7OÖ	8	10	10	14	12	10	12	10

Tables XII-XIV provide the solution times for the Scenarios #1-3 in the studied transmission systems. Due to the lower amount of LU factorizations, our techniques were more efficient than the other tested solvers.

TABLE XII
SOLUTION TIMES [ms] IN TRANSMISSION SYSTEMS FOR THE SCENARIO #1

	System #							
	3	4	5	6	7	8	9	10
NR	43.2	84.5	92.8	108.0	98.7	99.2	100.6	400.2
BEM	113.1	219.4	251.1	242.4	229.9	269.2	277.0	943.2
3OW	48.8	100.6	103.4	101.7	95.3	112.7	115.1	393.4
3OÖ	48.5	99.4	103.0	101.2	95.1	111.0	114.5	389.9
7OW	37.8	76.8	80.8	80.0	75.5	88.4	90.2	310.2
7OÖ	35.4	73.3	76.2	75.0	70.9	82.9	84.6	286.8

TABLE XIII
SOLUTION TIMES [ms] IN TRANSMISSION SYSTEMS FOR THE SCENARIO #2

	System #							
	3	4	5	6	7	8	9	10
NR	91.0	157.1	196.7	193.6	196.7	177.6	239.8	680.0
BEM	130.0	270.5	285.2	280.8	269.6	292.7	319.3	1090.4
3OW	106.7	199.5	230.6	226.0	211.9	218.1	257.5	779.1
3OÖ	106.5	192.5	229.5	223.8	210.7	214.6	254.1	771.8
7OW	71.2	149.2	154.8	152.3	143.0	167.3	171.8	583.5
7OÖ	66.7	140.5	144.4	142.1	133.8	156.8	161.1	548.8

TABLE XIV
SOLUTION TIMES [ms] IN TRANSMISSION SYSTEMS FOR THE SCENARIO #3

	System #							
	3	4	5	6	7	8	9	10
NR	86.9	228.0	208.2	294.9	278.5	247.8	273.6	901.0
BEM	305.5	833.9	733.2	1007.4	936.5	893.5	1136.5	3309.3
3OW	116.8	288.9	216.6	348.4	310.0	303.3	392.3	1098.2
3OÖ	115.6	287.9	215.7	348.1	308.2	303.1	390.4	1075.9
7OW	76.4	195.4	184.7	231.6	216.8	216.2	247.9	779.0
7OÖ	71.3	193.7	181.2	226.1	212.7	211.4	240.9	732.8

F. Time Saving

This Section is devoted on analyzing the computational savings offered by the developed PF methods. Fig. 3 shows the time reducing [%] of the developed 7OW and 7OÖ with respect to the other tested methods. As observed, the developed techniques offered important savings in comparison with other solvers. Thus, the mappings (8) and (9) were able to reduce up to ~20%, ~80% and ~30% the computational time exhibited by NR, BEM and the cubic techniques (6), (7), respectively.

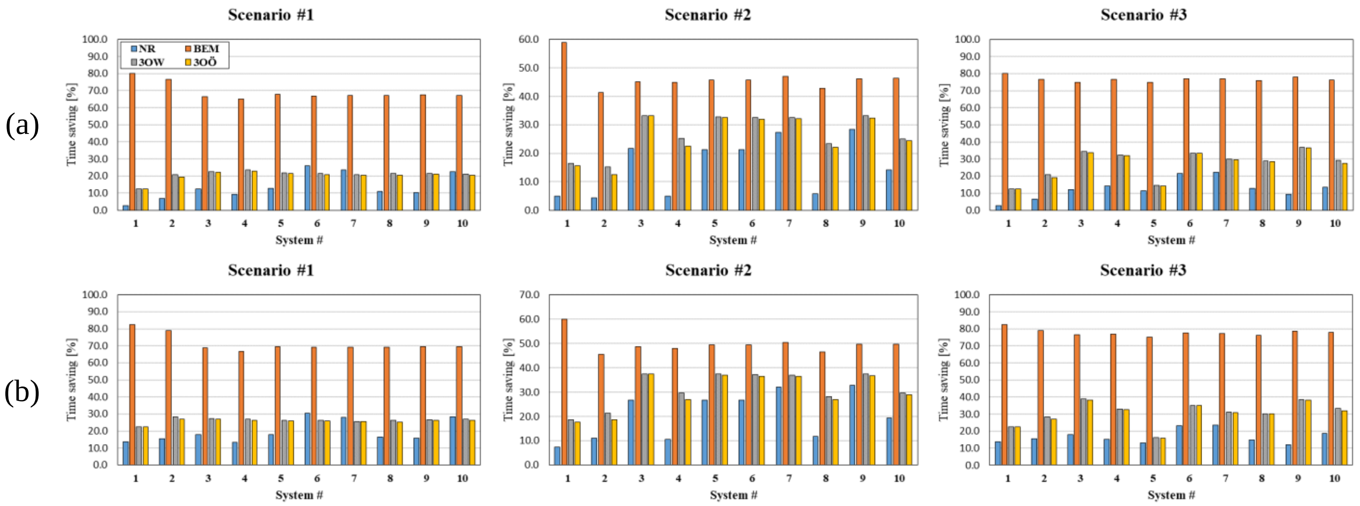

Fig. 3 Computational time saving offered by the developed 7OW (a) and 7OÖ (b) with respect to the other tested methods

G. *Scalability*

It can be also interesting explore the scalability of the developed methods to larger systems. To do that, let us consider that the total number of iterations keeps constant as the size of the system grows. This approach it reasonable for the Scenarios #1 and #2, which the total number of iterations employed by the considered methods were kept almost constant in all studied systems. Keeping this in mind, the execution time obtained for the Scenarios #1 and #2 can be properly extrapolated to larger systems. This approach is showed in Fig. 4 and 5, respectively. In this figure, it can be easily appreciated that the developed methods are more scalable than the remainder considered methodologies.

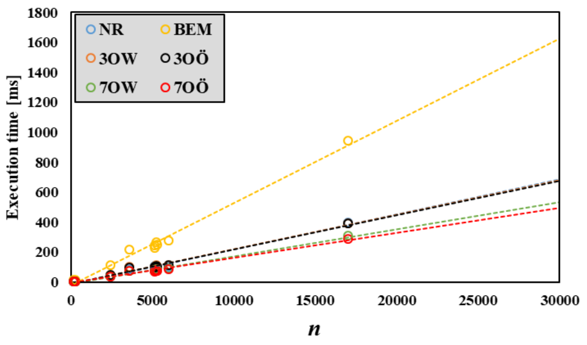

Fig. 4 Execution time of the different considered PF solution methods at Scenario #1, as function of the number of variables

In Fig. 4, it is also appreciated that the methods (6) and (7) are occasionally more scalable than NR. This is however confusing, and undoubtedly propitiated by the superior results obtained by the cubic methods for the Scenario #1 at System #10 with respect to NR. Nevertheless, as it can be seen in Fig. 5, as expected these cubic methods are less scalable than NR at Scenario #2.

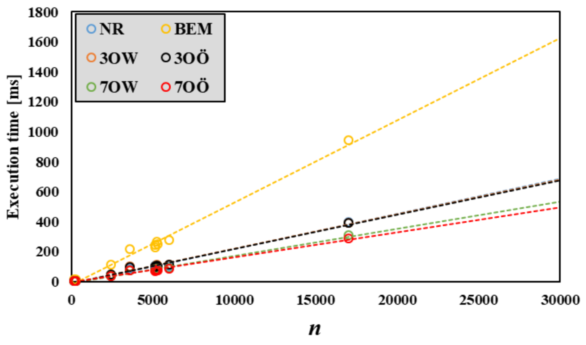

Fig. 5 Execution time of the different considered PF solution methods at Scenario #2, as function of the number of variables

H. *Ability to Calculate the Low Voltage Solution*

In previous simulations, the high voltage solution of the tested systems has been found. In this section, we analyze the ability of the developed methods for finding the low voltage solution. For this purpose, we have considered the IEEE 30-bus test system [30]. As commented in Section IV.D, the developed methods would converge to the low voltage solution if the initial guess lies within the so-called convergence region of the low voltage solution. To force our techniques to converge to the low voltage value, we have initialized their iterative procedures with $\boldsymbol{\delta}_{\boldsymbol{g}}^{(0)} = \boldsymbol{\delta}_{\boldsymbol{l}}^{(0)} = -100^\circ$ and $\boldsymbol{V}_{\boldsymbol{g}}^{(0)} = 0.7$ pu. With this initialization, both 7OW and 7OÖ converged to the low voltage solution in 2 iterations. Fig. 6 plots the solution achieved by techniques and it is compared with the high voltage solution of the system. As observed, the calculated solution presents low voltage magnitudes and more negative angles.

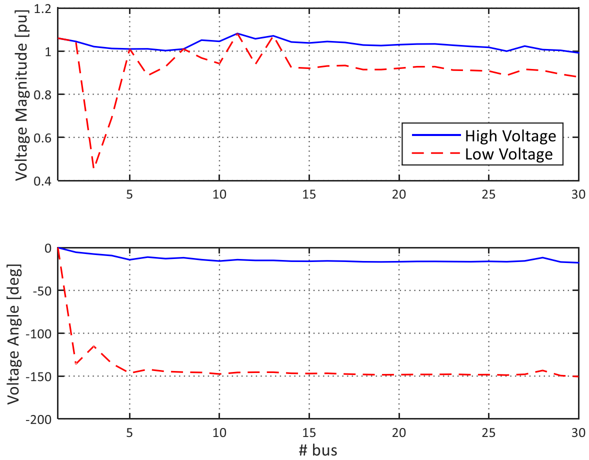

Fig. 6 High and low voltage solutions (calculated with 7OW and 7OÖ) for the IEEE 30-bus system.

However, a main question arises: how to select $\boldsymbol{x}^{(0)}$ so that the developed methods converge to the low voltage solution?. Evidently, this is the main difficulty of the Newton-based methods for calculating all the solutions of the PF problem. Nevertheless, as commented in Section IV.D, this difficulty may be easily overcome using continuation techniques such as in [25]. Thus, we have integrated the developed methods within the corrector stage of the Continuation Power Flow. For illustrating the performance of the developed methods within this tool, the P-V curve for the System #1 (bus #61) has been calculated using NR, 7OW and 7OÖ and plotted in Fig. 7. For tracing the curves, an adaptive step size has been considered. As observed, the three techniques yielded the same results with a high level of accuracy.

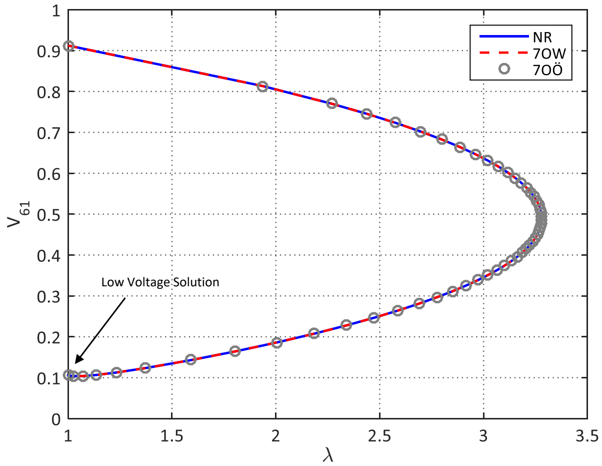

Fig. 7 P-V curve for the System #1 (bus #61): This curve has been traced using the Continuation Power Flow with NR and the developed methods at corrector stage

VI. CONCLUSIONS AND FUTURE WORKS

In this paper, two novel PF solution methods with seventh order of convergence have been developed. They are based on two classic cubic nonlinear techniques. These third order techniques are not competitive in PF analysis since the standard NR is normally more efficient. Consequently, introducing several modifications on these two cubic techniques with the aim of increasing their convergence order to seven have been proposed. The developed PF solution methods are more efficiency than NR as proved using a well-known efficiency index.

The developed methods aim at being more competitive than NR, in order to be suitable for widespread using in industry tools. Up to ten realistic distribution and transmission systems have been used for validating this point. Three scenarios including base load, maximum load level and reactive limits enforcement have been considered.

Convergence rates and solution times have been used as indexes for validating the developed methods. In all cases, the developed methods were more competitive than NR, FBS, BEM and the conventional cubic techniques. It is worth remarking that the developed methods enable computational savings higher than 20%, 80% and 30% with respect to NR, BEM and the cubic methods, respectively. It has been also checked that the developed PF solution methods are more scalable than the remainder tested techniques.

Suitability of the developed methods in other related problems should be validated in future works.

APPENDIX A: PROOF OF CONVERGENCE OF METHOD (8)

In this appendix, we demonstrate the seventh order of convergence of method (8) using the Taylor series technique (see e.g. [24]).

Theorem 4. *Let* $\mathbf{g}$ *be sufficiently differentiable at each point of an open neighborhood* D *of* $\boldsymbol{r} \in \mathbb{R}^n$*, this is a solution of the system* $\mathbf{g}(\boldsymbol{x}) = \mathbf{0}$*. Let us suppose that* $\mathbf{g}(\boldsymbol{x})$ *is continuous and nonsingular in* $\boldsymbol{x}$*. Then, the developed method (8) converges to* $\boldsymbol{r}$ *with order seven.*

Proof. Taylor expansion of $\mathbf{g}(\boldsymbol{x})$ and $\mathbf{g}'(\boldsymbol{x})$ around $\boldsymbol{r}$ yields:

$$\mathbf{g}\left(\boldsymbol{x}^{(k)}\right) = \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{e}^{(k)} + \boldsymbol{C}_2\boldsymbol{e}^{(k)^2} + \boldsymbol{C}_3\boldsymbol{e}^{(k)^3} + \boldsymbol{C}_4\boldsymbol{e}^{(k)^4} + \boldsymbol{C}_5\boldsymbol{e}^{(k)^5} + \boldsymbol{C}_6\boldsymbol{e}^{(k)^6}\right] + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^7}\right) \tag{13}$$

$$\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right) = \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{I} + 2\boldsymbol{C}_2\boldsymbol{e}^{(k)} + 3\boldsymbol{C}_3\boldsymbol{e}^{(k)^2} + 4\boldsymbol{C}_4\boldsymbol{e}^{(k)^3} + 5\boldsymbol{C}_5\boldsymbol{e}^{(k)^4} + 6\boldsymbol{C}_6\boldsymbol{e}^{(k)^5}\right] + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^6}\right) \tag{14}$$

where $\boldsymbol{e}^{(k)} = \boldsymbol{x}^{(k)} - \boldsymbol{r} \in \mathbb{R}^n$, $\boldsymbol{e}^i = \underbrace{(\boldsymbol{e}, \boldsymbol{e}, \dots, \boldsymbol{e})}_{i-\text{times}}$, $\boldsymbol{C}_j = (1/j!)[\mathbf{g}'(\boldsymbol{r})]^{-1}\mathbf{g}^{(j)}(\boldsymbol{r}) \in L_i(\mathbb{R}^n, \mathbb{R}^n)$, $\mathbf{g}^{(j)} \in L(\mathbb{R}^n \times \dots \times \mathbb{R}^n, \mathbb{R}^n)$, $[\mathbf{g}'(\boldsymbol{r})]^{-1} \in L(\mathbb{R}^n)$ and $\boldsymbol{I} \in \mathbb{R}^{n\times n}$ is the identity matrix.

Now, let us assume that:

$$\left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1} = \left[a_1\boldsymbol{I} + a_2\boldsymbol{e}^{(k)} + a_3\boldsymbol{e}^{(k)^2} + a_4\boldsymbol{e}^{(k)^3} + a_5\boldsymbol{e}^{(k)^4} + a_6\boldsymbol{e}^{(k)^5}\right][\mathbf{g}'(\boldsymbol{r})]^{-1} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^6}\right) \tag{15}$$

where, a's$\in \mathbb{R}$.

Considering the following inverse definition:

$$\left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right) = \mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1} = \boldsymbol{I} \tag{16}$$

And solving the resulting linear system, we can obtain:

$$\left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1} = \left[\boldsymbol{I} - 2\boldsymbol{C}_2\boldsymbol{e}^{(k)} + (4\boldsymbol{C}_2^2 - 3\boldsymbol{C}_3)\boldsymbol{e}^{(k)^2} + (12\boldsymbol{C}_2\boldsymbol{C}_3 - 8\boldsymbol{C}_2^3 - 4\boldsymbol{C}_4)\boldsymbol{e}^{(k)^3} + (16\boldsymbol{C}_2^4 - 36\boldsymbol{C}_2^2\boldsymbol{C}_3 + 16\boldsymbol{C}_2\boldsymbol{C}_4 + 9\boldsymbol{C}_3^2 - 5\boldsymbol{C}_5)\boldsymbol{e}^{(k)^4} + \left(96\boldsymbol{C}_2^2\boldsymbol{C}_3 - 32\boldsymbol{C}_2^5 - 48\boldsymbol{C}_2^2\boldsymbol{C}_4 - 54\boldsymbol{C}_2\boldsymbol{C}_3^2 + 20\boldsymbol{C}_2\boldsymbol{C}_5 + 24\boldsymbol{C}_3\boldsymbol{C}_4 - 6\boldsymbol{C}_6\right)\boldsymbol{e}^{(k)^5}\right][\mathbf{g}'(\boldsymbol{r})]^{-1} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^6}\right) \tag{17}$$

Now, let us take $\boldsymbol{e}_{\boldsymbol{y}}^{(k)} = \boldsymbol{y}^{(k)} - \boldsymbol{r} \in \mathbb{R}^n$ and $\boldsymbol{e}_{\boldsymbol{z}}^{(k)} = \boldsymbol{z}^{(k)} - \boldsymbol{r} \in \mathbb{R}^n$, observing the first and second steps of iterative algorithm (8), we can obtain:

$$\boldsymbol{e}_{\boldsymbol{y}}^{(k)} = \boldsymbol{C}_2\boldsymbol{e}^{(k)^2} + (2\boldsymbol{C}_3 - 2\boldsymbol{C}_2^2)\boldsymbol{e}^{(k)^3} + (4\boldsymbol{C}_2^3 - 7\boldsymbol{C}_2\boldsymbol{C}_3 + 3\boldsymbol{C}_4)\boldsymbol{e}^{(k)^4} + (20\boldsymbol{C}_2^2\boldsymbol{C}_3 - 8\boldsymbol{C}_2^4 - 10\boldsymbol{C}_2\boldsymbol{C}_4 - 6\boldsymbol{C}_3^2 + 4\boldsymbol{C}_5)\boldsymbol{e}^{(k)^5} + \left(16\boldsymbol{C}_2^5 - 52\boldsymbol{C}_2^3\boldsymbol{C}_3 + 28\boldsymbol{C}_2^2\boldsymbol{C}_4 + 33\boldsymbol{C}_2\boldsymbol{C}_3^2 - 13\boldsymbol{C}_2\boldsymbol{C}_5 - 17\boldsymbol{C}_3\boldsymbol{C}_4 + 5\boldsymbol{C}_6\right)\boldsymbol{e}^{(k)^6} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^7}\right) \tag{18}$$

$$\mathbf{g}\left(\boldsymbol{y}^{(k)}\right) = \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{e}_{\boldsymbol{y}}^{(k)} + \boldsymbol{C}_2\boldsymbol{e}_{\boldsymbol{y}}^{(k)^2} + \boldsymbol{C}_3\boldsymbol{e}_{\boldsymbol{y}}^{(k)^3} + \boldsymbol{C}_4\boldsymbol{e}_{\boldsymbol{y}}^{(k)^4} + \boldsymbol{C}_5\boldsymbol{e}_{\boldsymbol{y}}^{(k)^5} + \boldsymbol{C}_6\boldsymbol{e}_{\boldsymbol{y}}^{(k)^6}\right] + \boldsymbol{O}\left(\boldsymbol{e}_{\boldsymbol{y}}^{(k)^7}\right) \tag{19}$$

$$\boldsymbol{e}_{\boldsymbol{z}}^{(k)} = 2\boldsymbol{C}_2^2\boldsymbol{e}^{(k)^3} + (-9\boldsymbol{C}_2^3 + 7\boldsymbol{C}_2\boldsymbol{C}_3)\boldsymbol{e}^{(k)^4} + (30\boldsymbol{C}_2^4 - 44\boldsymbol{C}_2^2\boldsymbol{C}_3 + 10\boldsymbol{C}_2\boldsymbol{C}_4 + 6\boldsymbol{C}_3^2)\boldsymbol{e}^{(k)^5} + \left(-88\boldsymbol{C}_2^5 + 188\boldsymbol{C}_2^3\boldsymbol{C}_3 - 62\boldsymbol{C}_2^2\boldsymbol{C}_4 - 70\boldsymbol{C}_2\boldsymbol{C}_3^2 + 13\boldsymbol{C}_2\boldsymbol{C}_5 + 17\boldsymbol{C}_3\boldsymbol{C}_4\right)\boldsymbol{e}^{(k)^6} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^7}\right) \tag{20}$$

Now, as in (17), we can calculate:

$$\left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right) + \mathbf{g}'\left(\boldsymbol{y}^{(k)}\right)\right]^{-1} = \left[\frac{1}{2}\boldsymbol{I} - \frac{1}{2}\boldsymbol{C}_2^2\boldsymbol{e}^{(k)^2} - \boldsymbol{C}_2\boldsymbol{C}_3\boldsymbol{e}^{(k)^3} + \left(3\boldsymbol{C}_2^4 - \frac{3}{4}\boldsymbol{C}_2^2\boldsymbol{C}_3 - \frac{3}{2}\boldsymbol{C}_2\boldsymbol{C}_4\right)\boldsymbol{e}^{(k)^4} + \left(17\boldsymbol{C}_2^3\boldsymbol{C}_3 - 11\boldsymbol{C}_2^5 - 3\boldsymbol{C}_2\boldsymbol{C}_3^2 - 2\boldsymbol{C}_2\boldsymbol{C}_5\right)\boldsymbol{e}^{(k)^5}\right][\mathbf{g}'(\boldsymbol{r})]^{-1} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^6}\right) \tag{21}$$

Taking $\boldsymbol{e}_{\boldsymbol{w}}^{(k)} = \boldsymbol{w}^{(k)} - \boldsymbol{r} \in \mathbb{R}^n$ and proceeding as the third step of method (8), we can obtain:

$$\boldsymbol{e}_{\boldsymbol{w}}^{(k)} = 2\boldsymbol{C}_2^4\boldsymbol{e}^{(k)^5} + \left(\frac{23}{2}\boldsymbol{C}_2^3\boldsymbol{C}_3 - 13\boldsymbol{C}_2^5\right)\boldsymbol{e}^{(k)^6} + (22\boldsymbol{C}_2^6 - 15\boldsymbol{C}_2^4\boldsymbol{C}_3 - 16\boldsymbol{C}_2^3\boldsymbol{C}_4 - 11\boldsymbol{C}_2^2\boldsymbol{C}_3^2 + 8\boldsymbol{C}_2^2\boldsymbol{C}_5 + 12\boldsymbol{C}_2\boldsymbol{C}_3\boldsymbol{C}_4)\boldsymbol{e}^{(k)^7} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^8}\right) \tag{22}$$

$$\mathbf{g}\left(\boldsymbol{w}^{(k)}\right) = \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{e}_{\boldsymbol{w}}^{(k)} + \boldsymbol{C}_2\boldsymbol{e}_{\boldsymbol{w}}^{(k)^2} + \boldsymbol{C}_3\boldsymbol{e}_{\boldsymbol{w}}^{(k)^3} + \boldsymbol{C}_4\boldsymbol{e}_{\boldsymbol{w}}^{(k)^4} + \boldsymbol{C}_5\boldsymbol{e}_{\boldsymbol{w}}^{(k)^5} + \boldsymbol{C}_6\boldsymbol{e}_{\boldsymbol{w}}^{(k)^6}\right] + \boldsymbol{O}\left(\boldsymbol{e}_{\boldsymbol{w}}^{(k)^7}\right) \tag{23}$$

And proceeding as the fourth step of (8), we finally obtain:

$$\boldsymbol{e}^{(k+1)} = 2\boldsymbol{C}_2^6\boldsymbol{e}^{(k)^7} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^8}\right) \tag{24}$$

The proof is completed. □

APPENDIX B: PROOF OF CONVERGENCE OF METHOD (9)

In this appendix, we demonstrate the seventh order of convergence of method (9) using the Taylor series technique.

Theorem 5. *Let* $\mathbf{g}$ *be sufficiently differentiable at each point of an open neighborhood* D *of* $\boldsymbol{r} \in \mathbb{R}^n$*, this is a solution of the system* $\mathbf{g}(\boldsymbol{x}) = \mathbf{0}$*. Let us suppose that* $\mathbf{g}(\boldsymbol{x})$ *is continuous and nonsingular in* $\boldsymbol{x}$*. Then, the developed method (9) converges to* $\boldsymbol{r}$ *with order seven.*

Proof. Taking into account that $\boldsymbol{e}_{\boldsymbol{y}}$, $\mathbf{g}(\boldsymbol{y})$ and $\boldsymbol{e}_{\boldsymbol{z}}$ take the same values of (18), (19) and (20), respectively, and considering the inverse definition (16), we can directly calculate:

$$\left[\mathbf{g}'\left(\frac{\boldsymbol{y}^{(k)}+\boldsymbol{z}^{(k)}}{2}\right)\right]^{-1} = \left[\boldsymbol{I} - \boldsymbol{C}_2^2\boldsymbol{e}^{(k)^2} - 2\boldsymbol{C}_2\boldsymbol{C}_3\boldsymbol{e}^{(k)^3} + \left(6\boldsymbol{C}_2^4 - \frac{3}{4}\boldsymbol{C}_2^2\boldsymbol{C}_3 - 3\boldsymbol{C}_2\boldsymbol{C}_4\right)\boldsymbol{e}^{(k)^4} + \left(28\boldsymbol{C}_2^3\boldsymbol{C}_3 - 22\boldsymbol{C}_2^5 - 3\boldsymbol{C}_2\boldsymbol{C}_3^2 - 4\boldsymbol{C}_2\boldsymbol{C}_5\right)\boldsymbol{e}^{(k)^5}\right][\mathbf{g}'(\boldsymbol{r})]^{-1} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^6}\right) \tag{25}$$

Taking $\boldsymbol{e}_{\boldsymbol{w}}^{(k)} = \boldsymbol{w}^{(k)} - \boldsymbol{r}$ and proceeding as the third step of method (9), we can obtain:

$$\boldsymbol{e}_{\boldsymbol{w}}^{(k)} = 2\boldsymbol{C}_2^4\boldsymbol{e}^{(k)^5} + \left(\frac{43}{4}\boldsymbol{C}_2^2\boldsymbol{C}_3 - 13\boldsymbol{C}_2^5\right)\boldsymbol{e}^{(k)^6} + \left(22\boldsymbol{C}_2^6 - \frac{15}{2}\boldsymbol{C}_2^4\boldsymbol{C}_3 - 16\boldsymbol{C}_2^3\boldsymbol{C}_4 - \frac{31}{2}\boldsymbol{C}_2^2\boldsymbol{C}_3^2 + 8\boldsymbol{C}_2^2\boldsymbol{C}_5 + 12\boldsymbol{C}_2\boldsymbol{C}_3\boldsymbol{C}_4\right)\boldsymbol{e}^{(k)^7} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^8}\right) \tag{26}$$

$$\mathbf{g}\left(\boldsymbol{w}^{(k)}\right) = \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{e}_{\boldsymbol{w}}^{(k)} + \boldsymbol{C}_2\boldsymbol{e}_{\boldsymbol{w}}^{(k)^2} + \boldsymbol{C}_3\boldsymbol{e}_{\boldsymbol{w}}^{(k)^3} + \boldsymbol{C}_4\boldsymbol{e}_{\boldsymbol{w}}^{(k)^4} + \boldsymbol{C}_5\boldsymbol{e}_{\boldsymbol{w}}^{(k)^5} + \boldsymbol{C}_6\boldsymbol{e}_{\boldsymbol{w}}^{(k)^6}\right] + \boldsymbol{O}\left(\boldsymbol{e}_{\boldsymbol{w}}^{(k)^7}\right) \tag{27}$$

And proceeding as the fourth step of (9), we finally obtain:

$$\boldsymbol{e}^{(k+1)} = 2\boldsymbol{C}_2^6\boldsymbol{e}^{(k)^7} + \boldsymbol{O}\left(\boldsymbol{e}^{(k)^8}\right) \tag{28}$$

The proof is completed. □

REFERENCES

[1] J.J. Grainger, W.D. Stevenson, *Power System Analysis*. New York: McGraw-Hill, 1994.

[2] P. Kundur, *Power System Stability and Control*, New York: McGraw-Hill, 1994.

[3] A. Adel Mohamed and B. Venkatesh, "Line-Wise Power Flow and Voltage Collapse," *IEEE Trans. Power Syst.*, vol. 33, no. 4, pp. 3768-3778, July 2018.

[4] M. Tostado-Véliz, S. Kamel and F. Jurado, "A Robust Power Flow Algorithm Based on Bulirsch–Stoer Method," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 3081-3089, July 2019.

[5] G. Arcia-Garibaldi, P. Cruz-Romero and A. Gómez-Expósito, "Future power transmission: Visions, technologies and challenges," *Renewable Sustain. Energy. Rev.*, vol. 94, pp. 285-301, Oct. 2018.

[6] B. Stott and O. Alsac, "Fast Decoupled Load Flow," *IEEE Trans. Power Appar. Syst.*, vol. PAS-93, no. 3, pp. 859-869, May 1974.

[7] R. Idema and D.J.P. Lahaye, Computational methods in Power System Analysis, in *Atlantis Studies in Scientific Computing in Electromagnetics*, Ed. W. Schilders. Paris, France: Atlantis Press, 2014.

[8] V. M. da Costa, N. Martins and J. L. R. Pereira, "Developments in the Newton Raphson power flow formulation based on current injections," *IEEE Trans. Power Syst.*, vol. 14, no. 4, pp. 1320-1326, Nov. 1999.

[9] M. J. E. Alam, K. M. Muttaqi, and D. Sutanto, "A three-phase power flow approach for integrated 3-wire MV and 4-wire multigrounded LV networks with rooftop solar PV," *IEEE Trans. Power Syst.*, vol. 28, no. 2, pp. 1728–1737, May 2013.

[10] S.A. Saleh, "The Formulation of a Power Flow Using d-q Reference Frame Components-Part I: Balanced 3ϕ Systems," *IEEE Trans. Ind. Appl.*, vol. 52, no. 5, pp. 3682-3693, Sept/Oct. 2016.

[11] S. M. Fatemi, S. Abedi, G. B. Gharehpetian, S. H. Hosseinian and M. Abedi, "Introducing a Novel DC Power Flow Method With Reactive Power Considerations," *IEEE Trans. Power Syst.*, vol. 30, no. 6, pp. 3012-3023, Nov. 2015.

[12] S. Iwamoto and Y. Tamura, "A Load Flow Calculation Method for Ill-Conditioned Power Systems," *IEEE Trans. Power Appar. Syst.*, vol. PAS-100, no. 4, pp. 1736-1743, Apr. 1981.

[13] F. Milano, "Continuous Newton's Method for Power Flow Analysis," *IEEE Trans. Power Syst.*, vol. 24, no. 1, pp. 50-57, Feb. 2009.

[14] F. Milano, "Implicit Continuous Newton Method for Power Flow Analysis," *IEEE Trans. Power Syst.*, vol. 34, no. 4, pp. 3309-3311, July 2019.

[15] R. Pourbagher and S. Y. Derakhshandeh, "Application of high-order Levenberg–Marquardt method for solving the power flow problem in the ill-conditioned systems," *IET Gener. Transmiss. Distrib.*, vol. 10, no. 12, pp. 3017-3022, July 2016.

[16] M. Tostado-Véliz, S. Kamel and F. Jurado, "A powerful power-flow method based on Composite Newton-Cotes formula for ill-conditioned power systems," *Int. J. Elect. Power Energy Syst.*, vol. 116, Mar. 2020, Art. no. 105558

[17] X.Y. Xiao and H.W. Yin, "Accelerating the convergence speed of iterative methods for solving nonlinear systems," *Appl. Math. Comput.*, vol. 333, pp. 8-19, Sept. 2018.

[18] C. Chun and B. Neta, "Developing high order methods for the solution of systems of nonlinear equations," *Appl. Math. Comput.*, vol. 342, pp. 178-190, Feb. 2019.

[19] S. Weerakoon and T.G.I. Fernando, "A variant of Newton's method with accelerated third order convergence," *Appl. Math. Lett.*, vol. 13, no. 8, pp. 87-93, Nov. 2000.

[20] A.Y. Özban, "Some New Variants of Newton's Method," *Appl. Math. Lett.*, vol. 17, no. 6, pp. 677-682, Jun. 2004.

[21] H. Saadat, *Power System Analysis*, 3rd ed., North York, Toronto, Canada: PSA Publishing, 2010.

[22] D.K.R. Babajee, M.Z. Dauhoo, M.T. Darvishi, A. Karami and A. Barati, "Analysis of two Chebyshev-like third order methods free from second derivatives for solving systems of nonlinear equations," *J. Comput. Appl. Math.*, vol. 233, no. 8, pp. 2002-2012, Feb. 2010.

[23] T. Lotfi, P. Bakhtiari, A. Cordero, K. Mahdiani and J.R. Torregrosa, "Some new efficient multipoint iterative methods for solving nonlinear systems of equations", *Int. J. Comput. Math.*, vol. 92, no. 9, pp. 1921-1934, 2015.

[24] A. Cordero, J.L. Hueso, E. Martínez and J.R. Torregrosa, "A modified Newton-Jarratt's composition," *Numer. Algor.*, vol. 55, no. 1, pp. 87-99, Sept. 2010.

[25] V. Ajjarapu and C. Christy, "The continuation power flow: a tool for steady state voltage stability analysis," *IEEE Trans. Power Syst.*, vol. 7, no. 1, pp. 416-423, Feb. 1992.

[26] R. D. Zimmerman, C. E. Murillo-Sánchez, and R. J. Thomas, "Matpower: Steady-State Operations, Planning and Analysis Tools for Power Systems Research and Education," *IEEE Trans. Power Syst.*, vol. 26, no. 1, pp. 12-19, Feb. 2011.

[27] [Online] Matpower's download site. Available: https://matpower.org/download/ [Accessed 18 Mar. 2020]

[28] C. Josz, S. Fliscounakis, J. Maeght, P. Panciatici, AC Power Flow Data in Matpower and QCQP Format: iTesla, RTE Snapshots, and PEGASE. *arXiv: 1603.01533.* Available: http://arxiv.org/abs/1603.01533. [Accessed 18 Mar. 2020]

[29] S. Fliscounakis, P. Panciatici, F. Capitanescu and L. Wehenkel, "Contingency Ranking With Respect to Overloads in Very Large Power Systems Taking Into Account Uncertainty, Preventive, and Corrective Actions," *IEEE Trans. Power Syst.*, vol. 28, no. 4, pp. 4909-4917, Nov. 2013.

[30] A.B. Birchfield, T. Xu, K.M. Gegner, K.S. Shetye and T.J. Overbye, "Grid Structural Characteristics as Validation Criteria for Synthetic Networks," *IEEE Trans. Power Syst.*, vol. 32, no. 4, pp. 3258-3265, July 2017.

[31] [Online] Matpower's Limit Cases. Available: https://zenodo.org/record/3491654 [Accessed 18 Mar. 2020]

[32] [Online] Matpower's manual. Available: https://matpower.org/docs/manual.pdf [Accessed 18 Mar. 2020]

[33] W. H. Kersting, *Distribution System Modeling and Analysis*, 2nd ed. Boca Raton, FL: CRC Press, 2006.

[34] F. Milano, *Power System Modelling and Scripting*. New York: Springer, 2010.

Marcos Tostado-Véliz was born in Spain in 1987. He received the Electrical Engineering and Master's degrees in 2016 and 2017, respectively, from the University of Seville, Seville, Spain. He is currently working toward the Ph.D. degree in University of Jaén, Jaén, Spain. His primary research interests include optimal power system planning, operation, and control and numerical methods for power system analysis.

Salah Kamel received the international Ph.D. degree in Electrical Engineering from University of Jaen, Spain (Main) and Aalborg University, Denmark (Host) in 2014. He is an Associate Professor in Electrical Engineering Department, Aswan University. His research activities include power system analysis and optimization, smart grid and renewable energy systems.

Francisco Jurado (M'00–SM'06) obtained the MSc and PhD degrees, both in Electrical Engineering, from the National University of Distance Education, Madrid, Spain, in 1995 and 1999, respectively. He is a Full Professor at the Department of Electrical Engineering of University of Jaén, Spain. His research activities have focused on two topics: power systems and renewable energy.