A novel family of efficient Power-Flow methods with high convergence rate suitable for large realistic power systems

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

Abstract*— *High order Power-Flow (PF) solution methods are techniques with higher convergence rate than the standard Newton-Raphson (NR). This feature normally provokes that less iterations are necessary for achieving the required convergence tolerance. This advantage enables important computational savings in realistic large scale power systems, since some expensive computations may be avoided. This paper develops and analyzes a novel family of multistep PF solution techniques. The introduced solution paradigm just involves a LU decomposition along other cheap computations each iteration. In addition, it is proved that its convergence order is equal to the number of steps considered. A comparison with other high order PF techniques available in the literature, reveals that the introduced paradigm is more efficient when more than three steps are taken. Hence, two PF methods with fourth and fifth convergence rate are developed and validated in twelve systems under different demand conditions. Results prove that the developed PF solution methods are able to notably outperform NR and the other high order PF solvers proposed in the literature.

***Index Terms*— Power-Flow analysis, High order methods, Computational efficiency, Large realistic power systems.**

I. Introduction

PF is a nonlinear problem that plays a key role in power systems analysis [1]. Planification purposes, security analysis or control, optimal power flow analysis, along with serving as starting point for dynamic simulations, are some examples of the huge variety of applications of PF in power system analysis.

The solution of PF problem by NR was firstly proposed in [2]. Given the advantage of preserving Jacobian sparsity, NR became fully competitive [3]. Although nowadays NR can be considered as the most standard PF method, the solution of PF equations has attracted huge attention during decades by developing different techniques and formulations. Interested readers can find comprehensive literature reviews about this topic in [4]-[6].

NR has quadratic local convergence. An iteration of NR involves a LU decomposition along the solution of a linear system and a function evaluation. While the solution of the linear system and function evaluation do not suppose a significant computational cost, the LU decomposition is, by far, the heaviest computation involved in the solution of the PF problem using NR [7]. Computational cost of a LU decomposition is $O(n^3)$, being n the number of variables involved. Consequently, it becomes the most predominant aspect in the overall computational performance of NR in large-scale realistic systems.

Unlike NR, high order methods have convergence rate greater than two. Hence, this kind of techniques are usually able to save some iterations in comparison with NR. Thus, high order techniques might reduce the computational cost of a PF calculation by saving some calculations. It is worth mentioning that this sentence is only true if the considered method does not involve heavy calculations. Since NR is widely considered as the most standard PF solver, any other methodology which just requires a matrix factorization each iteration should be considered efficient enough, in the sense that it presents similar computational burden in comparison with NR.

To the best of our knowledge, only references [8], [9] have studied the application of high order methods to PF analysis. In [8], several high order methods previously developed in [10]-[12], were applied to PF analysis. On the other hand, the nonlinear solver proposed in [13], was adapted to systems of n nonlinear equations and applied to solve the PF problem in [9]. Table I summarizes the main computations involved each iteration of the referenced methods. From this table, it can be concluded that only the Darvishi's 3rd order and the Predictor-Corrector can be considered efficient. Moreover, the Jarrat's 4th order method implies the solution of a matrix system, which has a cost $O(n^{3n})$.

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. S. Kamel is currently with State Key Laboratory of Power Transmission Equipment & System Security and New Technology, Chongqing University, Chongqing, China 400030 (email: skamel@aswu.edu.eg).

TABLE I
MAIN COMPUTATIONS OF DIFFERENT HIGH ORDER PF SOLUTION METHODS

Method	Convergence order	Functions evaluations	LU decompositions	Linear systems solved	Matrix systems solved
Darvishi's 3rd order [8], [10]	3	2	1	2	0
Jarrat's 4th order [8], [11]	4	1	2	1	1
Sharma's 5th order [8], [12]	5	2	2	2	0
Predictor-Corrector [9], [13]	$1+\sqrt{2}$	1	1	1	0

Therefore, developing efficient high order PF solution methods may still be a hot topic and needs further research. In this paper, we address this issue by developing a novel family of multistep PF solution methods. It is demonstrated that the developed paradigm achieves N^{th} order of convergence (where N is the total number of steps considered). In addition, only one LU decomposition along with $N-1$ linear system solutions and function evaluations, are required each iteration. Thus, the developed approach can be considered efficient. Different PF solution techniques for $N=4$ and $N=5$ are compared with NR and the high order methods [8], [9], using twelve systems under different demand scenarios.

Remainder of this paper is organized as follows. The novel family of PF solution methods is introduced in Section II. Convergence analysis of the developed PF solution paradigm is carried out in Section III. The developed paradigm is compared with other PF solvers from a computational efficient perspective in Section IV. In Section V, numerical results with discussion are presented. Finally, the paper is concluded in Section VI.

II. DEVELOPED FAMILY OF PF SOLUTION METHODS

A. Brief description of the PF problem and its solution using NR

Typically, the PF formulation in polar coordinates is preferred, due to the size of the system is reduced compared with the rectangular form [14]. In this case, the PF problem can be formulated as a set of n nonlinear equations as follows:

$$\mathbf{g}(\boldsymbol{x}) = \mathbf{0} \tag{1}$$

where, $\boldsymbol{x} \in \mathbb{R}^n$ is the vector of unknowns and $\boldsymbol{g}: \mathbb{R}^n \mapsto \mathbb{R}^n$ are nonlinear, smooth equations defining the active and reactive power balance at network buses (see Appendix A for details).

NR supposes the most standard technique for solving the PF equations (1). This method proceeds for a generic k^{th} iteration as follows:

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} - \left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\boldsymbol{x}^{(k)}\right) \tag{2}$$

where, $\mathbf{g}'(\boldsymbol{x}) = \nabla_x^T \mathbf{g}(\boldsymbol{x}) \in \mathbb{R}^{n\times n}$ is the Jacobian matrix associated with (1). Typically, the algorithm stops when the maximum mismatch is lower than a given convergence tolerance $\varepsilon \in \mathbb{R}^+$. This condition can be expressed as follows:

$$\|\mathbf{g}(\boldsymbol{x})\|_\infty \le \varepsilon \tag{3}$$

where $\|\cdot\|_\infty$ is the infinity norm.

B. Developed multistep PF solution paradigm

The novel family of multistep high order PF solution methods is obtained on the basis of the following result [15], [16].

Theorem 1. *Let* $\mathbf{f_1}(\boldsymbol{x})$ *and* $\mathbf{f_2}(\boldsymbol{x})$ *be fixed point functions of the nonlinear equation* $\mathbf{g}(\boldsymbol{x}) = \mathbf{0}$. *Let the iterative methods* $\boldsymbol{x}^{(k+1)} = \mathbf{f_1}(\boldsymbol{x})$ *and* $\boldsymbol{x}^{(k+1)} = \mathbf{f_2}(\boldsymbol{x})$ *be of order* $p1$ *and* $p2$, *respectively. Then, the convergence order of the iterative method corresponding to the fixed point function* $\mathbf{f} = \mathbf{f_1}(\mathbf{f_2}(\boldsymbol{x})\,)$ *is* $p1p2$

A direct application of this theorem yields to nest two NR algorithms (2). Thus, it would be obtained a fourth order method. However, this technique would require two LU decompositions each iteration. Consequently, it would not be efficient. Alternatively, Jacobian matrix may be taken fixed and, therefore, LU decomposition can be reused several steps. This is the basis of multistep (or multipoint) methods [10]. Obtained algorithm generally presents lower convergence order than $p1p2$, however, this is normally offset by a high degree of efficiency.

On the basis of this idea, we develop a family of PF solution methods of the form:

$$\boldsymbol{x}^{(k)} = \boldsymbol{y}_1^{(k)} \tag{4a}$$

$$\boldsymbol{y}_i^{(k)} = \frac{1}{i-1}\sum_{j=1}^{i-1}\boldsymbol{y}_j^{(k)} - i\left[\mathbf{g}'\left(\boldsymbol{x}^{(k)}\right)\right]^{-1}\mathbf{g}\left(\frac{1}{i-1}\sum_{j=1}^{i-1}\boldsymbol{y}_j^{(k)}\right), \text{for } i = 2,3\ldots,N \tag{4b}$$

$$\boldsymbol{x}^{(k+1)} = \frac{1}{N}\sum_{j=1}^{N}\boldsymbol{y}_j^{(k)} \tag{4c}$$

Table II presents the main computations involved in an iteration of the algorithm (4). As can be seen, this technique can be considered efficient. Also, it is worth mentioning that some computations can be avoided, for example, it should be observed that:

$$\sum_{j=1}^{i-1}\boldsymbol{y}_j^{(k)} = \boldsymbol{y}_{i-1}^{(k)} + \sum_{j=1}^{i-2}\boldsymbol{y}_j^{(k)} \tag{5}$$

The second term of the right hand side of (5) may be reused from the previous step $(i-1)$. For the sake of completeness, Fig. 1 shows the flowchart of the developed methodology for solving the PF equations (1).

TABLE II
MAIN COMPUTATIONS OF THE DEVELOPED FAMILY OF PF SOLUTION METHODS

Computation	Nº of computations per iteration
Functions ev.	$N-1$
LU decomp.	1
Linear syst. solv.	$N-1$
Matrix syst. solv.	0

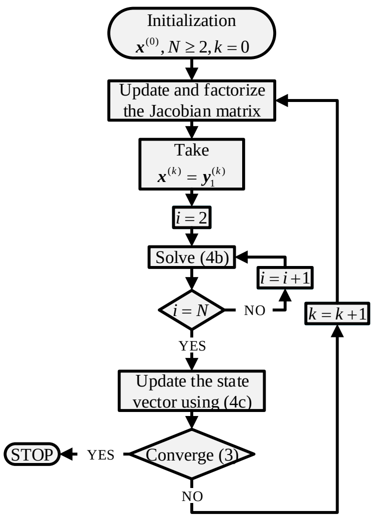

Fig. 1. Flowchart of the developed PF solution paradigm.

III. CONVERGENCE ANALYSIS OF THE NOVEL FAMILY OF PF SOLUTION METHODS

In this section, we demonstrate that the developed multistep PF solution paradigm, can achieve N order of convergence.

Theorem 2. *Let* $\mathbf{g}: \mathrm{D} \subseteq \mathbb{R}^n \mapsto \mathbb{R}^n$ *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 sequence* $\{\boldsymbol{x}^{(k)}\}_{k\geq 0}$ *obtained using the developed PF solution algorithm (4) converges to $\boldsymbol{r}$ with order N.*

Proof. From Taylor expansion of $\boldsymbol{g}(\boldsymbol{y}_1^{(k)})$ and $\boldsymbol{g}'(\boldsymbol{y}_1^{(k)})$ about $\boldsymbol{r}$ we have:

$$\mathbf{g}(\boldsymbol{y}_1^{(k)}) \approx \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{e}_1^{(k)} + \boldsymbol{C}_2 \boldsymbol{e}_1^{(k)^2} + \boldsymbol{C}_3 \boldsymbol{e}_1^{(k)^3} + \boldsymbol{C}_4 \boldsymbol{e}_1^{(k)^4}\right] + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^5}\right) \qquad (6)$$

$$\mathbf{g}'(\boldsymbol{y}_1^{(k)}) = \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{I} + 2\boldsymbol{C}_2 \boldsymbol{e}_1^{(k)} + 3\boldsymbol{C}_3 \boldsymbol{e}_1^{(k)^2} + 4\boldsymbol{C}_4 \boldsymbol{e}_1^{(k)^3}\right] + \boldsymbol{O}\left(\boldsymbol{e}_1^{(k)^4}\right) \qquad (7)$$

where, $\boldsymbol{C}_j = (1/j!)[\mathbf{g}'(\boldsymbol{r})]^{-1}\mathbf{g}^{(j)}(\boldsymbol{r}) \in \mathbb{R}^{n\times n}$, $j = 2,3,\ldots,$ and $\boldsymbol{e}_1 = \boldsymbol{y}_1 - \boldsymbol{r} \in \mathbb{R}^n$

Inversion of (7) yields (see [16]):

$$[\mathbf{g}'(\boldsymbol{y}_1^{(k)})]^{-1} = \left[\boldsymbol{I} - 2\boldsymbol{C}_2 \boldsymbol{e}_1^{(k)} + (4\boldsymbol{C}_2 - 3\boldsymbol{C}_3)\boldsymbol{e}_1^{(k)^2}\right][\mathbf{g}'(\boldsymbol{r})]^{-1} + \boldsymbol{O}\left(\boldsymbol{e}_1^{(k)^3}\right) \qquad (8)$$

where, $\boldsymbol{I} \in \mathbb{R}^{n\times n}$ is the identity matrix. Now, taking $i=2$ we have:

$$\boldsymbol{y}_2^{(k)} - \boldsymbol{r} = \boldsymbol{y}_1^{(k)} - 2\left[\mathbf{g}'(\boldsymbol{y}_1^{(k)})\right]^{-1}\mathbf{g}(\boldsymbol{y}_1^{(k)}) - \boldsymbol{r} = -\boldsymbol{e}_1^{(k)} + 2\boldsymbol{C}_2\boldsymbol{e}_1^{(k)^2} - 4(\boldsymbol{C}_2^2 - \boldsymbol{C}_3)\boldsymbol{e}_1^{(k)^3} - 2(-4\boldsymbol{C}_2^3 + 7\boldsymbol{C}_2\boldsymbol{C}_3 - 3\boldsymbol{C}_4)\boldsymbol{e}_1^{(k)^4} - 2(-8\boldsymbol{C}_2^4 + 16\boldsymbol{C}_2^2\boldsymbol{C}_3 - 6\boldsymbol{C}_2\boldsymbol{C}_4 - 3\boldsymbol{C}_3^2)\boldsymbol{e}_1^{(k)^5} + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^6}\right) \qquad (9)$$

At this point we can write:

$$\underbrace{\tfrac{1}{2}\textstyle\sum_{j=1}^{2}\boldsymbol{y}_j^{(k)} - \boldsymbol{r}}_{\boldsymbol{e}_2^{(k)}} = \boldsymbol{C}_2\boldsymbol{e}_1^{(k)^2} - 2(\boldsymbol{C}_2^2 - \boldsymbol{C}_3)\boldsymbol{e}_1^{(k)^3} - (-4\boldsymbol{C}_2^3 + 7\boldsymbol{C}_2\boldsymbol{C}_3 - 3\boldsymbol{C}_4)\boldsymbol{e}_1^{(k)^4} - (-8\boldsymbol{C}_2^4 + 16\boldsymbol{C}_2^2\boldsymbol{C}_3 - 6\boldsymbol{C}_2\boldsymbol{C}_4 - 3\boldsymbol{C}_3^2)\boldsymbol{e}_1^{(k)^5} + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^6}\right) \qquad (10)$$

$$\mathbf{g}\left(\tfrac{1}{2}\textstyle\sum_{j=1}^{2}\boldsymbol{y}_j^{(k)}\right) \approx \mathbf{g}'(\boldsymbol{r})\left[\boldsymbol{e}_2^{(k)} + \boldsymbol{C}_2\boldsymbol{e}_2^{(k)^2} + \boldsymbol{C}_3\boldsymbol{e}_2^{(k)^3} + \boldsymbol{C}_4\boldsymbol{e}_2^{(k)^4}\right] + \mathbf{O}\left(\boldsymbol{e}_2^{(k)^5}\right) \qquad (11)$$

Now, for $i=3$ we have:

$$\boldsymbol{y}_3^{(k)} - \boldsymbol{r} = \tfrac{1}{2}\textstyle\sum_{j=1}^{2}\boldsymbol{y}_j^{(k)} - 3\left[\mathbf{g}'(\boldsymbol{y}_1^{(k)})\right]^{-1}\mathbf{g}\left(\tfrac{1}{2}\sum_{j=1}^{2}\boldsymbol{y}_j^{(k)}\right) - \boldsymbol{r} = -2\boldsymbol{C}_2\boldsymbol{e}_1^{(k)^2} + 2(5\boldsymbol{C}_2^2 - 2\boldsymbol{C}_3)\boldsymbol{e}_1^{(k)^3} - (-35\boldsymbol{C}_2^3 + 35\boldsymbol{C}_2\boldsymbol{C}_3 - 6\boldsymbol{C}_4)\boldsymbol{e}_1^{(k)^4} + (74\boldsymbol{C}_2^4 - 100\boldsymbol{C}_2^2\boldsymbol{C}_3 + 18\boldsymbol{C}_2\boldsymbol{C}_4 + 12\boldsymbol{C}_3^2)\boldsymbol{e}_1^{(k)^5} + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^6}\right) \qquad (12)$$

$$\underbrace{\tfrac{1}{3}\textstyle\sum_{j=1}^{3}\boldsymbol{y}_j^{(k)} - \boldsymbol{r}}_{\boldsymbol{e}_3^{(k)}} = 2\boldsymbol{C}_2^2\boldsymbol{e}_1^{(k)^3} + (-9\boldsymbol{C}_2^3 + 7\boldsymbol{C}_2\boldsymbol{C}_3)\boldsymbol{e}_1^{(k)^4} + (30\boldsymbol{C}_2^4 - 44\boldsymbol{C}_2^2\boldsymbol{C}_3 + 10\boldsymbol{C}_2\boldsymbol{C}_4 + 6\boldsymbol{C}_3^2)\boldsymbol{e}_1^{(k)^5} + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^6}\right) \qquad (13)$$

Repeating for $i=4$ and $i=5$, one can check that:

$$\boldsymbol{e}_4^{(k)} = 4\boldsymbol{C}_2^3\boldsymbol{e}_1^{(k)^4} + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^5}\right) \qquad (14)$$

$$\boldsymbol{e}_5^{(k)} = 8\boldsymbol{C}_2^4\boldsymbol{e}_1^{(k)^5} + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^6}\right) \qquad (15)$$

Thus, we can deduce for a generic N

$$\boldsymbol{e}_N^{(k)} = \boldsymbol{e}_1^{(k+1)} = 2^{N-2}\boldsymbol{C}_2^{N-1}\boldsymbol{e}_1^{(k)^N} + \mathbf{O}\left(\boldsymbol{e}_1^{(k)^{N+1}}\right) \qquad (16)$$

Therefore, from (16) we can observe that the developed PF solution paradigm (4) converges with order N, and the proof is completed. □

IV. COMPARISON WITH OTHER HIGH ORDER PF SOLUTION METHODS

Other high order PF solution methods have been proposed in [8], [9]. In this section, a comparison of these methodologies with the developed PF solution paradigm is addressed. To do that, we can use the following efficiency index [17]:

$$FEI = p^{1/co} \qquad (17)$$

where, $p \in \mathbb{R}^+$ is the order of convergence and CO stands for the total computational cost of an iteration in terms of flops. In this regard, the following theorem is generally used to estimate the cost of a LU decomposition [16].

Theorem 2. *The number of products and quotients required for solving q linear systems of equations with the same matrix of coefficients, using LU factorization, is:*

$$o(n, q) = \frac{1}{3}n^3 + qn^2 - \frac{1}{3}n \tag{18}$$

It is also suitable to take into account the computational cost of each function evaluation $o(\mathbf{g})$ and Jacobian evaluation $o(\mathbf{g}')$, hence, the total computational cost of an iteration can be estimated as follows:

$$CO = o(\mathbf{g}) + o(\mathbf{g}') + o(n, q) \tag{19}$$

Theorem 3. *For the developed PF solution methods, we have*

$$FEI = N^{1 / \left(\frac{1}{3}n^3 + Nn^2 + \frac{3N-4}{3}n\right)}$$

Proof. The developed PF solution paradigm requires $N-1$ function evaluations along a Jacobian factorization each iteration. In addition, it requires to solve $N-1$ linear systems using the same LU decomposition, consequently: $CO = (N-1)n + n^2 + \frac{1}{3}n^3 + (N-1)n^2 - \frac{1}{3}n = \frac{1}{3}n^3 + Nn^2 + \frac{3N-4}{3}n$ □

Fig. 2 shows the value of (17) for various values of N along with the value of this index for the standard NR and the other high order methods considered in [8], [9]. As can be seen, the Darvishi's 3rd order method and the developed paradigm with $N = 3$ shows the same computational cost. Superiority of the developed paradigm with respect to the other methods is achieved for $N > 3$. Nevertheless, it can be seen that the computational efficiency is not significantly enhanced for $N > 5$. For those reasons, we have limited the analysis to $N = 4$ and $N = 5$. For these values, the developed method achieves fourth and fifth convergence rate, respectively.

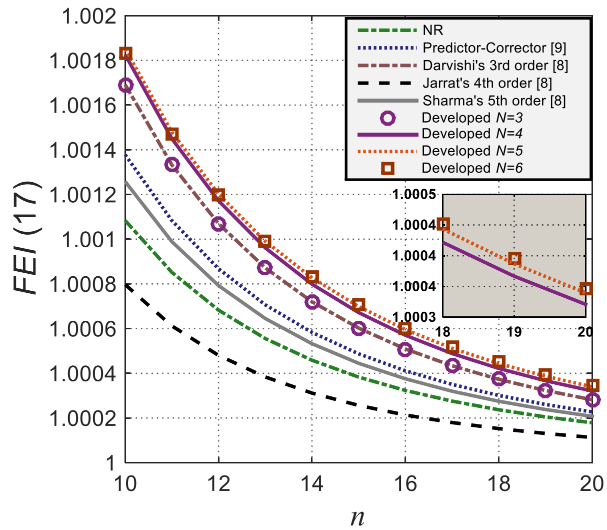

Fig. 2 Efficiency index (17) for various PF solution methods.

V. Numerical Results

In this section, the developed PF solution paradigm is validated using twelve test systems.

A. Studied Systems

For numerical experiments, we have considered various systems available at MATPOWER's database [18]-[21]. Table III reports the studied systems along with their assigned number and main characteristics.

Certainly, System #1 cannot be considered a large realistic system. However, it is a standard IEEE system which is well-known and widely reported. Thus, it has been included in order to fully validate the developed methods.

B. Considered PF methods

In order to validate the developed PF solution paradigm, various high order techniques along with the standard NR have been considered. Table IV reports the studied PF methods along the nomenclature considered.

All considered PF methods have been coded in MATPOWER v7.0 [18] following the guidelines provided in their references. In all cases, a flat start has been taken and $\varepsilon = 10^{-6}$ has been imposed as convergence criterion.

C. Comparison of the number of iterations

Firstly, the considered PF methods have been compared in terms of total number of iterations. In this regard, the highest convergence rate should be reflected in the fastest convergence. Table V reports the total number of iterations employed by the considered PF methods in the studied systems at base case scenario. 4N always converged earlier than NR, PC and 4J. It frequently converged in less iterations than 3D. As expected, 4N occasionally required an extra iteration compared with 5S. On the other hand, 5N always converged in less iterations than NR, PC, and 4J. In addition, it occasionally converged earlier than 3D and 5S. It is worth mentioning that 3D diverged in the Systems #6 and #12.

It is observed that 4J occasionally needed more iterations than 3D, which is intriguing since 4J presents higher convergence rate than 3D. This is due to the strong local convergence of 4J. The System #8 serves to illustrate that. Fig. 3 plots the convergence characteristics of different methods in this system. From this figure, it can be observed that 4J performs similarly to PC at the beginning of the procedure. In other words, 4J badly performs far away from the solution. In this case, 5S also presents local convergence since its convergence properties are deteriorated when it evolves far away from the solution. This explains why 5N occasionally converged earlier than 5S (oppositely, in other scenarios 5S outperforms 5N as it will be discussed later).

TABLE III
STUDIED TEST SYSTEMS

Syst. No.	Name	Buses	Branches	Generators	Load MW	Load MVar	n
#1	*case57*	57	80	7	1250.8	336.4	106
#2	*case_ACTIVSg500*	500	597	90	7750.7	2066.8	943
#3	*case1354pegase*	1354	1991	260	73059.7	13401.4	2447
#4	*case_ACTIVSg2000*	2000	3206	544	67109.2	19014.3	3607
#5	*case2383wp*	2383	2896	327	24558.4	8143.9	4438
#6	*case2736sp*	2736	3504	420	18074.5	5339.5	5237
#7	*case2737sop*	2737	3506	399	11267.2	3953.2	5280
#8	*case2746wop*	2746	3514	514	18962.1	5534.5	5141
#9	*case2746wp*	2746	3514	520	24873	7146.5	5127
#10	*case2869pegase*	2869	4582	510	132437.3	29007.8	5227
#11	*case3120sp*	3120	3693	505	21181.5	8723.2	5991
#12	*case9241pegase*	9241	16049	1.445	312354.1	73581.6	17036

TABLE IV
CONSIDERED PF SOLUTION METHODS

Nom.	Method	Ref.
NR	Standard NR	[2]
PC	Predictor-Corrector	[9]
3D	Darvishi's 3rd order	[8]
4J	Jarrat's 4th order	[8]
5S	Sharma's 5th order	[8]
4N	Developed $N = 4$	Present
5N	Developed $N = 5$	Present

TABLE V
COMPARISON OF THE TOTAL NUMBER OF ITERATIONS AT BASE CASE SCENARIO

No.	NR	PC	3D	4J	5S	4N	5N
#1	4	3	3	3	2	2	2
#2	4	3	3	3	2	2	2
#3	5	4	3	4	2	3	2
#4	5	4	3	4	2	3	2
#5	4	4	3	4	2	2	2
#6	6	5	*	5	3	3	3
#7	5	4	4	4	2	3	2
#8	6	5	4	5	3	3	2
#9	6	5	4	5	3	3	2
#10	5	4	3	4	2	3	2
#11	5	5	4	5	2	3	2
#12	6	5	*	5	3	3	3

* *Diverged*

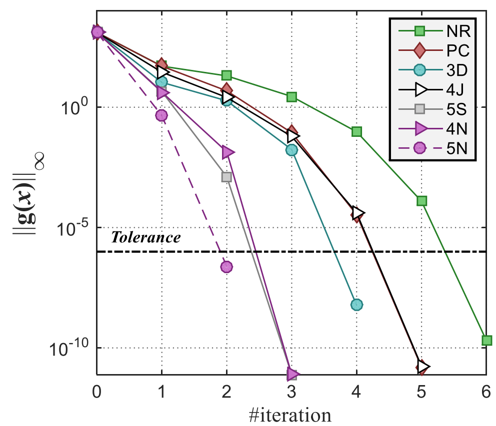

Fig. 3 Convergence characteristics of System #8 (*case2746wop*) at base case scenario.

The performance of tested PF methods with reactive limits has been also analysed. In that sense, the strategy described in [22] has been used for considering the generators' reactive limits. Table VI reports the total number of iterations of the studied methods at this scenario. In this case, similar conclusions as for the base case scenario can be extracted.

Next, let us examine how the loading level affects the convergence features of different PF methods. To do that, let us increase the active and reactive power injected at PQ buses and the active power injected at PV buses by multiplying them by a real factor namely loading level (λ). Fig. 4 shows the iteration counters at different loading levels. As expected, the convergence becomes slow as the loading level grows. In this case, as in the base case scenario, the highest convergence rate typically implies the lowest iteration counter, thus, 5S, 4N and 5N usually converged in less iterations than remainder methods. As expected, 5S and 5N sometimes saved an iteration in comparison with 4N, especially for high loading levels. Finally, 5N eventually outperformed 5S in the Systems #6, #7, #8, #9, #11 and #12 while the later occasionally offered the best performance in the Systems #1, #2 and #5. Nevertheless, these two techniques typically performed similarly since they present same convergence order.

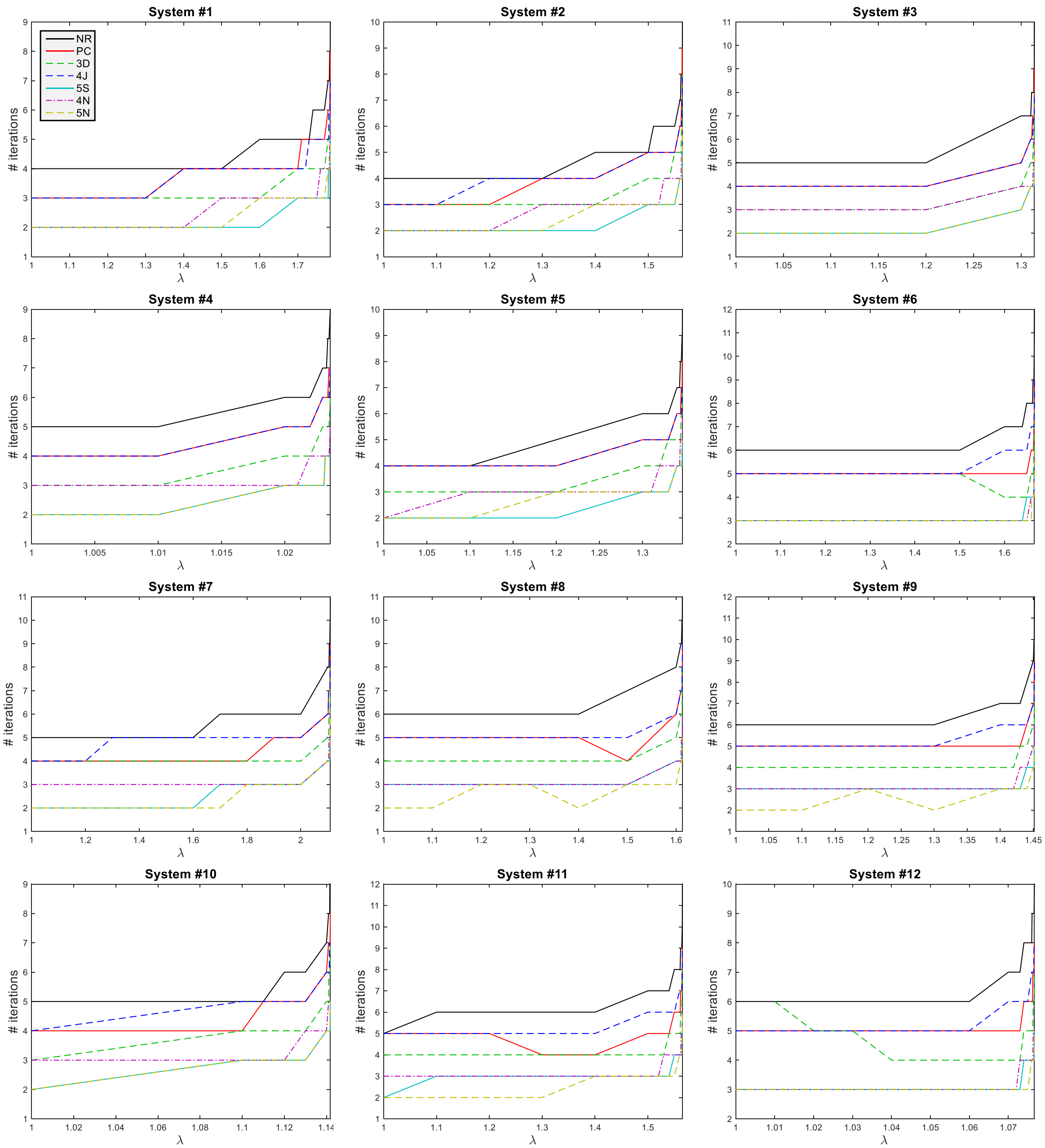

Fig. 4 Total Number of Iterations for different loading levels

Regarding high loading conditions, the so-called limit load case is especially relevant since it supposes the most demand scenario. To represent this case, let us increase λ in steps of 0.0001 pu until all considered methods diverge. Then, the immediately lower loading level is taken, e.g., in System #10, $\lambda = 1.1418$ pu since $\lambda = 1.1419$ pu gives rise the divergence. Loading levels considered at limit load scenario are reported in Appendix B (Table XII). The total number of iterations at limit loading scenario is reported in Table VII. As can be seen, 4N and 5N required less iterations than NR, PC, 3D and 4J in all tested systems. On the other hand, 5S typically converged earlier than 4N.

It is observed that 5S occasionally outperformed 5N with reactive limits enforcement and heavy loading conditions. Let us deeply analyze the System #3 in order to study this behavior. Fig. 5 plots the convergence pattern of both 5S and 5N in this system at limit load scenario. As can be seen, both methods performed with approximately quadratic convergence until fourth iteration. After that, convergence is accelerated (it is observed that 5N achieves fifth convergence rate at fifth iteration). In this case, 5S performed slightly better than 5N after fourth iteration, which allows reaching the required convergence tolerance earlier. This particular behavior is characteristic of local convergence, in other words, convergence features are only achieved in the vicinity of the solution. This pattern is observed in most of iterative methods (also for NR). Reader can also check that convergence behavior is strongly case dependent, for example, at base case scenario 5N occasionally required less iterations than 5S (see Fig. 3). Further analysis of convergence behavior of the developed techniques is out of scope of this work. Interested readers can find a deep analysis about the convergence of the iterative nonlinear solvers in [23] and references therein.

TABLE VI
COMPARISON OF THE TOTAL NUMBER OF ITERATIONS WITH REACTIVE LIMITS

No.	NR	PC	3D	4J	5S	4N	5N
#1	4	3	3	3	2	2	2
#2	9	8	6	8	5	5	5
#3	10	9	7	9	5	6	5
#4	13	12	8	12	7	8	7
#5	13	12	10	12	7	8	7
#6	11	10	*	10	6	6	6
#7	11	10	7	10	5	6	5
#8	16	15	11	15	9	10	8
#9	15	13	10	13	7	8	6
#10	12	10	8	11	5	7	6
#11	13	13	11	13	7	8	7
#12	13	12	*	12	7	7	7

** Diverged*

TABLE VII
COMPARISON OF THE TOTAL NUMBER OF ITERATIONS AT LIMIT LOAD SCENARIO

No.	NR	PC	3D	4J	5S	4N	5N
#1	9	8	7	7	5	6	5
#2	10	9	8	8	5	6	6
#3	11	9	8	8	5	6	6
#4	9	7	6	7	4	5	4
#5	10	8	7	7	5	6	5
#6	12	9	8	9	6	6	6
#7	11	9	8	9	5	6	6
#8	11	9	8	9	5	6	5
#9	12	9	8	9	5	6	5
#10	9	8	7	7	5	5	5
#11	12	9	8	9	6	6	6
#12	10	8	7	8	5	5	5

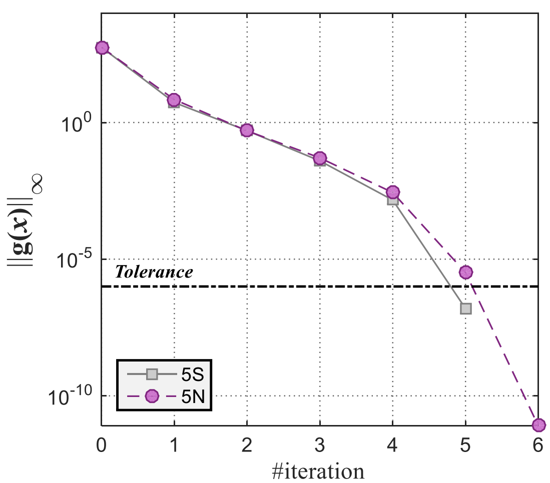

Fig. 5 Convergence characteristics of 5S and 5N in the System #3 (*case1354pegase*) at limit load scenario.

D. *Comparison of the execution time*

The total computational time is one of the most relevant indicators for comparing PF methods, due to its importance for online industry tools. Nevertheless, it can be difficult to measure due to it strongly depends on the coding language, machine used, parallel computational activities, etc. In this regard, all simulations have been run under Windows 10 on a 3.4 GHz Intel Core i7-8750H CPU 2.2 GHz personal laptop (16.00 GB RAM). In addition, the reported results have been obtained as the mean value of 5000 simulations, in order to minimize the influence of other computational activities.

Table VIII reports the execution time of the different considered PF methods at base case scenario. Typically, 4N outperformed NR, PC, 3D, 4J and 5S. Only in those infrequent cases where 3D and 4N employed the same number of iterations (Systems #3, #4 and #10), the former was more efficient. 4N was also more efficient than 5N in the Systems #1, #2, #5 and #6 while 5N was the winner in the remainder systems.

Comparison of the execution time in case of considering the reactive limits and at limit load scenario are reported in Tables IX and X, respectively. With reactive limits, 4N was the most efficient methodology in the Systems #1, #2, #6 and #12, while 5N was the winner in the remainder Systems. As at base case scenario, 3D diverged in the Systems #6 and #12. With limit load conditions, 4N was the most efficient method in the Systems #2, #3, #6, #7, #10, #11 and #12, while the developed 5N outperformed the other considered methods in the remainder systems.

Fig. 6 shows the computation time of an iteration of different PF methods at limit load scenario. Clearly, NR, PC, 3D, 4N and 5N present the most efficient behaviour since they need a LU decomposition each iteration. Oppositely, an iteration of 4J is much heavier due to it needs to solve a matrix system. In the case of 5S, it is computationally more expensive due to it requires two matrix factorizations each iteration. From Fig. 6, it is also observed that an iteration of 4N is heavier than that of NR, PC and 3D. This is due to 4N requires to solve three linear systems along with three function evaluations, while NR only needs to solve a linear system and a function evaluation. On the other hand, PC and 3D require to solve two linear systems. PC also needs a function evaluation while 3D computes two function evaluations. For the same reasons, an iteration of 5N is heavier than that of 4N, since the former requires to solve four linear systems and four function evaluations.

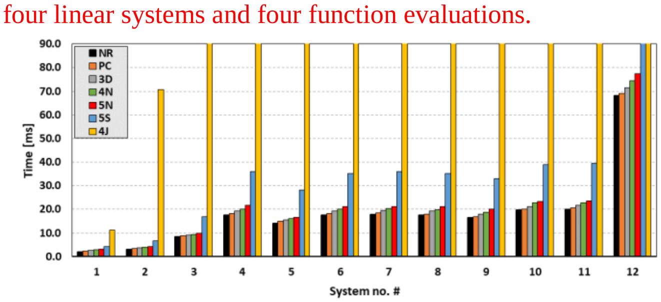

Fig. 6 Comparison of the iteration time at limit load scenario.

Due to NR is frequently considered as the standard PF solution method, it is interesting observe the computational saving offered by the developed PF methods with respect this technique. In this regard, Fig. 7 shows the speedup ratio of the developed PF methods compared with NR. As can be noted, the developed methods are able to notably outperform NR in all studied systems. Superiority is typically more remarkable with 5N, in large scale systems and at limit load scenario. It is worth mentioning that the developed methods are able to improve more than twice the computational time offered by NR in some cases.

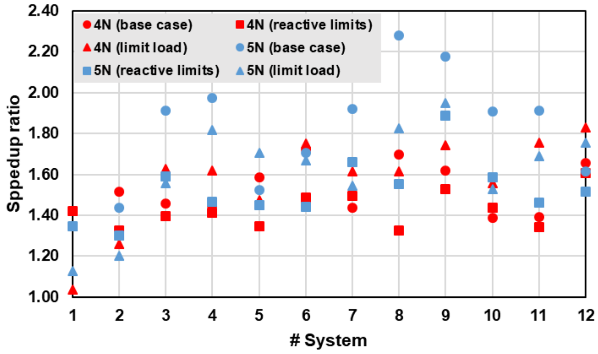

Fig. 7 Speedup ratio of the developed PF methods with respect to NR

E. Influence of the R/X ratio

A system's R/X ratio may affect the performance of a PF method [9]. Normally, more iterations or instability are experienced when the overall R/X ratio is increased. Table XI reports the total number of iterations required by the developed 4N and 5N for solving the studied systems at different R/X ratios. A comparison with the base case scenario (Table V) is also reported. As can be checked, the developed PF methods are barely affected by the R/X ratio, since they only required an extra iteration in a few cases compared with the base case scenario.

VI. Conclusions and Future Works

In this paper, a novel family of multistep PF methods have been developed. The introduced solution paradigm is able to reach N^{th} order of convergence, where N is the total number of steps considered. In addition, the introduced paradigm can be considered efficient as only one LU factorization is required each iteration.

The developed solution paradigm has been compared with the standard NR and the other high order PF methods considered in the literature, using a well-known efficiency index. Superiority of the developed paradigm for $N \geq 4$ with have been proved.

The developed solution paradigm has been validated for $N = 4$ (4N) and $N = 5$ (5N) using twelve test systems under different demand scenarios. Results exhibited by the developed 4N and 5N were very promising and competitive.

Future works should study the applicability of the introduced solution paradigm in other related problems such as Continuation Power-Flow [24] or contingency analysis [25].

TABLE VIII
Comparison of the Execution Time [ms] at Base case scenario

No.	NR	PC	3D	4J	5S	4N	5N
#1	7.8	6.9	7.1	33.2	8.6	5.5	5.8
#2	14.1	12.2	13.1	229.1	14.0	9.3	9.8
#3	43.2	35.9	29.2	3069.2	35.3	29.6	22.6
#4	84.5	71.5	57.3	7635.0	71.0	58.4	42.8
#5	60.3	61.9	51.5	10432.0	61.2	38.0	39.6
#6	109.2	93.3	--	17871.0	110.1	63.4	64.1
#7	92.8	77.0	83.4	14639.0	76.8	64.5	48.3
#8	108.0	93.0	80.6	17309.0	109.0	63.7	47.4
#9	98.7	85.4	77.6	17189.0	102.1	60.9	45.3
#10	99.2	82.2	69.0	20385.0	83.0	71.5	52.0
#11	100.6	103.4	93.6	22779.0	84.9	72.4	52.6
#12	400.2	341.5	--	351530.0	416.0	241.4	247.6

TABLE IX
Comparison of the Execution Time [ms] with Reactive Limits

No.	NR	PC	3D	4J	5S	4N	5N
#1	7.8	6.9	7.1	33.2	8.6	5.5	5.8
#2	33.5	31.9	26.4	628.7	34.2	25.3	25.8
#3	86.9	81.4	68.2	6822.6	86.6	62.2	54.6
#4	228.0	221.6	158.6	24492.0	251.3	161.4	155.5
#5	207.5	198.4	176.5	33390.0	223.5	154.3	143.0
#6	205.6	194.2	--	36470.0	228.6	138.3	142.8
#7	208.2	195.5	151.0	37273.0	192.7	139.3	125.3
#8	294.9	286.1	234.5	55062.0	330.5	222.3	190.0
#9	278.5	249.5	209.5	46215.0	261.3	182.5	147.5
#10	247.8	219.3	183.8	55657.0	209.5	172.4	156.1
#11	273.6	283.0	258.3	60904.0	296.0	203.8	187.0
#12	901.0	841.9	--	866720.0	963.1	560.7	593.8

TABLE X
COMPARISON OF THE EXECUTION TIME [ms] AT LIMIT LOAD SCENARIO

No.	NR	PC	3D	4J	5S	4N	5N
#1	18.0	18.4	18.2	77.7	21.5	17.4	16.0
#2	31.0	29.7	30.4	566.4	33.2	24.6	25.8
#3	91.0	78.3	71.5	6000.0	84.4	56.0	58.5
#4	157.1	128.0	115.0	13264.0	143.5	97.0	86.4
#5	141.5	116.8	109.2	17812.0	140.8	96.0	83.0
#6	212.0	164.7	154.8	31024.0	211.2	121.0	127.1
#7	196.7	165.3	156.4	32229.0	180.3	121.8	127.5
#8	193.6	162.3	154.1	30242.0	176.5	119.8	106.0
#9	196.7	152.2	144.4	30136.0	165.6	113.0	100.9
#10	177.6	160.7	148.7	35546.0	194.2	114.2	116.2
#11	239.8	185.2	173.5	40635.0	236.3	136.5	142.0
#12	680.0	552.6	487.0	574000.0	661.4	371.7	387.0

TABLE XI
ITERATION NUMBERS WITH HIGH R/X RATIOS

No.	Resistances increased by 20%		Impedances reduced by 20%	
	4N	5N	4N	5N
#1	2 (+0)	2 (+0)	2 (+0)	2 (+0)
#2	2 (+0)	2 (+0)	2 (+0)	2 (+0)
#3	3 (+0)	2 (+0)	3 (+0)	2 (+0)
#4	3 (+0)	2 (+0)	3 (+0)	2 (+0)
#5	2 (+0)	2 (+0)	2 (+0)	2 (+0)
#6	**4 (+1)**	3 (+0)	**4 (+1)**	3 (+0)
#7	3 (+0)	2 (+0)	3 (+0)	2 (+0)
#8	3 (+0)	**3 (+1)**	3 (+0)	**3 (+1)**
#9	3 (+0)	**3 (+1)**	3 (+0)	**3 (+1)**
#10	3 (+0)	2 (+0)	3 (+0)	2 (+0)
#11	3 (+0)	2 (+0)	3 (+0)	2 (+0)
#12	**4 (+1)**	3 (+0)	3 (+0)	3 (+0)

In parenthesis is reported a comparison with respect the base case scenario (Table IV)

APPENDIX A: PF FORMULATION IN POLAR COORDINATES

The polar form of PF equations is typically preferred since size of the system is reduced relative to the rectangular form [14]. In this case, voltage angles at PV and PQ buses, along voltage magnitudes at PQ buses form the state vector as:

$$\boldsymbol{x} = [\boldsymbol{\delta_{PV}}|\boldsymbol{\delta_{PQ}}|\boldsymbol{V_{PQ}}]^T \tag{20}$$

where, $\boldsymbol{\delta_{PV}} \in \mathbb{R}^{n_g}$ is the vector of voltage angles at PV buses, $\boldsymbol{\delta_{PQ}} \in \mathbb{R}^{n_g}$ is the vector of voltage angles at PQ buses and $\boldsymbol{V_{PQ}} \in \mathbb{R}^{n_l}$ is the vector of voltage angles at PQ buses. On the other hand, $n_g \in \mathbb{N}$ and $n_l \in \mathbb{N}$ represent the total number of PV and PQ buses, respectively.

The nonlinear relationships between the power nodal injections and nodal voltages are given by [26]:

$$\mathbf{g}(\boldsymbol{x}) = \begin{cases} \mathbf{g}_{\boldsymbol{P}}, \text{For all buses} \\ \mathbf{g}_{\boldsymbol{Q}}, \text{For PV buses} \end{cases} \tag{21}$$

$$g_{P_i} = P_i^{sch} - \sum_{j=1}^{n_l+n_g} |V_i|\,|V_j||Y_{ij}|\cos(\theta_{ij} - \delta_i + \delta_j) \tag{22}$$

$$g_{Q_i} = Q_i^{sch} - \sum_{j=1}^{n_l} |V_i|\,|V_j||Y_{ij}|\sin(\theta_{ij} - \delta_i + \delta_j) \tag{23}$$

where, $P_i^{sch} \in \mathbb{R}$ 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. $n_l \in \mathbb{N}$ is the total number of PQ buses and $n_g \in \mathbb{N}$ is the total number of PV buses.

APPENDIX B: LIMIT LOADING LEVELS

In this paper, the studied systems have been studied at base case and limit loading conditions. Furthermore, the loading level of the system has been modified using the following expressions:

$$P_i^{sch} = \lambda P_i^{sch} \quad \text{for PQ and PV buses} \tag{24}$$

$$Q_i^{sch} = \lambda Q_i^{sch} \quad \text{for PQ buses} \tag{25}$$

where $\lambda \in \mathbb{R}^+$ is the loading level. Table XII reports the loading levels considered in simulations.

TABLE XII
LOADING LEVELS CONSIDERED AT LIMIT LOAD SCENARIO

Syst. No.	$\lambda[pu]$	Syst. No.	$\lambda[pu]$
#1	1.7855	#7	2.1097
#2	1.5646	#8	1.6129
#3	1.3139	#9	1.4516
#4	1.0236	#10	1.1418
#5	1.3463	#11	1.5653
#6	1.6671	#12	1.0767

REFERENCES

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

[2] W. F. Tinney and C. E. Hart, “Power flow solution by Newton’s method,” *IEEE Trans. Power App. Syst.*, vol. PAS-86, no. 11, pp. 1449–1460, Nov. 1967.

[3] W. F. Tinney and J. W. Walker, “Direct solutions of sparse network equations by optimally ordered triangular factorization,” *Proc. IEEE*, vol. 55, no. 11, pp. 1801–1809, Nov. 1967.

[4] B. Sttot, “Review of load flow calculation methods,” *Proc. IEEE*, vol. 62, no. 7, pp. 916–929, Jul. 1974.

[5] *Electric Energy Systems: Analysis and Operation*, A. Gómez-Expósito, A. Conejo, and C. Cañizares, Eds. Boca Raton, FL, USA: CRC, 2008.

[6] M. Tostado, S. Kamel and F. Jurado, “Several robust and efficient load flow techniques based on combined approach for ill-conditioned power systems,” *Int. J. Electr. Power Energy Syst.*, vol. 110, pp. 349-356, Sept. 2019.

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

[8] 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. Transm. Distrib.*, vol. 10, no. 12, pp. 3017-3022, Sep. 2016.

[9] M. Tostado, S. Kamel and F. Jurado, “Developed Newton-Raphson based Predictor-Corrector load flow approach with high

convergence rate," *Int. J. Electr. Power Energy Syst.*, vol. 105, pp. 785-792, Feb. 2019.

[10] M.T. Darvishi and A. Barati, "'A third-order Newton-type method to solve systems of nonlinear equations," *Appl. Math. Comput.*, vol. 187, no. 2, pp. 630–635, April 2007.

[11] M.Q. Khirallah and M.A. Hafiz, "Solving system of non-linear equations using family of Jarratt methods," *Int. J. Differential Equations Applications*, vol. 12, no. 2, pp. 69-83, Jan. 2013.

[12] J.R. Sharma and P. Gupta, "An efficient fifth order method for solving systems of nonlinear equations," *Comput. Math. Appl.*, vol. 67, no. 3, pp. 591-601, Feb. 2014.

[13] T.J. McDougall and S.J. Wotherspoon, "A simple modification of Newton's method to achieve convergence of order 1+√2," *Appl. Math. Letters*, vol. 29, pp. 20-25, Mar. 2014.

[14] J. E. Tate and T. J. Overbye, "A comparison of the optimal multiplier in polar and rectangular coordinates," *IEEE Trans. Power Syst.*, vol. 20, no. 4, pp. 1667–1674, Nov. 2005.

[15] J.F. Traub, *Iterative Methods for the Solution of Equations*. New York: Chelsea, 1982.

[16] A. Cordero, J.L. Hueso, E. Martínez and J.R. Torregrosa, "A modified Newton-Jarratt's composition," *Numer. Algor*., vol. 5, no. 1, pp. 87-99, Sept. 2010.

[17] 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. Computer Math*., vol. 92, no. 9, pp. 1921-1934, Aug. 2015.

[18] 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.

[19] 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.

[20] C. Josz, S. Fliscounakis, J. Maeght and P. Panciatici, "AC Power Flow Data in Matpower and QCQP Format: iTesla, RTE Snapshots, and PEGASE," 2016, arXiv:1603.01533

[21] 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.

[22] M. Tostado-Véliz, S. Kamel and F. Jurado, "Promising framework based on multistep continuous Newton scheme for developing robust PF methods," *IET Gener. Transm. Distrib.*, vol. 14, no. 2, pp. 265-274, Jan. 2020.

[23] J.M. Ortega and W.C. Rheinboldt, *Iterative Solution of Nonlinear Equations in Several Variables*. Philadelphia: Soc. Ind. Appl. Math, 2000.

[24] 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.

[25] Z. Chen, H.D. Chiang, Y. Tang and N. Zhou, "An online line switching methodology with look-ahead capability to alleviate power system overloads based on a three-stage strategy," *Int. J. Electr. Power. Energy Syst.*, vol. 115, Feb. 2020. doi: 10.1016/j.ijepes.2019.105500

[26] H. Saadat H, *Power System Analysis*, 3rd ed. North York, Toronto: PSA Publishing, 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 PhD degree from University of Jaen, Spain (Main) and Aalborg University, Denmark (Host). From Nov. 2014, he joined the Electrical Engineering Department, Faculty of Engineering, Aswan University, Egypt as an Assistant Professor. He is currently a Postdoctoral Research Fellow in State Key Laboratory of Power Transmission Equipment and System Security and New Technology, School of Electrical Engineering, Chongqing University, Chongqing, China. His research activities include power system modeling, analysis and optimization, renewable energy and smart grid technologies.

Francisco Jurado (M'00–SM'06) was born in Linares, Jaén, Spain. He received the M.Sc. and Dr. Ing. degrees from the National University of Distance Education, Madrid, Spain, in 1995 and 1999, respectively. Since 1985, he has been a Professor with the Department of Electrical Engineering, University of Jaén, Jaén. His current research interests include power systems, modeling, and renewable energy.