A Powerful Power-Flow Method based on Composite Newton-Cotes formula for Ill-conditioned Power Systems

Marcos Tostado-Véliz[1], Salah Kamel[2,3], Francisco Jurado [1,*]

[1]Department of Electrical Engineering, University of Jaén, 23700 EPS Linares, Jaén, Spain

[2]Department of Electrical Engineering, Faculty of Engineering, Aswan University, 81542 Aswan, Egypt

[3]State Key Laboratory of Power Transmission Equipment and System Security and New Technology, Chongqing University, Chongqing, China

***Abstract* –** This paper proposes a novel Power-Flow method based on Composite Newton-Cotes formula. The proposed method aims to be robust and efficient enough to properly manage ill-conditioned power systems. Self-adapted mechanisms for tuning the parameters of proposed PF method are provided. The proposed method is tested on several ill-conditioned systems ranging from 3012 to 27318 buses at loadbase and heavy loading conditions. Moreover, its ability to handle the generators' reactive limits is also tested. The obtained results prove that the proposed method is remarkably robust and more efficient than other well-known Power-Flow techniques.

Keywords: Power-Flow, Ill-conditioned power systems, Composite Newton-Cotes, Generator's reactive limits.

---

*Corresponding author, Tel.: +34 953 648518; Fax: +34 953 648586.
E-mail addresses: fjurado@ujaen.es (F. Jurado), mtostado@ujaen.es (M. Tostado), skamel@aswu.edu.eg (S. Kamel)

1. Introduction

The Power-Flow (PF), is one of the most important problems in power systems analysis. From a mathematical point of view, solution of the PF problem is obtained by solving a set of nonlinear equations, which relates the nodal voltages with the injected nodal powers. Due to the nonlinearity of PF equations, they cannot be directly solved. Consequently, a numerical iterative technique must be used to solve these equations.

The solution of PF problem using the standard Newton-Raphson's technique was firstly proposed in [1]. It constitutes the most widely used methodology for solving the PF problem. NR has quadratic convergence characteristics, however, Jacobian matrix which is considered the largest computational burden of this methodology, has to be updated and factorized each iteration. In order to reduce this computational burden, decoupled techniques have been proposed [2-4]. These methods are quite efficient, however, they tend to use much more iterations since the information at first iteration is repeatedly used and, consequently, convergence becomes linear or near-linear.

PF problem has been addressed in serval references by developing other solvers and formulations. For example, several PF alternative formulations [5-8], high order approaches [9, 10] and linear versions of the PF problem [11, 12], have been proposed over decades.

Most of cases can be efficiently solved using some standard technique like NR method, and a flat start. However, there are idiosyncratic cases which the NR fails to converge if the initial guess is not close enough to the solution. They are called ill-conditioned systems which have been addressed in multiple works by developing several robust techniques.

In [13], Iwamoto and Tamura developed a robust PF method based on a modification of standard NR. Each iteration, an optimal multiplier is calculated. It is used to truncate the increment vector in order to avoid the divergence. This method extraordinarily robust, however, it might be totally inefficient due to many iterations are often required to reach the

solution. Anyway, it is considered as benchmark in multiple references. This philosophy has been profusely applied to the PF problem in other works [14-16].

The Continuous Newton's method [17], establishes a formal analogy between a system of nonlinear equations and a set of autonomous differential equations. This idea has been exploited in the PF problem by Milano [18]. As result, various robust and efficient methodologies have been proposed. They have been tested on a large-scale ill-conditioned system showing good performance. In [19], this issue has been addressed by proposing a combined approach based on the Broyden's method.

A family of robust PF techniques that uses the Levenberg iteration instead of the traditional Newton scheme has been recently developed in various papers. In [20], Levenberg-Marquardt method has been adapted for solving the PF problem. However, it was observed that the reliability of this method strongly depends on the value of so-called damping factor, moreover, the convergence of this method is very slow. In order to overcome these drawbacks, several high order schemes have been proposed in [21, 22].

Several efficient techniques based on numerical methods have been recently developed. In [23], the Adams-Bashforth's formulas have been extended to the PF problem by developing three efficient and robust solvers. In [24], the Bulirsch-Stoer method has been adapted for solving systems of nonlinear equations and its superiority with respect to other robust techniques has been proved. Recently, a combined approach based on Newton's method, numerical integration techniques and homotopy's schemes has been proposed in [25]. These approaches have shown very good performance in several ill-conditioned systems from 118- to 13569-buses.

In this paper, ill-conditioned systems are tackled. Recently, the authors have proposed several techniques [23-25]. The ability of these techniques to efficiently solve the PF problem of ill-conditioned systems, especially large and very-large scale systems, has been

demonstrated based on numerical results. However, the need for developing novel robust methodologies are still exist in order to avoid any potential failure or bad performance for these techniques. Hence, we develop a novel PF method based on Newton-Cotes formulas. Superiority of developed method over the other PF techniques under different scenarios is demonstrated. However, the topic discussed and contribution of the work could be summarized as follows:

- A novel PF method based on the Newton-Cotes formulas has been proposed to solve the ill-conditioned power systems;
- A deep discussion and self-adapted mechanisms for tuning parameters of proposed PF method have been provided;
- Proposed PF method has been validated using 6 ill-conditioned systems ranging from 3012, to 27318-buses at loadbase and heavy loading conditions;
- Proposed method has been compared with other standard and robust PF techniques;
- Ability of the proposed method for managing generators' reactive limits has been validated;
- A comparison between the proposed and NR methods regarding to the safe initialization has been provided and results have been discussed.

Remainder of this paper is organized as follows. Newton-Cotes formulas are described in Section 2. Section 3 outlines the PF problem. The proposed PF method is described in Section 4. In Section 5, several numerical experiments are carried out and the main results are presented and discussed. Finally, the main conclusions are presented in Section 6.

2. Newton-Cotes formulas

Newton-Cotes is a family of popular numerical integration methods. They approximate the value of the integral by evaluating the function in several equidistant points.

Firstly, let us consider the integral of the following continuous function $\mathrm{f}(x)$ over the main interval $[a, b]$:

$$\int_a^b \mathrm{f}(x)dx \tag{1}$$

where $\mathrm{f}: \mathbb{R} \longmapsto \mathbb{R}$ and $x \in \mathbb{R}$. Value of (1) can be approximated by evaluating f at two points, it means, by a first-degree polynomial as follows:

$$\int_a^b \mathrm{f}(x)dx \approx \frac{b-a}{2}(\mathrm{f}(a) + \mathrm{f}(b)) \tag{2}$$

While (2) corresponds with the well-known Trapezoidal rule [26], value of (1) can be conveniently approximated using a third (Simpson's rules) or fourth (Boole's Rule) degree polynomials.

Newton-Cotes formulas are limited to the number of functions evaluations considered. Alternatively, accuracy of these methods can be improved by reducing the size of the interval, it means, dividing $[a, b]$ in several smaller subintervals. This idea is sketched in Fig. 1, where it is easily checked that accuracy in the calculation is improved as more subintervals are considered. However, computational efficient of the method is affected since more calculations are required.

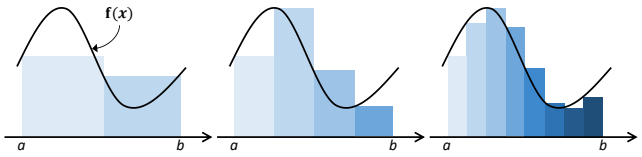

Fig 1 Sketch of Composite Integration techniques using 2 (left), 4 (middle) and 8 (right) subintervals

These methods are known as Composite Newton-Cotes formulas [27]. If we extend the Trapezoidal Rule (2) to its composite counterpart, we obtain the Composite Trapezoidal Rule, which calculates the value of (1) as follows:

$$\int_a^b \mathrm{f}(x)dx \approx \frac{b-a}{n_{si}}\left(\frac{\mathrm{f}(a)}{2} + \sum_{i=1}^{n_{si}-1} \mathrm{f}\left(a + i\frac{b-a}{n_{si}}\right) + \frac{\mathrm{f}(b)}{2}\right) \tag{3}$$

where, $n_{si} \in \mathbb{N}$ is the number of subintervals. In the same way, other Newton-Cotes formulas can be derived to their composite rule.

2.1.- Motivation on the usage of Newton-Cotes formulas

Motivation on the usage of Newton-Cotes formulas for developing a novel PF method is mainly due to the following two powerful ideas:

- The Continuous Newton's method [17], suggests that any numerical method may be conveniently adapted for solving nonlinear and systems of nonlinear equations. This is a powerful idea which allows to develop a huge variety of numerical nonlinear techniques. In this work, we explore the usage of the family of Newton-Cotes formulas in the PF analysis in an efficient way. Thus, a powerful nonlinear method and its application for solving the PF problem are comprehensively presented.
- Many Newton-like methods are basically founded to solve a nonlinear equation by integrating the area under the function evaluated. For the standard Newton's, this area is approximated by a rectangle. However, other more accuracy approximations can be used. In the literature, other integrations approaches have been already used like triangles [28] or Gauss-quadrature formulas [29]. Consequently, the Newton-Cotes formulas can be used in the same way and more accuracy with respect the standard Newton's method is expected.

3. Power-Flow problem

The PF problem in polar coordinates is defined as a set of nonlinear smooth equations as follows:

$$\mathbf{g}(\boldsymbol{x}) = \mathbf{0} \tag{4}$$

where; $\mathbf{g}$: $\mathbb{R}^n \longmapsto \mathbb{R}^n$ are the active and reactive power mismatches, and $\boldsymbol{x} \in \mathbb{R}^n$ is the vector of unknowns, which are respectively defined as follows:

$$\Delta P_i = P_i^{sch} - \sum_{j=1}^{n} |V_i| \left|V_j\right| \left|Y_{ij}\right| \cos\left(\theta_{ij} - \delta_i + \delta_j\right) \tag{5}$$

$$\Delta Q_i = Q_i^{sch} - \sum_{j=1}^{n_l} |V_i| \left|V_j\right| \left|Y_{ij}\right| \sin\left(\theta_{ij} - \delta_i + \delta_j\right) \tag{6}$$

$$\boldsymbol{x} = [\boldsymbol{\delta_{PV}}, \quad \boldsymbol{\delta_{PQ}}, \quad \boldsymbol{V_{PQ}}]^T \tag{7}$$

where; P_i^{sch} and Q_i^{sch} are the scheduled active and reactive power at bus i respectively, $V_i \angle \delta_i$ is the complex voltage at bus i, $Y_{ij} \angle \theta_{ij}$ is the ij^{th} element of admittance matrix, $\boldsymbol{\delta_{PV}} \in \mathbb{R}^{n_g}$ and $\boldsymbol{\delta_{PQ}} \in \mathbb{R}^{n_l}$ are the voltage angle vector of PV and PQ buses, respectively, $\boldsymbol{V_{PQ}} \in \mathbb{R}^{n_l}$ is the voltage magnitude vector of PQ buses, n_l is the total number of PQ buses and n_g is the total number of PV buses ($n = 2n_l + n_g$).

Practically, it is said that the solution of PF problem is reached when the error vector is small enough. This criterion can be expressed as follows:

$$\|\mathbf{g}(\boldsymbol{x})\|_\infty \leq \varepsilon \tag{8}$$

where, $\varepsilon \in \mathbb{R}^+$ is the so-called converge tolerance, which must be fixed small enough (typically smaller than 10^{-3}). Frequently, PF algorithm also stops if the number of iterations surpassed a predefined value $k^{max} \in \mathbb{N}$. In this case, one cannot claim successful of the method and it is considered failed.

However, the convergence of a numerical technique cannot be always ensured. It mainly depends on the initial guess considered to initialize the method. Frequently, $\boldsymbol{x}^{(0)}$ should be chosen close to the solution of (4) to ensure the convergence in a relatively few iterations number. Anyway, the convergence is guaranteed if $\boldsymbol{x}^{(0)}$ lies inside of the so-called Region of Attraction [18, 30].

3.1.- Ill-conditioned systems

In some references, the ill-conditioning in PF equations is strongly related with the condition number of Jacobian matrix. The condition number can be generally calculated as follows [31]:

$$\vartheta = \|\boldsymbol{J}(\boldsymbol{x})\|_F \times \|[\boldsymbol{J}(\boldsymbol{x})]^{-1}\|_F \tag{9}$$

where, $\boldsymbol{J}(\boldsymbol{x}) = \nabla_{\boldsymbol{x}}^T \mathbf{g}(\boldsymbol{x}) \in \mathbb{R}^{n\times n}$ is the Jacobian matrix and $\|\cdot\|_F$ denotes the Frobenius norm. Nevertheless, a formal definition of ill-conditioned systems is provided in [18]. Broadly, a power system can be classified as ill-conditioned if, despite that solution of the PF does exist, it is not reachable using standard techniques and the flat start. In this paper, this definition is preferred.

4. Proposed method for solving the PF problem

In this paper, we develop a novel method for solving the PF problem of ill-conditioned power systems. This method is inspired on the Composite Newton-Cotes formulas outlined in Section 2. Proposed PF method has been called Composite Newton-Cotes Power-Flow method (CNC-PF). In subsequent paragraphs, the CNC-PF is developed for solving the PF equations in polar coordinates (i.e. (5) and (6)). Nevertheless, it is worth to mention that it is applicable to any system of nonlinear equations in the same way here explained.

Firstly, let us begin from a determined value of (7) at k^{th} iteration, i.e. $\boldsymbol{x}_a^{(k)}$. Let us previously define:

$$\boldsymbol{\Omega}(\boldsymbol{y},\boldsymbol{z}) = -[\boldsymbol{J}(\boldsymbol{y})]^{-1}\mathbf{g}(\boldsymbol{z}) \tag{10}$$

where, $\boldsymbol{y} \in \mathbb{R}^n$ and $\boldsymbol{z} \in \mathbb{R}^n$ are arbitrary vectors. The limits of main interval can be now computed using (11).

$$\boldsymbol{x}_b^{(k)} = \boldsymbol{x}_a^{(k)} + h^{(k)}\boldsymbol{\Omega}\left(\boldsymbol{x}_a^{(k)}, \boldsymbol{x}_a^{(k)}\right) \tag{11}$$

where, $h \in \mathbb{R}^+$ is the step size, which is calculated as follows:

$$h^{(k)} = \min\begin{Bmatrix} \max\begin{Bmatrix} h^{max} \\ h^{min} \end{Bmatrix} / \left\| \boldsymbol{\Omega}\left(\boldsymbol{x}_a^{(k)}, \boldsymbol{x}_a^{(k)}\right) \right\|_\infty \end{Bmatrix} \quad (12)$$

where, $h^{min} \in \mathbb{R}^+$ and $h^{max} \in \mathbb{R}^+$ are the minimum and maximum step sizes, respectively. Now, the main interval $[\boldsymbol{x}_a, \boldsymbol{x}_b]$ is splitted in $m-1$ intermediate points as follows:

$$\boldsymbol{x}_i^{(k)} = \boldsymbol{x}_a^{(k)} + \frac{h^{(k)}}{i}\boldsymbol{\Omega}\left(\boldsymbol{x}_a^{(k)}, \boldsymbol{x}_a^{(k)}\right), \text{for } i \in [2,3,\dots,m^{(k)}] \quad (13)$$

One should note that Jacobian matrix is only calculated at $\boldsymbol{x}_a$, therefore, it only needs to be factorized once each iteration. Finally, the PF state vector is updated as follows:

$$\boldsymbol{\phi}^{(k)} = \psi^{(k)}\boldsymbol{\Omega}\left(\boldsymbol{x}_a^{(k)}, \boldsymbol{x}_a^{(k)}\right) + \rho^{(k)}\sum_{i=2}^{m^{(k)}}\boldsymbol{\Omega}\left(\boldsymbol{x}_a^{(k)}, \boldsymbol{x}_i^{(k)}\right) + \psi^{(k)}\boldsymbol{\Omega}\left(\boldsymbol{x}_a^{(k)}, \boldsymbol{x}_b^{(k)}\right) \quad (14)$$

$$\boldsymbol{x}_a^{(k+1)} = \boldsymbol{x}_a^{(k)} + \frac{h^{(k)}}{\beta^{(k)}}\boldsymbol{\phi}^{(k)} \quad (15)$$

where, $\psi \in \mathbb{R}^+$ and $\rho \in \mathbb{R}^+$ are the extreme and intermediate slope coefficients, respectively and $\beta \in \mathbb{R}^+$ is the damping factor. The performance of proposed method is strongly affected by these parameters, in order to simplify the procedure, we have developed a self-tuning scheme. Thus, different parameters are updated each iteration as follows:

$$\rho^{(k)} = \frac{1}{\left\|\boldsymbol{x}_a^{(k)} - \boldsymbol{x}_b^{(k)}\right\|_\infty^{\mu}} \quad (16)$$

$$\gamma^{(k)} = \frac{m^{(k)}}{\rho^{(k)}} \quad (17)$$

$$\psi^{(k)} = \min\left[\gamma^{(k)}, \psi^{max}\right] \quad (18)$$

$$m^{(k)} = \max\begin{Bmatrix} \min\begin{Bmatrix} \text{round}(\gamma^{(k)}) \\ m^{max} \end{Bmatrix} \\ m^{min} \end{Bmatrix} \quad (19)$$

$$\beta^{(k)} = \max\left[\gamma^{(k)}, \beta^{min}\right] \quad (20)$$

where; $\mu \in \mathbb{R}$ and $m^{min} \in \mathbb{N} \geq 2$, $m^{max} \in \mathbb{N}$, $\beta^{min} \in \mathbb{R}$ and $\psi^{max} \in \mathbb{R}$ define the minimum and maximum values of different parameters. On the other hand, $\text{round}(\cdot)$ is a function which rounds to the nearest integer.

Certainly, tuning parameters is always a difficult task of any algorithm, since they are frequently based on heuristic criteria and personal experiences. However, these parameters can be tuned based on basic ideas and logic criteria. In order to facilitate the applicability of proposed method, the criteria used for tuning all parameters are summarized in the following points:

- **Step size ($\boldsymbol{h}$):** the proposed self-adapted mechanism for tunning the step-size only reduces it when the convergence is at risk. This issue can be intuitively measured by the size of $\boldsymbol{\Omega}(\boldsymbol{x}_a, \boldsymbol{x}_a)$, that is, the size of the Newton's increment vector. The idea behind this mechanism is as; if the size of Newton's increment vector is too large, the step size should be reduced in order to compensate this long step and reduce the risk of divergence. Oppositely, if the increment vector is reasonably short, the step size can be increased with the aim to accelerate the convergence. Keeping this in mind and always finding an equilibrium between efficiency and robustness, bounds of the step-size have to be fixed. On the one hand, the minimum step-size can be tuned as small as necessary, since it would be only used for very complicated problems. In this kind of cases, the efficiency should be sacrificed in order to ensure the convergence, this is achieved using a very small step-size. On the other hand, regarding the maximum step-size, it may be tuned slightly higher than 1. One should note that very large step-sizes may provoke instability of the algorithm.
- **Parameter $\boldsymbol{\mu}$:** this parameter is devoted to properly scale the size of the main interval considered. We have found that establishing $0.01 \leq \mu \leq 0.1$ works quite well in most cases.
- **Slope coefficients ($\boldsymbol{\rho}$ and $\boldsymbol{\psi}$):** these parameters can be viewed as the weights attributed for each intermediate point considered. Basically, as it is shown in Fig. 2, behavior of the developed CNC-PF assimilates of a truncated version of the standard NR when the

intermediate slope coefficient ρ is high. Oppositely, when the extreme slope coefficient ψ is large, the CNC-PF is similar to a high order NR technique. It is expected that the main interval is large at first iterations, in this case, both slope coefficients are tuned based on the self-adapted mechanism and avoid the divergence. At end iterations, when the convergence may be ensured, size of the main interval tends to drastically decrease. In this situation, ρ is tuned high while ψ tends to vanish (see Fig. 3). This is explained in the fact that, frequently the solution at last iterations lies presumably close to the intermediate points. Increasing the value of ρ, one gives more importance to intermediate than extreme points, so that, convergence is normally accelerated as it is shown in Fig. 4 (note that point generated by larger values of ρ is very close to the solution). Anyway, the extreme slope coefficient should be upper bounded in order to avoid attributing too much weight to extreme points, which may lead to the divergence. We have founded that $1 \leq \psi^{max} \leq 2$ works quite well in most cases.

- **Number of intermediate points (m):** In this case, it is worth to mention that one cannot take less than 1 sub-intervals. Therefore, at least one intermediate point should be taken. Consequently, a good option is fixing $m^{min} \geq 2$ (it is worth to mention that $\boldsymbol{x}_b = \boldsymbol{x}_1$). Regarding the upper bound, one should note that, intuitively, considering more intermediate points should imply more robustness since more information is extracted from the increment vector, nevertheless, too many intermediate points should not be considered since they would imply many computations and, consequently, computational burden would grow. Therefore, we recommend tuning $m^{max} \geq 3$ so that at least two intermediate points are considered. The self-adapted mechanism of this parameter is focused on reducing it as soon as possible in order to save computational effort without affecting the robustness.

- **Damping factor (β):** this factor has been included in order to mimetic the standard formulation of Newton-Cotes formula. In that case, it takes different values depending on the numerical method employed (Trapezoidal, Simpson…). In the proposed CNC-PF, it acts as a compensation for the step size to update the state vector. Its logical behavior is reduced as algorithm evolves in the same sense of the step size. Anyway, it should be lower bounded in order to avoid excessively the increment vector. We have founded that establishing $2 \leq \beta^{min} \leq 4$ works quite well in many cases.

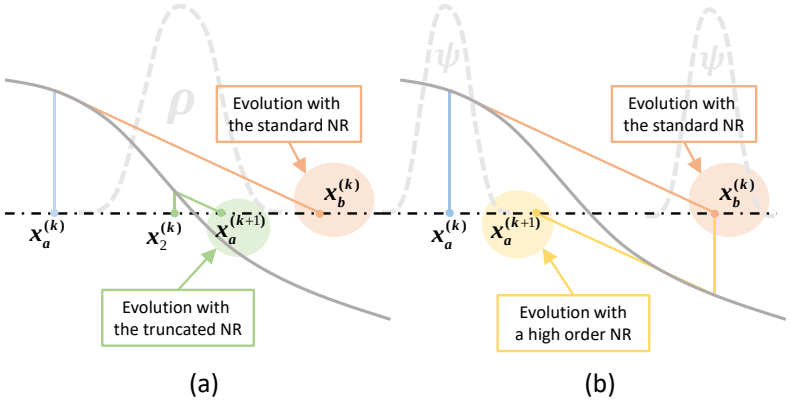

Fig 2 Behavior of proposed CNC-PF for large values of the intermediate (a) and extreme slope coefficients (b)

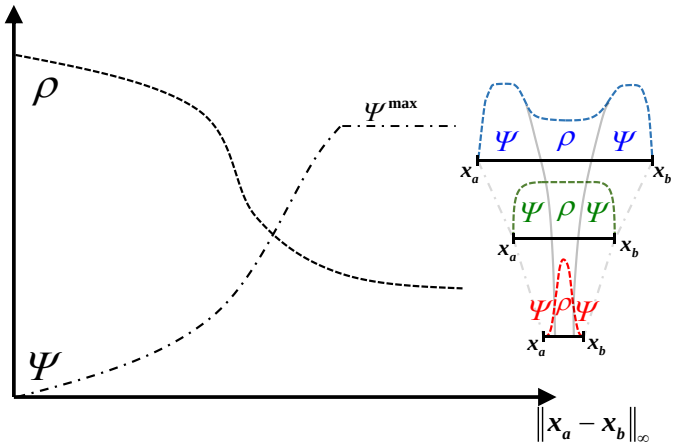

Fig 3 Behavior of slope coefficients depending on the size of main interval

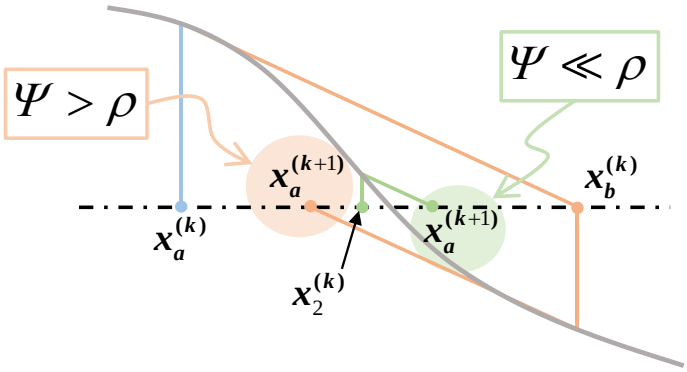

Fig 4 Behavior of proposed CNC-PF when the size of main interval is small enough (Note as the point calculated for a large value of $\boldsymbol{\rho}$ is closer to the solution)

The following steps clarify how the composite Newton-Cotes formula is integrated with power flow equations:

Step 0: Read the input data, construct the admittance matrix, initialize parameters ($m^{(0)} = m^{\max}$), select an initial guess point $\boldsymbol{x}^{(0)}$ and initialize the iteration counter $k = 0$;

Step 1: Set the initial guess as the upper bound of the main interval at the first iteration, it means, $\boldsymbol{x}_a^{(0)} = \boldsymbol{x}^{(0)}$. Then, evaluate the PF equations at this point $\mathbf{g}\left(\boldsymbol{x}_a^{(0)}\right)$;

Step 2: Evaluate and factorize $\boldsymbol{J}\left(\boldsymbol{x}_a^{(k)}\right)$ using LU decomposition;

Step 3: Calculate (10) by taking $\boldsymbol{y} = \boldsymbol{z} = \boldsymbol{x}_a^{(k)}$;

Step 4: Calculate the step size using (12);

Step 5: Calculate the lower bound of the main interval, it means $\boldsymbol{x}_b^{(k)}$, using (11). Evaluate the PF equations at this point $\mathbf{g}\left(\boldsymbol{x}_b^{(k)}\right)$;

Step 6: Calculate (10) by taking $\boldsymbol{y} = \boldsymbol{x}_a^{(k)}$ and $\boldsymbol{z} = \boldsymbol{x}_b^{(k)}$. One should note that the Jacobian matrix has not to be factorized again.

Step 7: Calculate the different parameters $\rho^{(k)}$, $\gamma^{(k)}$, $\psi^{(k)}$, $m^{(k)}$and $\beta^{(k)}$ using equations (16)-(20), respectively;

Step 8: Set the counter for the Composite loop $i = 2$

Step 9: Calculate the i^{th} intermediate value of (7) using (13), it means $\boldsymbol{x}_i^{(k)}$. Evaluate the PF equations at this point $\mathbf{g}\left(\boldsymbol{x}_i^{(k)}\right)$;

Step 10: Calculate (10) by taking $\boldsymbol{y} = \boldsymbol{x}_a^{(k)}$ and $\boldsymbol{z} = \boldsymbol{x}_i^{(k)}$. One should note that the Jacobian matrix has not to be factorized again;

Step 11: If $i = m^{(k)}$, then go to Step 12, otherwise, Set $i = i + 1$ and go back to Step 9.

Step 12: Calculate $\boldsymbol{\phi}^{(k)}$ using (14). The state vector (7) can now be updated using (15). One should note that the updated step vector is taken as the upper bound of the main interval at the following iteration, this means, $\boldsymbol{x}_a^{(k+1)}$ is calculated at this step;

Step 13: Check if the convergence criteria (8) is satisfied at this point ($\left\|\mathbf{g}\left(\boldsymbol{x}_a^{(k+1)}\right)\right\|_\infty \leq \varepsilon$),then stop, otherwise go to Step 14.

Step 14: Set $k = k + 1$, if $k > k^{\max}$, then stop, otherwise go back to Step 2.

In addition, the above solution steps are summarized in the flowchart of Fig. 5.

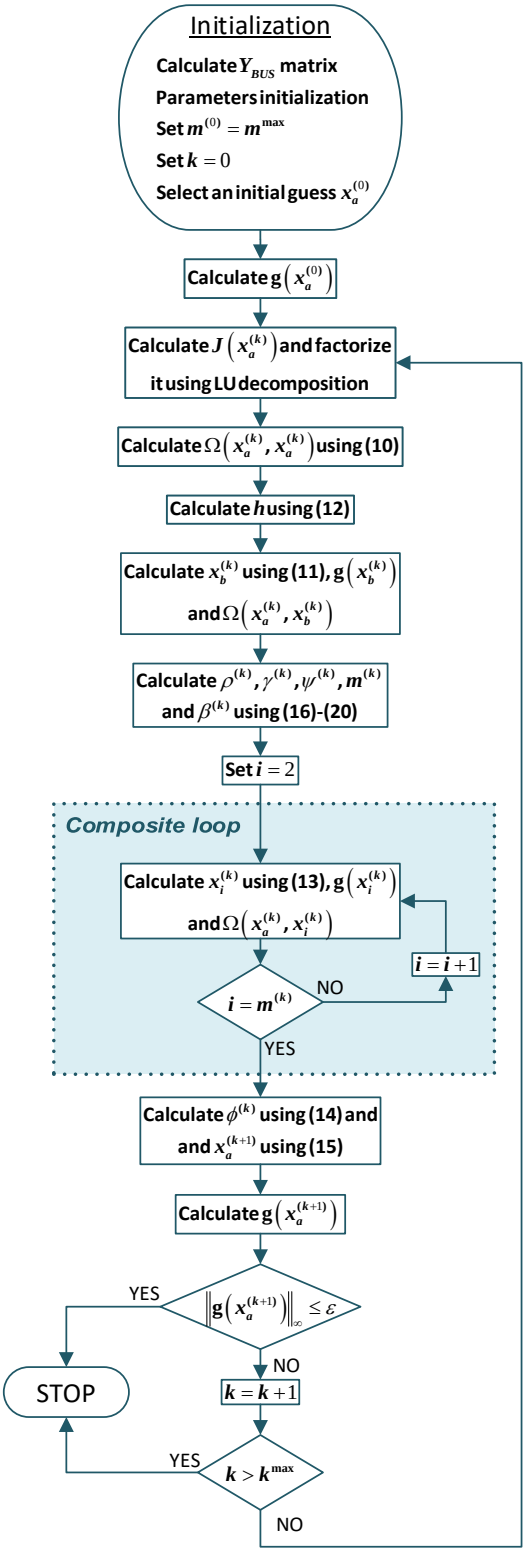

Fig 5 Flowchart of proposed CNC- PF method

4.1.- Managing reactive limits

Proposed method allows to easily incorporate any available strategy for considering reactive limits available for the standard NR. One common procedure [18, 24], proceeds as follows:

Step 1: Solve the PF problem using the proposed method. If there are not PV buses and PF has not found a solution, the problem is declared unfeasible.

Step 2: Check if any reactive limit is violated.

Step 3: Create a $\mathbb{Q}^+$ set with these generators' indices whose higher limit is violated.

Step 4: Create a $\mathbb{Q}^-$ set with these generators' indices whose lower limit is violated.

Step 5: Convert those generators $\in [\mathbb{Q}^+, \mathbb{Q}^-]$ to PQ buses.

Step 6: For each $j \in \mathbb{Q}^+$ set $Q_j = Q_j^{max}$

Step 7: For each $j \in \mathbb{Q}^-$ set $Q_j = Q_j^{min}$. Return to Step 1.

4.2.- Computational burden and rate of convergence

As it is well-known, the heaviest part of any PF calculation is the factorization of the Jacobian matrix. Even more, if one takes into account that normally the PF Jacobian matrix is not positive definite, consequently, LU decomposition has to be used instead of Cholesky's factorization. One can note that the proposed method only updates the Jacobian matrix once per iteration (at $\boldsymbol{x}_a$). Remainder calculations are vectors sums and products, which can be generally addressed in an efficient way [18].

In this sense, the developed CNC-PF should be quite efficient since only one LU decomposition is required each iteration. This supposes a computational burden similar to NR. Certainly, this is an outstanding feature in robust PF methods. For instance, the PF techniques based on the 4th order Runge-Kutta and the Bulirsch-Stoer methods developed in [18] and [24] respectively, require more than one factorization each iteration. On the other hand, the

Iwamoto's method, which can be considered as the standard robust PF technique, requires only one factorization each iteration, however, this method suffers of slow convergence. In Section 5, the superiority of developed method with respect to other robust PF techniques is demonstrated based on experimental results.

In order to calculate the increment vector at intermediate points and $\boldsymbol{x}_b$ efficiently, the calculation of Jacobian matrix at these points is avoided. Instead, only the function is evaluated and the Jacobian matrix at $\boldsymbol{x}_a$ is used.

Regarding to the order of convergence, it can be simply proved by reducing the proposed method without taking any intermediate point, thus, we can affirm in this case that $\boldsymbol{x}_a^{(k+1)} = \boldsymbol{x}_b^{(k)}$ if $h = 1$. Consequently, the proposed method would be the standard NR with quadratic convergence. Taking into account this situation is never possible, we can claim that order of convergence of proposed method is always smaller than 2 and it is strongly case-dependent.

4.3.- Behavior at turning points

As it is well-known, methods based on the Newton's increment vector (Newton-like methods), fail at turning points due to the singularity of Jacobian matrix. Proposed method should be considered as a Newton-like method and, consequently, it will fail at turning points (the maximum loadability point (MLP)). Continuation methods [32], are able to overcome issues of the Newton-like methods at turning points. In this case, the singularity of Jacobian matrix is avoided by progressively approximating the MLP and incrementing the system by adding a parameterization equation. Thus, multiple PF solutions are calculated carrying out two stages called predictor and corrector. At the corrector stage, the augmented PF equations are solved using some available technique. The proposed CNC-PF method can be successfully applied at the corrector stage of continuation techniques, as their issues at the MLP might be

overcome. This topic is out of the scope of the present paper and it should be addressed in the future works.

4.4.- Managing uncertainty and stochastic behavior of the input data

In modern power systems, managing the uncertainty or stochastic behavior in the input data is crucial. Uncertainty in input data may be provoked by error in estimations or measurements. This issue is becoming more noticeable in actual power system analysis, due to increasing penetration of renewable energies among other factors [33-40].

The proposed method does not take directly into account the uncertainty or stochastic behavior in the input data. Proposed formulation supposes that load/generation profiles are known with exactitude. Nevertheless, CNC-PF may handle uncertainty/stochastic behavior in input data using some available techniques in the literature. For example, the proposed method can be properly used in Monte Carlo simulations as multiple PF solutions have to be solved. Moreover, recently some interesting methodologies have been proposed in [41, 42]. Basically, they manage the uncertainty in the input data solving several PF problems. As Monte Carlo, developed method can be directly applied to those techniques and other similar.

Assessing the performance of the CNC-PF in this kind of problems is out of the scope of the present paper. This topic should be tackled in future works.

5. Tests and Results

During this section, the proposed CNC-PF method is tested on several ill-conditioned power systems under different scenarios. In all simulations, $\varepsilon = 10^{-8}$ has been considered. On the other hand, a flat start has been considered as initial guess. It is well-known that a flat initial guess is, generally, a reasonable option [43].

The proposed method and remainder considered PF techniques have been coded in MATPOWER v6.0 [44].

In order to prove the superiority of proposed CNC-PF method, it has been compared with the following standard and robust PF techniques:

- Standard NR [1]
- Iwamoto's method [13]
- Levenberg-Marquardt's method (LM) [20]
- Corrected Levenbenberg-Marquardt with non-monotone line search (LMNM) [22]
- 4^{th} order Runge-Kutta formula (RK4) [18]
- 2^{nd} order Adams-Bashforth's method (AB2) [23]
- Reverse Bulirsch-Stoer (RBS) [24]
- Ralston-homotopy methodology (RH) [25]

On the other hand, all studied methods have been used in an optimal way, based on the guidelines provided in their correspondent references. LM requires to tune the so-called damping factor, in this case, an adaptive mechanism has been used for this parameter as it is commented in reference [20]. Parameters involved in the PF solution based on LMNM method, have been tuned based on the recommended ranges of reference [22]. RK4 does not need to tune any parameter, it just requires to initialize the step size, in this case, as in reference [18], the initial step size has been taken equal to 1. For AB2, RBS and RH, parameters recommended in their references [23-25] have been used.

For the sake of completeness, flowcharts for the standard NR, LM and RK4 methods are shown in Fig. 6. Comprehensive flowcharts of remainder considered PF solution methods can be found in their respective references.

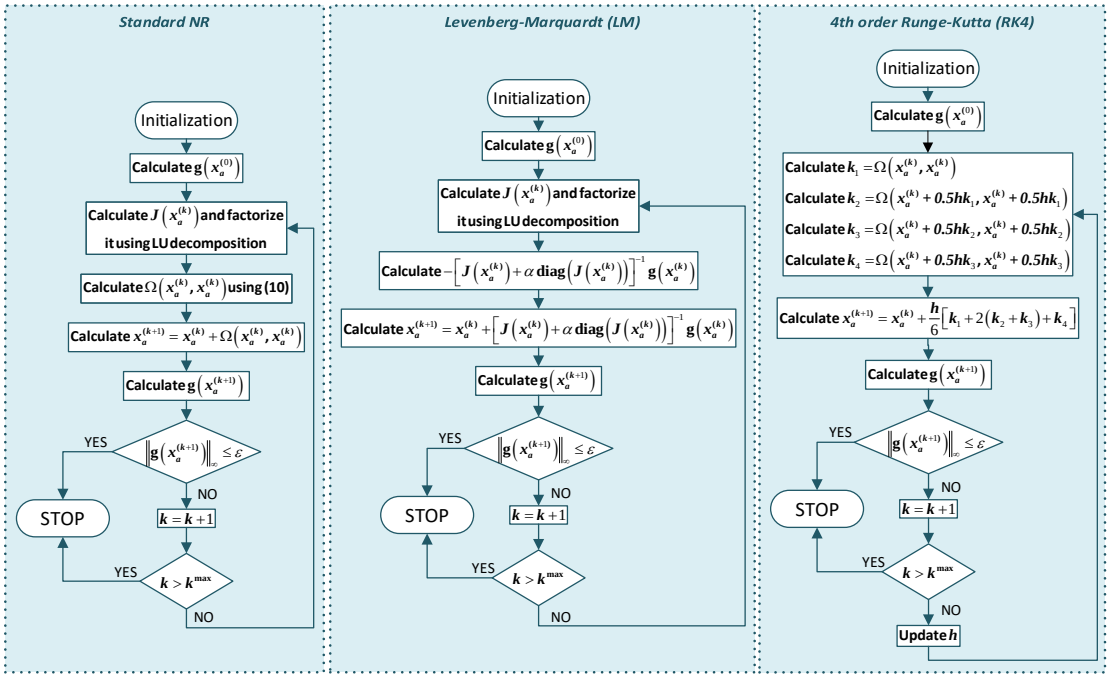

Fig 6 Flowcharts of standard NR, LM and RK4 methods

Parameters employed in simulations are reported in Table 1.

Table 1 Parameters considered in simulations

Parameter	Value
h^{min}	0.2
h^{max}	1.2
μ	0.06
ψ^{max}	2
m^{min}	2
m^{max}	3
β^{min}	2

5.1.- Description of studied cases

The proposed method has been validated using the following ill-conditioned systems:

- System #1: the 3012-bus portion of the Polish system at 2007-08 winter evening peak [45].
- System #2: the 3375-bus portion of the Polish system at 2007-08 winter evening peak [45].

- System #3: a duplicated version of System #2.
- System #4: the combined version forming by the 1354- and 3x2869-bus portion of the European transmission system from EU PEGASE project [46, 47].
- System #5: the 13659-bus portion of the European transmission system from EU PEGASE project [46, 47].
- System #6: a duplicated version of system #5.

Main characteristics of studied systems are reported in Table 2.

Table 2 Main characteristics of the studied ill-conditioned systems

System	Buses	Branches	Generators	Load		Nº of variables (n)
				MW	MVar	
System #1	3012	3572	502	27169	10200	5725
System #2	3375	4161	596	48363	19527	6355
System #3	6024	7145	770	54339	20401	11451
System #4	9961	15740	1.790	470371	100424	18131
System #5	13659	20467	4.092	381431	98523	23225
System #6	27318	40935	8.184	762863	197046	46451

5.2.- Number of iterations employed by different PF techniques

Table 3 reports the total number of iterations employed by different PF techniques used for solving the studied systems at loadbase conditions.

Table 3 Total number of iterations employed by different PF techniques used for solving the studied systems at loadbase conditions

Method	System #1	System #2	System #3	System #4	System #5	System #6
NR	Diverge	Diverge	Diverge	Diverge	Diverge	Diverge
Iwam.	2929	1753	Fail*	490	Fail*	Fail*
LM	> 5000	> 5000	> 5000	> 5000	> 5000	> 5000
LMNM	2502	2559	2502	2464	2362	2362
RK4	Diverge	Diverge	Diverge	Diverge	31	31
AB2	31	32	32	31	23	51
RBS	15	15	Fail*	15	16	16
RH	17	17	Diverge	Diverge	17	17
CNC-PF	**9**	**10**	**11**	**10**	**11**	**11**

* Low voltage solution was obtained

As expected, NR diverged in all studied systems. Results for LM have not been reported since this algorithm was intentionally stopped when the number of iterations surpassed 5000. RK4 diverged in Systems #1, #2, #3 and #4. RBS obtained the low voltage (unstable) solution of PF problem in System #3 while RH diverged in Systems #3 and #4. Both AB2 and CNC-PF

successfully solved all studied systems. In all cases, the proposed method employed less iterations than remainder considered techniques.

In order to demonstrate that developed CNC-PF produced an accurate solution, the following index is defined:

$$\eta = \|\boldsymbol{x}_{\boldsymbol{CNC-PF}} - \boldsymbol{u}\|_{\infty} \quad (21)$$

where $\boldsymbol{x}_{\boldsymbol{CNC-PF}}$ and $\boldsymbol{u}$ are the values of (7) obtained by CNC-PF and other method at the last iteration. This index is used to verify that the solution obtained by the CNC-PF is accurate and proximately identical with those obtained by the other PF methods. Table 4 presents the values of η for the developed CNC-PF compared with remainder considered PF methods.

Table 4 Value of η for the different considered methodologies compared with the developed CNC-PF

Method	System #1	System #2	System #3	System #4	System #5	System #6
Iwam.	5.7×10^{-9}	1.1×10^{-9}	--	3.4×10^{-8}	--	--
LMNM	1.8×10^{-8}	1×10^{-8}	2.1×10^{-7}	1.2×10^{-6}	9.8×10^{-6}	1.5×10^{-5}
RK4	--	--	--	--	9.7×10^{-9}	5.7×10^{-9}
AB2	1.8×10^{-9}	2.5×10^{-9}	5.6×10^{-9}	1.3×10^{-8}	6.7×10^{-9}	3.2×10^{-9}
RBS	2.2×10^{-9}	1.7×10^{-9}	--	1.5×10^{-8}	3.1×10^{-8}	2.9×10^{-8}
RH	1.6×10^{-9}	1.9×10^{-9}	--	--	3.1×10^{-9}	4.4×10^{-8}

Based on the results reported in Table 4, one can conclude that the developed CNC-PF method is accurate since the value of η was low enough in all studied cases compared with all other considered methods.

When the loading level of system is increased, the condition is degraded. This is frequently reflected in slow convergence. Table 5 reports the total number of iterations employed by considered techniques for solving the studied systems at heavy loading conditions. To increase the loading level of studied systems, the scheduled active and reactive power have been modified as follows:

$$P_i^{sch} = \lambda P_i^{sch}, \forall i \in [\boldsymbol{PQ}, \boldsymbol{PV}]^T \quad (22)$$

$$Q_i^{sch} = \lambda Q_i^{sch}, \forall i \in [\boldsymbol{PQ}]^T \quad (23)$$

where, $\lambda \in \mathbb{R}^+$, $\boldsymbol{PQ} \in \mathbb{N}^{n_l}$ and $\boldsymbol{PV} \in \mathbb{N}^{n_g}$ are the sets of PQ and PV buses indices, respectively.

Table 5 Total number of iterations employed by different PF techniques for solving the studied systems with heavy loading conditions

Method	**System #1** $\lambda = 1.2734$	**System #2** $\lambda = 1.1586$	**System #3** $\lambda = 1.0806$	**System #4** $\lambda = 1.0111$	**System #5** $\lambda = 1.0017$	**System #6** $\lambda = 1.0017$
NR	Diverge	Diverge	Diverge	Diverge	Diverge	Diverge
Iwam.	2699	1580	Diverge	489	58	57
LM	> 5000	> 5000	> 5000	> 5000	> 5000	> 5000
LMNM	2502	2559	Diverge	2464	Fail*	Fail*
RK4	Diverge	Diverge	Diverge	Diverge	Fail*	Fail*
AB2	35	36	38	32	Diverge	Diverge
RBS	15	15	Diverge	15	Diverge	Diverge
RH	19	19	Diverge	Diverge	17	17
CNC-PF	**10**	**10**	**12**	**11**	**10**	**11**

* Low voltage solution was obtained

The performance of the PF techniques are similar to their performance in the loadbase scenario. In this case, Iwamoto also successfully converged in Systems #5 and #6 while LMNM, RK4, AB2 and RBS failed. On the other hand, the proposed method successfully solved all studied systems. As expected, CNC-PF employed less iterations than remainder techniques in all cases.

5.3.- Execution time employed by different PF techniques

Execution time of any algorithm is normally quite difficult to measure. It is strongly affected by the computer platform, software environment (C, Java, Matlab, etc) and other computational activities. Nevertheless, it can be effectively used to compare different algorithms if similar efforts are made to optimize the code.

Table 6 presents the computational time (in seconds) of PF techniques used for solving the studied systems at loadbase and heavy loading conditions. 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) over Matlab R2014b. The computation time has been calculated as mean value of 100 simulations, in order to avoid influence of other computational activities.

Table 6 Execution time [s] of different PF techniques used for solving the studied systems

Method	System #1	System #2	System #3	System #4	System #5	System #6
			LOADBASE			
Iwam.	58.2254	37.9755	--	29.3796	--	--
LMNM	60.0984	96.9247	127.0688	234.8461	407.2067	862.5420
RK4	--	--	--	--	10.4643	21.7048
AB2	0.5779	0.6579	1.1251	1.8322	2.0749	9.0716
RBS	1.1045	1.2448	--	2.8841	4.8995	9.6801
RH	0.3688	0.3778	--	--	1.4834	3.0772
CNC-PF	**0.1986**	**0.2440**	**0.4393**	**0.6520**	**1.0525**	**2.1383**
			HEAVY LOADING			
Iwam.	52.7882	33.7690	--	29.3668	5.0419	10.3210
LMNM	60.0984	96.9247	--	234.8461	--	--
RK4	--	--	--	--	--	--
AB2	0.6488	0.7398	1.3440	1.8993	--	--
RBS	1.1045	1.2448	--	2.8841	--	--
RH	0.3749	0.4100	--	--	1.4834	3.0772
CNC-PF	**0.2186**	**0.2440**	**0.4808**	**0.7205**	**0.9621**	**2.1383**

From this table, it can be observed that CNC-PF is the fastest method. LMNM, RK4 and RBS require more than one factorization each iteration, moreover they employed more iterations than CNC-PF in all considered cases. Iwamoto, AB2, RH and proposed method have similar computational burden, nevertheless, CNC-PF employed less iterations in all systems.

In order to demonstrate that the computational time reported are coherent, Table 7 presents the total number of Jacobian matrix factorizations for each studied case. As commented previously, factorization of the Jacobian matrix is by far the heaviest computational burden of PF calculation. Hence, it is expected that a reducing number of matrix factorizations will be reflected in a competitive execution time.

Table 7 Total number of Jacobian matrix factorizations required for solving the studied systems

Method	System #1	System #2	System #3	System #4	System #5	System #6
LOADBASE						
Iwam.	2929	1753	--	490	--	--
LMNM	5004	5118	5004	4928	4724	4724
RK4	--	--	--	--	124	124
AB2	32	33	33	32	24	52
RBS	36	36	--	36	38	38
RH	17	17	--	--	17	17
CNC-PF	**9**	**10**	**11**	**10**	**11**	**11**
HEAVY LOADING						
Iwam.	2699	1580	--	489	58	57
LMNM	5004	5118	--	4928	--	--
RK4	--	--	--	--	--	--
AB2	36	37	39	33	--	--
RBS	36	36	--	36	--	--
RH	19	19	--	--	17	17
CNC-PF	**10**	**10**	**12**	**11**	**10**	**11**

As commented previously, the proposed CNC-PF only requires one matrix factorization each iteration, which supposes a computational burden similar to NR. Iwamoto's method, AB2 and RH also shows similar computational burden, however, CNC-PF always required less iterations than remainder methods. From Table 7, it can be observed that the developed technique is always more efficient than remainder studied techniques, since CNC-PF always requires less factorizations.

5.4.- Reactive power limits enforcement

The ability of proposed method for solving the PF considering the generators' reactive limits is assessed. The strategy described in Section 4 has been used to achieve that.

In this case, we have opted the simplified steady state model of excitation system, as in most papers devoted in the Power-Flow problem. Hence, the voltage magnitudes of PV buses are kept constant until their reactive limits are violated. In case of violation, the generated reactive powers are fixed constant at their limits and the voltages are relaxed, which indicate that these buses are converted to PQ buses. This model, although simple indeed, is considered valid for PF purposes and used in multiple references [18, 23-25], as in the original code of MATPOWER [44].

Fig. 6 shows the convergence characteristics of PF techniques used for solving the studied systems considering generators' reactive limits while Table 8 reports the required execution time in seconds and total number of iterations. One should note that NR and LM have not been included in this analysis due to their failure to solve the studied systems at the loadbase scenario.

Table 8 Execution time [s] and total number of iterations (in parenthesis) of PF methods used for solving the studied systems considering generators' reactive limits

Method	System #1	System #2	System #3	System #4	System #5	System #6
Iwam.	135.1999 (6839)	109.4977 (5003)	Fail*	94.2391 (1551)	Fail*	Fail*
LMNM	139.7590 (5603)	246.4160 (6606)	304.2820 (5603)	767.7020 (7711)	737.6870 (4213)	1516.7 (4213)
RK4	Fail*	Fail*	Fail*	Fail*	17.4661 (53)	35.5547 (53)
AB2	0.7159 (35)	0.7952 (35)	1.3459 (36)	Diverge	Diverge	Diverge
RBS	2.3291 (38)	3.2487 (47)	Fail*	10.6635 (54)	8.4887 (29)	17.4791 (29)
RH	0.8071 (43)	1.1062 (53)	Fail*	Fail*	2.7132 (31)	5.4530 (31)
CNC-PF	**0.4393 (20)**	**0.5891 (24)**	**0.8506 (21)**	**1.8176 (28)**	**1.5253 (16)**	**3.0617 (16)**

* Fail for solving the loadbase scenario

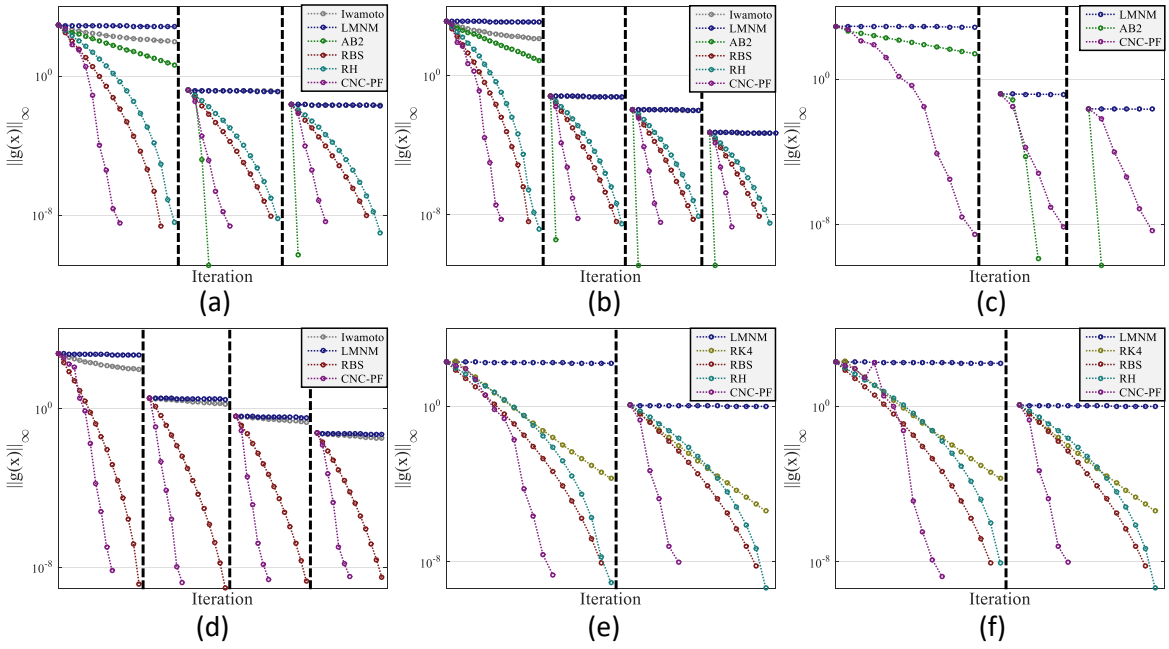

Fig 6 Convergence characteristics of different PF methods in System #1 (a), System #2 (b), System #3 (c), System #4 (d), System #5 (e) and System #6 (f) considering generators' reactive limits

In this case, performance of different methods mainly depend on how many PF problems have to be solved to achieve the feasible solution. AB2 diverged in Systems #4, #5 and #6. In all studied systems, the proposed CNC-PF outperformed remainder techniques in the number of iterations and execution time. From Fig. 6, it can be observed that CNC-PF converges much faster than remainder methods. Nevertheless, AB2 shows occasionally faster convergence after the first PF solution. It is explained in the fact that this methodology incorporates an acceleration mechanism, which is activated when the residual is smaller than a predetermined threshold. Anyway, CNC-PF always employed less iterations during the whole iterative procedure, due to AB2 employs many iterations for the first PF calculation.

5.5.- Comparison with NR at safe initialization

In all simulations, a flat start has been considered for initializing the iterative procedure. With this scenario, the proposed CNC-PF clearly outperformed other PF methods, especially NR. Failure of the standard NR in the simulations considered is mainly due to its local convergence, which provokes instability when the initial guess point is far away from the solution.

Nevertheless, an improved initialization strategy can be used in order to improve the numerical stability of NR. The simplest and secure method to find good starting point is to use the Gauss-Seidel method (GS). GS always converges to the stable equilibrium point if it exists. Disadvantage of the GS is the hyperbolic characteristic of convergence i.e. initially this method converges very fast but later when approaching solution becomes very slow. This slow convergence is the reason why GS is not widely used. However, thanks to its fast initial convergence it can be used as the method to find good starting point for the NR. We have analysed the performance of this hybrid GS-NR in comparison with the proposed CNC-PF. To do that, the iterative procedure is initialized via GS after the first iteration, then the algorithm

switches to the standard NR. Table 9 presents the execution time consumed by GS-NR and proposed CNC-PF in the studied systems.

Table 9 Execution time [s] of hybrid GS-NR and CNC-PF

Method	System #1	System #2	System #3	System #4	System #5	System #6
LOADBASE						
GS-NR	0.2463	0.2948	Fail*	2.1265	Fail*	Fail*
CNC-PF	**0.1986**	**0.2440**	**0.4393**	**0.6520**	**1.0525**	**2.1383**
HEAVY LOADING						
GS-NR	0.3673	0.4112	Fail*	2.1478	4.8179	15.2570
CNC-PF	**0.2186**	**0.2440**	**0.4808**	**0.7205**	**0.9621**	**2.1383**
GENERATORS' REACTIVE LIMITS ENFORCED						
GS-NR	0.5760	0.8896	Fail*	7.4971	Fail*	Fail*
CNC-PF	**0.4393**	**0.5891**	**0.8506**	**1.8176**	**1.5253**	**3.0617**

* Low voltage solution was obtained

Divergence of NR is prevented in all cases, however, GS-NR occasionally converged to the low voltage solution. Moreover, the proposed CNC-PF was always faster, especially in very large systems.

The superiority of developed method is explained in the fact that GS is quite inefficient especially in large systems. This is due to iterative solution of PF using GS involves several loops, which is always less efficient than vectors and matrix calculations [30].

6. Conclusions

This paper has proposed a novel method, CNC-PF, for solving the PF problem of ill-conditioned power systems. It is inspired by the Composite Newton-Cotes formula. The proposed CNC-PF method is robust and efficient enough to properly manage ill-conditioned systems, while standard techniques (like NR), are likely to fail. Moreover, its computational burden is similar to NR since only one matrix factorization is required each iteration. The proposed CNC-PF has been tested on several ill-conditioned systems at loadbase and heavy loading conditions. Its performance has been compared with other well-known PF techniques. The results shown the outstanding robustness and capability of proposed method for successfully solving all studied cases while other robust and standard techniques have shown

different convergence difficulties. Finally, the ability of the proposed CNC-PF for solving the PF when generators' reactive limits are considered, has been tested. In all numerical experiments, the proposed CNC-PF is the most efficient method, employing less iterations than remainder considered techniques. Moreover, it is the fastest, which demonstrated by comparing the execution time. The obtained results demonstrate that the proposed CNC-PF supposes an important contribution for efficiently solving the PF in ill-conditioned systems.

Future works should be focused on applying the proposed PF method to other related problems such as continuation and uncertain power flow analysis.

References

[1] Tinney WF, Hart CE. Power flow solution by Newton's method. IEEE Trans Power App Syst 1967; PAS-86(11): 1449–60.

[2] Stott B, Alsac O. Fast decoupled load flow. IEEE Trans Power App Syst 1974; PAS-93(3): 859–69.

[3] Van Amerongen RAM. A General-Purpose Version of the Fast Decoupled Load Flow. IEEE Trans Power Syst 1989; 4(2): 760-70.

[4] Wang L, Lin XR. Robust fast decoupled power flow. IEEE Trans Power Syst 2000; 15(1): 208-15.

[5] Garcia PAN, Pereira JLR, Carneiro S, Vinagre MP, Gomes FV. Improvements in the representation of PV buses on three-phase distribution power flow. IEEE Trans Power Deliv 2004; 19(2): 894-6.

[6] Yan P, Sekar A. Steady-state analysis of power system having multiple FACTS devices using line-flow-based equations. IEE Proc – Gen, Transm and Distrib 2005; 152(1): 31-9.

[7] Kamel S, Abdel-Akher M, Jurado F. Improved NR current injection load flow using power mismatch representation of PV bus. Int J Elect Power Energy Syst 2013; 53: 64-8.

[8] Saleh SA. The Formulation of a Power Flow Using d-q Reference Frame Components—Part I: Balanced 3ϕ Systems. IEEE Trans Ind Appl 2016, 52(5): 3682-93.

[9] Derakhshandeh SY, Pourbagher R. Application of high-order Newton-like methods to solve power flow equations. IET Gener, Transm Distrib 2016; 10(8): 1853-9.

[10] Tostado M, Kamel S, Jurado F. Developed Newton-Raphson based Predictor-Corrector load flow approach with high convergence rate. Int J Elect Power Energ Syst 2019; 105: 785-92.

[11] Powell L. Power System Load Flow Analysis, ser. McGraw-Hill Professional Series. New York: McGraw- Hill; 2004, NY, USA.

[12] Fatemi SM, Abedi S, Gharehpetian GB, Hosseinian SH, Abedi M. Introducing a Novel DC Power Flow Method With Reactive Power Considerations. IEEE Trans Power Syst 2015; 30(6): 3012-23.

[13] Iwamoto S, Tamura Y. A Load Flow Calculation Method for Ill-Conditioned Power Systems. IEEE Trans Power App Syst 1981; PAS-100(4): 1736-43.

[14] Schaffer MD, Tylavsky DJ. A nondiverging polar-form Newton-based power flow. IEEE Trans Ind Appl 1988; 24(5): 870-7.

[15] Bijwe PR, Kelapure SM. Nondivergent fast power flow methods. IEEE Trans Power Syst 2003; 18(2): 633-8.

[16] Shahriari A, Mokhlis H, Karimi M, Bakar AHA, Illias HA. Quadratic Discriminant Index for Optimal Multiplier Load Flow Method in ill conditioned system Int J Elect Power Energ Syst 2014; 60: 378-88.

[17] Neuberger JW. Sobolev Gradients and Differential Equations, 2nd ed. New York: Springer; 2010, NY, USA.

[18] Milano F. Continuous Newton's Method for Power Flow Analysis. IEEE Trans Power Syst 2009; 24(1): 50-7.

[19] Tostado-Véliz M, Kamel S, Jurado F. Development of combined Runge-Kutta Broyden's load flow approach for well and ill-conditioned power systems. IET Gener, Transm Distrib 2018; 12(21): 5723-9.

[20] Milano F. Analogy and Convergence of Levenberg's and Lyapunov-Based Methods for Power Flow Analysis. IEEE Trans Power Syst 2016; 31(2): 1663-4.

[21] Pourbagher R, Derakhshandeh SY. Application of high-order Levenberg–Marquardt method for solving the power flow problem in the ill-conditioned systems. IET Gener, Transm Distrib 2016; 10(12): 3017-22.

[22] Pourbagher R, Derakhshandeh SY. A powerful method for solving the power flow problem in the ill-conditioned systems. Int J Electr Power Energy Syst 2018; 94: 88-96.

[23] Tostado-Véliz M, Kamel S, Jurado F. Development of different Load Flow methods for solving Large-scale ill-conditioned Systems. Int Trans Elect Energy Syst 2019; 29(4).

[24] Tostado-Véliz M, Kamel S, Jurado F. A robust Power Flow Algorithm Based on Bulirsch-Stoer Method. IEEE Trans Power Syst 2019; 34(4): 3081-3089.

[25] Tostado M, Kamel S, Jurado F. Several robust and efficient load flow techniques based on combined approach for ill-conditioned power systems. Int J Elect Power Energ Syst 2019; 110: 349-56.

[26] Kress R. Numerical Analysis. New York: Springer; 1998, NY, USA.

[27] Press WH, Teukolsky SA, Vetterling TH, Flannery BP. Numerical Recipes: The art of scientific computing, 3rd ed. Cambridge: Cambridge University Press; 2007, UK.

[28] Weerakoon S, Fernando TGI. A variant of the Newton's method with Accelerated Third-Order Convergence. Appl Math Letters 2000; 13: 87-93.

[29] Noor MA, Waseem M. Some iterative methods for solving a system of nonlinear equations. Comput Math Appl 2009; 57(1): 101-6.

[30] Milano F. Power System Modelling and Scripting. Berlin/Heidelberg: Springer; 2010, Germany.

[31] Liu CS. A Dynamical Tikhonov Regularization for Solving Ill-posed Linear Algebraic Systems. Acta Applicandae Mathematicae 2013; 123(1): 285-307.

[32] Allgower, Eugene L, Kurt Georg. Numerical continuation methods: an introduction, vol. 13. Berlin: Springer Science & Business Media; 2012, Germany.

[33] Fang D, Zou M, Coletta G, Vaccaro A, Djokic SZ. Handling uncertainties with affine arithmetic and probabilistic OPF for increased utilisation of overhead transmission lines. Electr Power Syst Res 2019; 170: 364-77.

[34] Coletta G, Vaccaro A, Villacci D, Fang D, Djokic SZ. Affine arithmetic for efficient and reliable resolution of weather-based uncertainties in optimal power flow problems. Int J Electr Power Energy Syst 2019; 110: 713-24.

[35] Cataliotti A, Cervellera C, Cosentino V, Di Cara D, Gaggero M, Maccio D, Marsala G, Ragusa A, Tinè G. An Improved Load Flow Method for MV Networks Based on LV Load Measurements and Estimations. IEEE Trans Instr Meas 2019; 68(2): 430-8. DOI: 10.1109/TIM.2018.2847818

[36] Cataliotti A, Cosentino V, Di Cara D, Tinè G. LV Measurement Device Placement for Load Flow Analysis in MV Smart Grids. IEEE Trans Instr Meas 2016; 65(5): 999-1006. DOI: 10.1109/TIM.2015.2494618

[37] Kundu P, Pradhan AK. Power Network Protection Using Wide-Area Measurements Considering Uncertainty in Data Availability. IEEE Syst J 2018; 12(4): 3358-68.

[38] O'Connell A, Soroudi A, Keane A. Distribution Network Operation Under Uncertainty Using Information Gap Decision Theory. IEEE Trans Smart Grid 2018; 9(3): 1848-58.

[39] Rakpenthai C, Uatrongjit S. Optimal worst case solutions for state estimation of power system with uncertain network parameters. IEEJ Trans Electr Electronic Eng 2016; 11: S37-S42.

[40] Sihag S, Tajer A. Power System State Estimation under Model Uncertainty. IEEE J Selected Topics Signal Proc 2018; 12(4): 593-606. DOI: 10.1109/JSTSP.2018.2827322

[41] Xu X, Yan Z. Probabilistic load flow calculation with quasi-Monte Carlo and multiple linear regression. Int J Electr Power Energy Syst 2017; 88: 1-12.

[42] Amid P, Crawford C. A Cumulant-Tensor-Based Probabilistic Load Flow Method. IEEE Trans Power Syst 2018; 33(5): 5648-56.

[43] Stott B. Effective starting process for Newton-Raphson load flows. Proc Inst Elect Eng 1971; 118(8): 983–7.

[44] Zimmerman RD, Murillo-Sánchez CE, Thomas RJ. Matpower: Steady-State Operations, Planning and Analysis Tools for Power Systems Research and Education. IEEE Trans Power Syst 2011; 26(1): 12-9.

[45] MATPOWER cases. Available: http://www.pserc.cornell.edu/matpower/

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

[47] Fliscounakis S, Panciatici P, Capitanescu F, Wehenkel L. Contingency Ranking with Respect to Overloads in Very Large Power Systems Taking Into Account Uncertainty, Preventive, and Corrective Actions. IEEE Trans Power Syst 2013; 28(4): 4909-17.