Power Flow Solution of Ill-conditioned Systems Using Current Injection Formulation: Analysis and a Novel Method

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]*Electrical Engineering Department, 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 400030, China*

Abstract **–** Along the conventional approach in power mismatches, the current injection form of the Power Flow (PF) equations is also quite popular and extensively considered for PF analysis, however, this approach has not been profusely studied for solving ill-conditioned systems yet. This paper aims at filling this gap by studying various conventional and modern robust solvers in current injection formulation. In addition, a novel robust PF technique using this formulation is developed. Several snapshots of the Polish Transmission systems are used to validate the studied PF solvers. It is concluded that the PF equations tend to be notably worse conditioned when they are handled in the current injection form compared with its power mismatch form, which complicates the solution of ill-conditioned systems. In this regard, the results obtained with the proposed technique are promising, showing a good trade-off between robustness and efficiency.

Keywords: Power Flow, Ill-conditioned systems, Current Injection Mismatch, Computational efficiency.

---

*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

PF tool is extensively used in power system planning, operation, voltage stability studies or short-circuit analysis [1], among others. It also finds applications in market studies or real time applications. In general, PF solves a set of nonlinear algebraic power equations using an iterative method [2].

An alternative PF formulation based on current rather than power injections was proposed in [3]. This approach has been studied in either rectangular [4] or polar coordinates [5] along with serving as base for developing hybrid formulations [6-8].

One of the main concerns in the usage of the current injection formulation (CIF) of the PF equations is the representation of the PV buses. This is due to the reactive power mismatch for this bus type is unknown. This has been addressed in some references by introducing the nodal reactive power mismatch as a variable in [4, 9], and an additional equation for voltage magnitude constraint is required. The approach proposed in [10] directly treats the injected reactive power at PV buses as a variable, which is a function of the unknowns. In [7, 11, 12], the PV buses are treated in power injection mismatches while the PQ buses are represented in CIF.

With the increasing usage of FACTS technologies in transmission and distribution systems, important efforts have been made for developing FACTS models for PF analysis using CIF. In [13], a generic representation of voltage and power control devices for NR PF solution using CIF was presented. In [14], a three phase PF model suitable for CIF was presented. This approach is comprehensively used for solving the PF problem of distribution networks. To solve other network configurations, several models have been developed such as the four conductor (3P+N) representation in [15], the multiphase model in [16], the combined three-phase medium voltage + four-conductor low voltage scheme in [17] or the governor model proposed in [18].

Either based on power or current injection formulations, the PF constitutes a nonlinear problem which is traditionally solved using iterative techniques. Power systems can be classified as well or ill-conditioned cases. For well-conditioned cases, conventional PF methods such as NR, decoupled techniques [19] or high order Newton-like methods [20], converge from a flat start without any problem. However, these methods may experience numerical stability problems in ill-conditioned cases. For these cases, alternative robust PF solvers have been presented during decades.

Robust PF techniques became to be studied at early 80's, where the optimal step-size based methods have been proposed [21]. In fact, they are modified versions for NR method. In these methods, the optimal multiplier is calculated as result of an optimization problem, which is raised from the second order Taylor expansion of the PF equations in rectangular coordinates. Typically, the optimal multiplier decreases as the PF solution is reached, so that, the optimal step-size based methods are often quite slow [22]. Although the power injection formulation in rectangular coordinates is preferred for the optimal multiplier-based techniques, polar forms [23] and CIF [24] have also been studied.

Continuation, homotopy and metaheuristic techniques have been also considered for solving ill-conditioned systems [25-27]. Continuation and homotopy methods are widely used in related tools like voltage stability analysis. For example, the technique presented in [25], uses an augmented form of the PF equations where the loading level is introduced as a parameter, thus, various operating points for different loading levels are calculated up to the maximum loadability point. While this technique is useful for determining the loading margin of the system or calculating all the type-1 PF solutions, it needs high computational burden as many PF solutions have to be obtained. Similarly, those metaheuristic techniques as applied to PF analysis are normally less efficient than the traditional iterative methods.

Another category is the robust solvers based on the Continuous Newton's method [28]. The Continuous Newton's framework establishes an analogy between the solution of a nonlinear system and the solution of a set of differential equations. Based on this analogy, any numerical method (e.g. the Runge-Kutta formulas [29]), can be used for developing robust and efficient PF solution techniques. This idea was firstly exploited by Milano in [30] and by the authors in [31]. This kind of methods are actually Newton-based techniques, so that they still fail at turning points. In addition, they are not globally convergent, where they are sensitive to the starting guess. The PF solvers based on the Continuous Newton's method are inevitably less efficient than NR method. For example, if one considers a Runge-Kutta formula with four stages as in [30], four LU factorizations must be calculated for each iteration during the iterative process. Hence, the resulting technique would be four times computationally more expensive than NR method. In [30], it was pointed out that maybe the explicit form of the Continuous Newton's method is tractable for PF analysis, however, the same author recently studied the implicit form of the Continuous Newton's method for solving the PF equations in [32].

By establishing the PF equations as a set of artificial dynamic equations, a widely convergent solution paradigm was proposed in [33, 34]. The resulting solution framework can be solved using integration routines such as the ordinary differential equations solvers in MATLAB. However, the computational burden of these techniques dramatically grows with the system size, hence they are not suitable for large-scale systems without sophisticated arrangements like parallel computation schemes as pointed out in [33].

Recently, the Levenberg-type methods have gained popularity for solving ill-conditioned cases [35-37]. These methods use a regularization scheme for improving the conditioning of the Jacobian matrix, so that they are even robust at turning points. A critical point of these techniques is the so-called damping factor and how it is managed.

If this parameter is not correctly updated, the solver may either converge slowly or find an inaccurate solution [36]. Typically, a self-updated mechanism is adopted for properly tuning the damping factor [35, 37], however, there is no a unified approach for establishing this mechanism as observed in [35, 38]. In large systems, the Levenberg-type methods are low efficient because they need to compute an extra matrix-matrix product compared with NR method.

1.1.- Contributions

Referring to the available literature about PF robust solvers, it can be observed that most of the references deal with the PF formulation in the standard power injection form. To the best of our knowledge, only the reference [24] has succinctly studied how this formulation behaves in ill-conditioned cases. Motivated by this gap, this paper aims at covering the following points:

- Firstly, this paper aims at discerning how the PF equations behave in power or current-based formulations. This test is made by measuring the degree of ill-conditioning of the Jacobian matrix using both formulations. In addition, the behavior of NR in some standard systems using the power and current-based formulations is outlined.
- Studying the performance of various recent PF robust solvers in current injection formulation. In this case, the Newton-like techniques such as [30-32] are suitable for being easily adapted to alternative formulations such as CIF.
- Developing a novel efficient and robust PF solution technique suitable for CIF.
- Various realistic large-scale systems serve to show the performance of the developed method along with NR and the other tested robust solvers in CIF.

1.2.- Paper Organization

Remainder of this paper is organized as follows. The PF problem in current injection formulation is outlined in Section 2. Section 3 describes a novel PF solution method in current injection formulation. Various numerical experiments with results are presented in Section 4. This paper is concluded in Section 5.

2.- Current injection formulation

During this paper, the polar form of CIF is followed as described in [5], since this polar form is customary followed by most of the studied robust PF solvers. In addition, this form brings less equations than the rectangular formulation, so that the size of the system is small and the computations using vectors and matrix operations [39] are, usually, less expensive.

Firstly, the current nodal balance for the i^{th} bus is given by:

$$I_i^{gen} - I_i^{dem} = \sum_{j=1}^{n_b} Y_{ij} V_j \tag{1}$$

where, $I_i^{gen} \in \mathbb{C}$ and $I_i^{dem} \in \mathbb{C}$ denote the generator and demanded current at i^{th} bus, respectively; n_b is the number of buses; $V_j = |V_j| \angle \delta_j \in \mathbb{C}$ is the complex voltage at j^{th} bus; and $Y_{ij} = |Y_{ij}| \angle \theta_{ij} \in \mathbb{C}$ is the ij^{th} element of the nodal admittance matrix (i.e. $\mathbf{Y}_{\text{bus}} \in \mathbb{C}^{n_b \times n_b}$). Equation (1) can be easily rewritten in matrix form as follows:

$$\Delta \boldsymbol{i} = \mathbf{Y}_{\text{bus}} \boldsymbol{v} \tag{2}$$

where, $\Delta \boldsymbol{i} \in \mathbb{C}^{n_b \times 1}$ is the vector of nodal current mismatches ($\Delta i_i = I_i^{gen} - I_i^{dem}$); and $\boldsymbol{v} \in \mathbb{C}^{n_b \times 1}$ is the vector of nodal voltages ($v_i = |V_i| \angle \delta_i$). It is useful to express (2) as a function of the injected nodal powers as follows:

$$\Delta \boldsymbol{s}^* \oslash \boldsymbol{v}^* = \mathbf{Y}_{\text{bus}} \boldsymbol{v} \tag{3}$$

where, $\boldsymbol{s} \in \mathbb{C}^{n_b \times 1}$ is the vector of nodal complex power balances (i.e. $\Delta s_i = |\Delta S_i| \angle \varphi_i = S_i^{gen} - S_i^{dem}$ where $S_i^{gen} \in \mathbb{C}$ and $S_i^{dem} \in \mathbb{C}$ denote the generator and demanded complex injected power at i^{th} bus, respectively); $\oslash$ is the Hadamard division (i.e. element-by-

element division); and $[\cdot]^*: \mathbb{C} \to \mathbb{C}$ is the conjugate operator. By separating (3) into real and imaginary parts, the following expressions are obtained for the i^{th} bus:

$$F_i = \left|\frac{\Delta s_i}{V_i}\right| \cos(-\varphi_i + \delta_i) - \sum_{j=1}^{n_b} |Y_{ij} V_j| \cos(\theta_{ij} + \delta_i) \quad (4)$$

$$M_i = \left|\frac{\Delta s_i}{V_i}\right| \sin(-\varphi_i + \delta_i) - \sum_{j=1}^{n_b} |Y_{ij} V_j| \sin(\theta_{ij} + \delta_i) \quad (5)$$

For treating the PV buses, the reactive power mismatch may be included as a variable [4, 9], therefore, an additional equation must be added to the system formed by (4) and (5) as follows [5]:

$$H_i = -\sum_{j=1}^{n_b} |Y_{ij} V_i V_j| \sin(\theta_{ij} - \delta_i + \delta_j) \quad (6)$$

The system (4)-(6) has n unknowns:

- The voltage magnitudes at PQ buses.
- The voltage angles at PQ and PV buses.
- The reactive power mismatches at PV buses.

For solving the system (4)-(6) using NR, the following iterative algorithm is considered:

$$\begin{bmatrix} \boldsymbol{\delta} \\ \boldsymbol{V} \\ \boldsymbol{Q} \end{bmatrix}_{k+1} = \begin{bmatrix} \boldsymbol{\delta} \\ \boldsymbol{V} \\ \boldsymbol{Q} \end{bmatrix}_{k} - \begin{bmatrix} \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{\delta}} & \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{V}} & \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{Q}} \\ \frac{\partial \boldsymbol{M}}{\partial \boldsymbol{\delta}} & \frac{\partial \boldsymbol{M}}{\partial \boldsymbol{V}} & \frac{\partial \boldsymbol{M}}{\partial \boldsymbol{Q}} \\ \frac{\partial \boldsymbol{H}}{\partial \boldsymbol{\delta}} & \frac{\partial \boldsymbol{H}}{\partial \boldsymbol{V}} & \frac{\partial \boldsymbol{H}}{\partial \boldsymbol{Q}} \end{bmatrix}_{k}^{-1} \begin{bmatrix} \boldsymbol{F} \\ \boldsymbol{M} \\ \boldsymbol{H} \end{bmatrix}_{k} \rightarrow \boldsymbol{x}_{k+1} = \boldsymbol{x}_k - [\nabla_{\boldsymbol{x}} \boldsymbol{g}(\boldsymbol{x}_k)]^{-1} \boldsymbol{g}(\boldsymbol{x}_k) \quad (7)$$

where, the subindexes denote the iteration counter; $\boldsymbol{Q} \in \mathbb{R}^{n_{PV}}$ is the vector of nodal reactive power at PV buses (where n_{PV} is the number of PV buses); $[\cdot]^{-1}$ is the inverse operator; and the elements of the Jacobian matrix $\nabla_{\boldsymbol{x}} \boldsymbol{g} \in \mathbb{R}^{n \times n}$ can be found in [5]. For further details about the CIF in polar coordinates, one can also consult [40].

2.1.- Ill-conditioned cases

Ill-conditioning of PF equations may be due to high loading level and/or large R/X ratios among other factors [37]. Typically, the degree of ill-conditioning is measured by the so-called condition number, which is defined as follows [41]:

$$\mathrm{CN} = \|\nabla_{\boldsymbol{x}}\boldsymbol{g}\|_{\mathrm{F}} \times \|[\nabla_{\boldsymbol{x}}\boldsymbol{g}]^{-1}\|_{\mathrm{F}} \tag{8}$$

where, $\|\cdot\|_{\mathrm{F}}: \mathbb{R}^{n\times n} \rightarrow \mathbb{R}^{+}$ is the Frobenius norm which may be calculated in (8) as follows:

$$\|\nabla_{\boldsymbol{x}}\boldsymbol{g}\|_{\mathrm{F}} = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{n}\left|a_{ij}\right|^{2}} \tag{9}$$

where, $a_{ij} \in \mathbb{R}$ is the ij^{th} element of the Jacobian matrix. Indeed, as the condition number grows, the degree of ill-conditioning also grows. While (8) provides an estimation of the degree of ill-conditioning of a system, it does not actually determine if a case is well or ill-conditioned. In this paper, although the condition number is still used for illustrating the conditioning of a system, a case is determined to be ill-conditioned according to the definition 1 [30].

Definition 1. *Ill-conditioned system:* a power system case will be called ill-conditioned (or ill-posed) if, despite its solution exists, NR fails for reaching it from a flat start (all load voltage magnitudes equal to 1 and all voltage angles equal to 0).

By the definition 1, the ill-conditioned cases are intimately linked with the initialization of the iterative procedure. It is well-known that NR has local convergence features, i.e. it fails if the starting guess lies outside of the so-called Region of Attraction (see [30] for further explanation).

3.- The developed PF solver

This section presents a novel PF solver which is able to efficiently managing with the set of equations (5) when they are ill-conditioned.

3.1.- Formulation

As previously commented, the regularization methods are widely used for managing with ill-conditioned nonlinear problems. The Tikhonov [41] and Lavrentiev's schemes [42] are undoubtedly the most popular regularization mechanism. While the former is more stable and produces better condition numbers, the latter is more efficient. Due to that, the Lavrentiev scheme is used in the developed PF solver. The Lavrentiev's regularization modifies the Jacobian matrix by adding some elements on its diagonal as follows:

$$\widetilde{\nabla_{\boldsymbol{x}}\boldsymbol{g}} = \nabla_{\boldsymbol{x}}\boldsymbol{g} + \alpha\boldsymbol{I} \tag{10}$$

where, $\boldsymbol{I} \in \mathbb{R}^{n\times n}$ is the identity matrix and $\alpha \in \mathbb{R}^{+}$ is the so-called damping factor. Other regularization matrices instead of the identity matrix can be found in the literature [43], however, the identity matrix preserves the sparsity pattern of the Jacobian matrix and it is considered as the most standard approach.

Regarding the damping factor, this parameter should be defined as a series $\{\alpha\}_{\infty} \to 0$. In this point, one should note that (10) is equal to $\nabla_{\boldsymbol{x}}\boldsymbol{g}$ for $\alpha = 0$. This is important since the obtained results may be inaccurate if the damping factor is not closely to zero [36]. On the other hand, at least at the beginning of the iterative procedure, the damping factor should be greater than zero in order to properly ensure the continuity of the operator (10). Keeping this in mind, the damping factor may be updated each iteration as follows [38]:

$$\alpha = q^{k}\alpha \tag{11}$$

where $q \in (0,1)$.

The new proposal poses a recursive version of a two-step damping Lavrentiev method as follows:

$$\boldsymbol{y}_k^l = \boldsymbol{x}_k^{l-1} - h\left[\widetilde{\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}_k^0)}\right]^{-1}\boldsymbol{g}\left(\boldsymbol{x}_k^{l-1}\right) \tag{12a}$$

$$\boldsymbol{x}_k^l = \boldsymbol{y}_k^l - h\left[\widetilde{\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}_k^0)}\right]^{-1}\boldsymbol{g}\left(\boldsymbol{y}_k^l\right) \text{ for } l = 1,2,\dots,N \tag{12b}$$

where, $h \in \mathbb{R}^+$ is the step size. As observed, equation (12) defines a loop for $j = 1,2,\dots,N$ where $N \in \mathbb{N}$. Within this loop, $\boldsymbol{x}_k^0 = \boldsymbol{x}_k$ while the updated state vector would be given by $\boldsymbol{x}_{k+1} = \boldsymbol{x}_k^N$. Regarding the step size, it is recommended initializing it with a sufficiently small value, for example, $h = 0.5$. Posteriorly, it can be updated using a line search technique as follows [44]:

$$A_{red} = \|\boldsymbol{g}(\boldsymbol{x}_k)\|_\infty^2 - \|\boldsymbol{g}(\boldsymbol{x}_{k+1})\|_\infty^2 \tag{13}$$

$$P_{red} = \|\boldsymbol{g}(\boldsymbol{x}_k)\|_\infty^2 - \left\|\boldsymbol{g}(\boldsymbol{x}_k) + \left[\nabla_{\boldsymbol{x}}\widetilde{\boldsymbol{g}(\boldsymbol{x}_k)}\right]^\dagger \boldsymbol{g}(\boldsymbol{x}_k)\right\|_\infty^2 \tag{14}$$

$$r = {}^{A_{red}}\!/_{P_{red}} \tag{15}$$

$$h = \begin{cases} \min(\sigma h, h_{\max}), & \text{if } r < \rho \\ h, & \text{if } r \geq \rho \end{cases} \tag{16}$$

where, $[\cdot]^\dagger$ is the adjoint operator, $\sigma > 1$, $h_{\max} \in \mathbb{R}^+$ is the maximum step size and $\rho \in \mathbb{R}^+$. The parameters q and α are strongly related to (9), thus, while q is taken constant, α evolves with this parameter & the iteration counter k and it determines the influence of the identity matrix in (8). These two parameters are not directly related to h in (10), which is only determined by r, σ, ρ and $h_{\max}$. One can assert that the iterative procedure is correctly evolving if $r < \rho$, therefore it is suitable to enlarge the step size in order to accelerate the convergence. The step size should be also updated within the loop (12). In this regard, the indexes (13) and (14) are calculated as follows:

$$A_{red} = \|\boldsymbol{g}(\boldsymbol{x}_k)\|_\infty^2 - \left\|\boldsymbol{g}\left(\boldsymbol{y}_k^l\right)\right\|_\infty^2 \tag{17}$$

$$P_{red} = \|\boldsymbol{g}(\boldsymbol{x}_k)\|_\infty^2 - \left\|\boldsymbol{g}\left(\boldsymbol{y}_k^l\right) + \left[\nabla_{\boldsymbol{x}}\widetilde{\boldsymbol{g}(\boldsymbol{x}_k)}\right]^\dagger \boldsymbol{g}\left(\boldsymbol{y}_k^l\right)\right\|_\infty^2 \tag{18}$$

On the other hand, the reader should note that the larger N provides the faster convergence. It is founded in the fact that the loop (12) is, actually, a multistep algorithm (one should not confuse the steps in (12) with the iterations denoted by k). This kind of

techniques present higher convergence rate as more steps are computed [45]. However, one should be careful since the higher convergence rate may imply the lower degree of robustness and stability [46]. Therefore, it seems logical to initialize $N = 1$ and update it each iteration as follows:

$$N = \begin{cases} N + 1, & \text{if } r < \rho \\ N, & \text{if } r \geq \rho \end{cases} \tag{19}$$

In order to summarize, the proposed algorithm is outlined in Fig. 1, while it is fully described in Algorithm 1 using pseudocode. In this paper, $\|\boldsymbol{g}\|_\infty$ is considered as convergence criterion while the algorithm also stops if the iteration counter surpasses $k_{\max}$. This latter situation normally indicates divergence. It is also worth mentioning that the developed algorithm only requires a LU decomposition each iteration, which supposes a competitive computational burden compared with the NR.

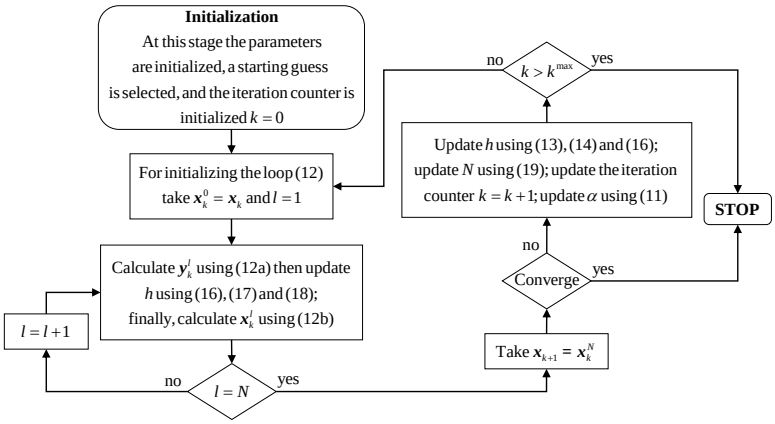

Fig. 1. Flowchart of the developed PF solver

Algorithm 1: Main steps of the developed solver

Let $\boldsymbol{x}_0$, ε, $k_{\max}$, h, $h_{\max}$, σ, ρ, α, N and q be given
Initialize iteration counter $k \leftarrow 0$
while $\|\boldsymbol{g}(\boldsymbol{x}_k)\|_\infty > \varepsilon$ **do**
    $j \leftarrow 1$
    $\boldsymbol{x}_k^l \leftarrow \boldsymbol{x}_k$
    for $l \leq N$ **do**
        $\boldsymbol{y}_k^l \leftarrow$ solve (12a)
        $h \leftarrow$ solve (16) *# using (17) and (18)*
        $\boldsymbol{x}_k^l \leftarrow$ solve (12b)
    end do
    $\boldsymbol{x}_{k+1} \leftarrow \boldsymbol{x}_k^N$
    if $\|\boldsymbol{g}(\boldsymbol{x}_{k+1})\|_\infty < \varepsilon$
        break *#Convergence*
    else
        $h \leftarrow$ solve (16) *# using (13) and (14)*
        $N \leftarrow$ solve (19)
        $k \leftarrow k + 1$
        if $k > k_{\max}$
            break *#Fail*
        end
        $\alpha \leftarrow$ solve (11)
    end
return solution $\boldsymbol{x}_k$

3.2.- Fixed points of the developed method

The following theorem is devoted to proving that the PF solution is a fixed point of the developed PF solver.

Theorem 1. *Fixed Points of the developed method:* let $\boldsymbol{x}^\star$ be a PF solution (i.e. $\boldsymbol{x}^\star$ such that $\boldsymbol{g}(\boldsymbol{x}^\star) = \boldsymbol{0}$). Then, it is a fixed point of the developed method defined by the Algorithm 1.

Proof. By evaluating (10) at $\boldsymbol{x}_k^0 = \boldsymbol{x}^\star$ for $j = 1$ one obtains:

$$\begin{cases} \boldsymbol{y}_k^1 = \boldsymbol{x}^\star - h\left[\nabla_{\boldsymbol{x}}\widetilde{\boldsymbol{g}(\boldsymbol{x}^\star)}\right]^{-1}\boldsymbol{g}(\boldsymbol{x}^\star) = \boldsymbol{x}^\star \\ \boldsymbol{x}_k^1 = \boldsymbol{x}^\star - h\left[\nabla_{\boldsymbol{x}}\widetilde{\boldsymbol{g}(\boldsymbol{x}^\star)}\right]^{-1}\boldsymbol{g}(\boldsymbol{x}^\star) = \boldsymbol{x}^\star \end{cases} \tag{20}$$

It can be easily checked that for any value of j:

$$\boldsymbol{y}_k^j = \boldsymbol{x}_k^j = \boldsymbol{x}^\star \tag{21}$$

Therefore:

$$\boldsymbol{x}_k^N = \boldsymbol{x}_{k+1} = \boldsymbol{x}^\star \tag{22}$$

which completes the proof. □

3.2.- Stability of the developed method

In this section, the numerical stability of the developed method is proved. Firstly, it is suitable to introduce the following definition.

Definition 1. *Type-m equilibrium point:* let $\boldsymbol{\omega}$ be the dynamic function associated with a iterative procedure defined as $\boldsymbol{\omega} = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k$. Let also be $\boldsymbol{x}^\star$ and equilibrium point of $\boldsymbol{\omega}$. Then, it is said that $\boldsymbol{x}^\star$ is a type-m equilibrium point for $\boldsymbol{\omega}$, if and only if the Jacobian matrix associated with $\boldsymbol{\omega}$ at $\boldsymbol{x}^\star$ has exactly m eigenvalues with a positive real part.

It is well-known that the type-0 equilibrium points are asymptotically stable [30], therefore, one expects from the PF solution to be a type-0 equilibrium point of the developed method. The following Theorem discusses this aspect.

Theorem 2. *Stability of the developed method:* let $\boldsymbol{x}^\star$ be a PF solution (i.e. $\boldsymbol{x}^\star$ such that $\boldsymbol{g}(\boldsymbol{x}^\star) = \boldsymbol{0}$). Then, it is a type-0 equilibrium point for the developed method described in Algorithm 1 if $\alpha = 0$.

Proof. The dynamic function associated with the developed method is:

$$\boldsymbol{\omega} = \boldsymbol{x}_{k+1} - \boldsymbol{x}_k = \boldsymbol{x}_k^N - \boldsymbol{x}_k^0 = -\sum_{j=1}^{N} h[\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}_k^0) + \alpha\boldsymbol{I}]^{-1}\boldsymbol{g}(\boldsymbol{y}_k^j) \tag{23}$$

Since $\boldsymbol{y}_k^j\big|_{\boldsymbol{x}^\star} = \boldsymbol{x}_k^j\big|_{\boldsymbol{x}^\star} = \boldsymbol{x}^\star$:

$$\boldsymbol{\omega}|_{\boldsymbol{x}^\star} = -Nh\left[\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}^\star) + \alpha\boldsymbol{I}\right]^{-1}\boldsymbol{g}(\boldsymbol{x}^\star) \tag{24}$$

By differentiating (24) w.r.t. $\boldsymbol{x}$, one obtains:

$$\nabla_{\boldsymbol{x}}\boldsymbol{\omega}|_{\boldsymbol{x}^\star} = -Nh\left(\nabla_{\boldsymbol{x}}\left[\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}^\star) + \alpha\boldsymbol{I}\right]^{-1}\boldsymbol{g}(\boldsymbol{x}^\star) + \left[\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}^\star) + \alpha\boldsymbol{I}\right]^{-1}\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}^\star)\right) \tag{25}$$

Solution of (25) is not trivial, however, one can usually affirm that at $\boldsymbol{x}^{\star}$ one has $\alpha \approx 0$, therefore:

$$\nabla_{\boldsymbol{x}}\mathbf{G}|_{\boldsymbol{x}^{\star},\alpha=0} = -Nh\left(\nabla_{\boldsymbol{x}}\left[\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}^{\star})\right]^{-1}\mathbf{g}(\boldsymbol{x}^{\star}) + \left[\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}^{\star})\right]^{-1}\nabla_{\boldsymbol{x}}\boldsymbol{g}(\boldsymbol{x}^{\star})\right) = -Nh\boldsymbol{I} \quad (26)$$

which completes the proof. □

3.3.- A strategy for handling equipment limits

In this subsection, a simple strategy for taking into account the generators' reactive power limits is presented. The developed mechanism is based on a logic strategy which described with the following steps:

Step 1: solve the PF problem using the developed algorithm and obtain the solution $\boldsymbol{x}_k$. Save the values $\bar{N} = N$ and $\bar{\alpha} = \alpha$.

Step 2: check if any generator has violated some reactive limit. In affirmative case go to step 3, otherwise, return $\boldsymbol{x}_k$ as solution.

Step 3: create $\mathcal{Q}^{+}$ a set of indexes which the upper reactive limit has been hit. create $\mathcal{Q}^{-}$ a set of indexes which the lower reactive limit has been hit.

Step 4: convert those buses $\in \mathcal{Q}^{+}, \mathcal{Q}^{-}$ to PQ. If there are not PV buses, declare the problem infeasible, otherwise go to Step 5.

Step 5: for those buses $\in \mathcal{Q}^{+}$ take $Q_i^{sp} = Q_i^{\max}$, where $Q_i^{\max}$ is the upper reactive bound at i^{th} bus. For those buses $\in \mathcal{Q}^{-}$ take $Q_i^{sp} = Q_i^{\min}$, where $Q_i^{\min}$ is the lower reactive bound at i^{th} bus.

Step 6: take $N = \bar{N}$, $\alpha = \bar{\alpha}$ and $\boldsymbol{x}_0 = \boldsymbol{x}_k$. Go to Step 1.

The idea behind the developed strategy is to take advantage of the information obtained at previous PF solution. Indeed, the obtained solution is used for initializing the following iterative procedure. It is expected that $\boldsymbol{x}_k$ provides a better approximation to

$x^{\star}$. This facilitates the convergence, which allows us to use the previous values of N and α with the aim of speeding up the convergence of the following PF solutions.

It is worth mentioning that, although the described steps are focused on the generators' reactive limits, the developed strategy is easily extensible to other equipment limits like transformers tap changers.

4.- Numerical results

In this section, several numerical results are presented in the aim of achieving the targets of this work.

4.1.- On the ill-conditioning of PF equations in current injection formulation

At this point, it is discerned if the PF equations are better conditioned in power or current mismatch formulation. To do that, the condition number (8) in the snapshots of the Polish Transmission system has been analyzed [47]. Fig. 2 shows a comparison of the condition numbers for these systems using both tested formulations with a flat start. As observed, the condition numbers are notably higher in the case of current mismatches formulation.

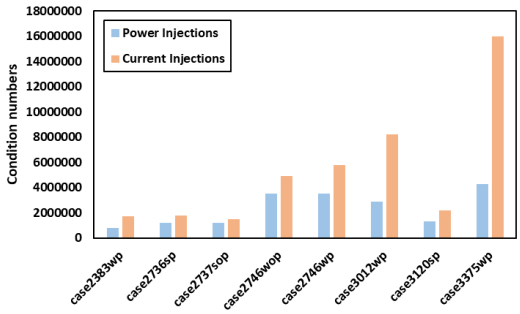

Fig. 2. A comparison between the condition numbers with a flat start for the snapshots of the Polish transmission systems using power and current mismatch formulations.

Now, let us analyse how the R/X ratio affects the ill-conditioning of both tested formulations. To do that, the branches resistances have been multiplied by different factors $\mu \in \mathbb{R}^{+}$. Results of this scenario with a flat start for the *case2736sp* are shown in

Fig. 3. As in preceding simulation, the condition numbers are higher in the case of the current injection formulation. In addition, it is worth noting how the PF equations in current injections form are more sensitive to the R/X ratio.

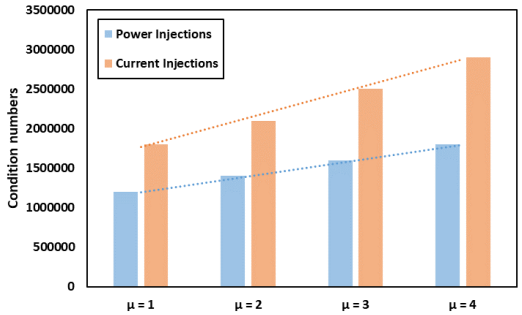

Fig. 3. A comparison between the condition numbers with a flat start for the *case2736sp* using power and current mismatch formulations at different R/X ratios. The dotted lines indicate the trending of each series.

As observed from the results of Fig. 2 and 3, the PF equations tend to be worse conditioned when they are posed in current injections form. For this reason, some robust PF solvers devoted to be used with power mismatch formulation may fail when they are raised in current injections form, as demonstrated in the following section.

For further show how the formulation affects the performance of a PF solver, let us consider the solution of the well-known IEEE 30-, and 118-bus cases [48] using NR and the developed solver in both power and current injection formulations. Fig. 4 shows the convergence profiles of these solvers for solving the studied systems from a flat start. In this case, the branches resistances have been multiplied by 2. As observed, the convergence properties of the NR are deteriorated when CIF is considered, while the developed technique is well-behaved with both formulations.

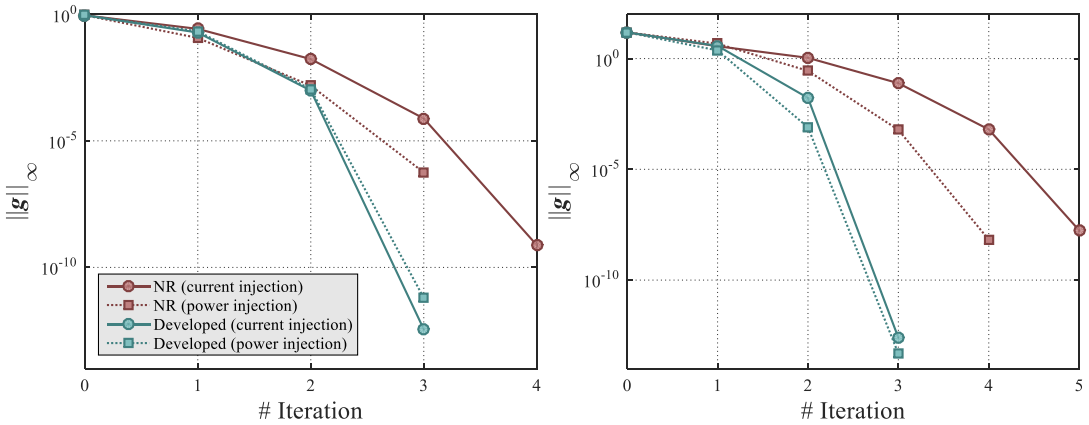

Fig. 4 Convergence profiles for IEEE 30-bus (left) and 118-bus (right) cases with branch resistances multiplied by 2 (i.e. $\mu = 2$)

4.2.- A comparison of different PF solvers in CIF

Next, various PF solvers in current injection formulation are compared. To do that, various realistic cases based on synthetic grids [49], the Polish Transmission System [47] and the French Transmission system [47] are considered. The following methods in CIF are considered hereinafter:

- Conventional NR.
- Second order power flow method (SOPF) with optimal multiplier in [24].
- High Order Levenberg-Marquardt method (HOLM) [35].
- Reverse-Bulirsch-Stoer method (RBS) [30].
- Implicit Continuous Newton's method using the Backward-Euler method (BEM) [32]. The implementation of this solver labelled as ICNM-J1 in [32] is considered for simulations.
- The developed PF solver described in Algorithm 1.

All tested techniques have been coded in Matpower v7.0 [39], following the guidelines provided in their respective references. For the developed technique, it is considered that $h_{\max} = 1$, $\sigma = 1.2$, $q = 0.2$ [38] and $\rho = 10^{-4}$ [44]. All simulations have been

initialized using a flat start, $\varepsilon = 10^{-6}$ and $k_{\max} = 200$. For the developed method, $h = 0.5$, $N = 1$ and $\alpha = 0.1$, are initially considered.

4.2.1.- Convergence rates for base cases

The total number of iterations is typically used as an index for comparing the convergence rates of different methods. Table 1 provides the total iterations of the tested techniques for base cases. As observed, the NR only successfully solved the *case_ACTIVSg500*, *case2383wp*, *case2737sop* and *case3120sp*. On the other hand, SOPF failed in the *case3012wp*, *case3375wp* and *case6468rte*; while the RBS failed in the *case3375wp* and *case6468rte*. Some of the cases where NR failed are well-conditioned in power mismatches form (see results in [50]). On the other hand, the RBS solved the *case3375wp* without any problem in power mismatches form [31]. These two points reaffirm the conclusions extracted in the previous section. On the other hand, the HOLM, BEM and the developed method successfully solved all the considered systems. As observed, the developed method required less iterations than the remainder techniques.

Table 1. Total iterations for base cases

System	NR	SOPF	HOLM	RBS	BEM	Developed
case_ACTIVSg500	5	7	5	12	12	3
case2383wp	5	6	13	12	12	3
case2736sp	Fail	11	12	20	14	3
case2737sop	5	10	8	14	13	3
case2746wop	Fail	11	11	14	13	3
case2746wp	Fail	11	10	14	13	3
case3012wp	Fail	Fail	12	14	13	3
case3120sp	8	10	11	14	13	3
case3375wp	Fail	Fail	11	Fail	15	3
case6468rte	Fail	Fail	11	Fail	13	3

The convergence profiles for the *case2736sp* are plotted in Fig. 4. It is observed that the developed technique has superior convergence properties compared with the remainder studied methods. The developed technique produced the lowest residual after 1st iteration. It is worth noting that, in this case, NR showed an oscillatory pattern.

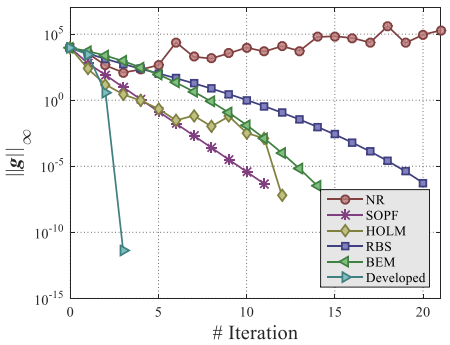

Fig. 4. Convergence profiles for the *case2736sp*.

4.2.2.- Convergence rates with generators' reactive limits

In this subsection, the ability of the tested techniques for managing with the generators' reactive limits is explored. To consider this point within the proposed technique, the strategy described in Section 3.3, has been considered. For remainder methodologies, the mechanism described in [51], has been included. Table 2 presents the total number of iterations in case of considering the generators' reactive limits. Similar conclusions as for the base case scenario is observed.

Table 2. Total iterations with reactive limits

System	NR	SOPF	HOLM	RBS	BEM	Developed
case_ACTIVSg500	9	19	9	31	31	5
case2383wp	13	27	25	45	43	7
case2736sp	Fail	48	18	47	41	6
case2737sop	10	25	12	40	38	6
case2746wop	Fail	37	23	56	53	8
case2746wp	Fail	34	18	51	49	7
case3012wp	Fail	Fail	16	33	32	5
case3120sp	16	37	23	56	53	7
case3375wp	Fail	Fail	15	Fail	31	6
case6468rte	Fail	Fail	16	Fail	34	5

4.2.3.- Convergence rates for limit case

In this subsection, the performance of the tested methods under stressed heavy conditions is studied. To do that, the nodal injected active and reactive powers are modified as follows:

$$P_i^{sp} = \lambda P_i^{sp} \tag{27}$$

$$Q_i^{sp} = \lambda Q_i^{sp}, \text{for PQ buses} \quad (28)$$

where, $\lambda \in \mathbb{R}^+$ is the loading level; $P_i^{sp} \in \mathbb{R}$ and Q_i^{sp} are the specified active and reactive power injected at i^{th} bus, respectively. This factor is progressively increased until all considered methods fail, taking the immediately lower value. For example, for the *case3120sp* a loading factor $\lambda = 1.4036$ pu has been taken since $\lambda = 1.4037$ pu gives rise divergence. Table 3 provides the total iterations under these conditions. In this case, the NR failed in all tested cases. The SOPF failed in the *case3012wp*, *case3375wp* and *case6468rte*. The RBS failed in the *case_ACTIVSg500*, *case2383wp*, *case3012wp*, *case3375wp* and *case6468rte*; while it converged to the low voltage solution in the *case2736sp*. The BEM failed in the *case_ACTIVSg500*, *case2746wp*, *case3375wp* and *case6468rte*; it also converged to the low voltage value in the *case2383wp*, *case2736sp* and *case3120sp*. The HOLM and the developed technique converged to the high voltage value in all cases. Once again, the developed technique required less iterations than the remainder approaches.

Table 3. Total iterations for limit cases

System	λ [pu]	NR	SOPF	HOLM	RBS	BEM	Developed
case_ACTIVSg500	1.4505	Fail	8	11	Fail	Fail	6
case2383wp	1.1411	Fail	7	18	Fail	14*	6
case2736sp	1.6621	Fail	12	39	20*	14*	5
case2737sop	1.9063	Fail	11	22	14	15	9
case2746wop	1.4218	Fail	11	21	14	15	6
case2746wp	1.2988	Fail	11	19	14	Fail	6
case3012wp	1.1765	Fail	Fail	24	Fail	15	7
case3120sp	1.4036	Fail	11	20	14	14*	7
case3375wp	1.1166	Fail	Fail	26	Fail	Fail	5
case6468rte	1.0811	Fail	Fail	18	Fail	Fail	5

* *Low voltage solution*

4.2.4.- Solution times

The computational efficiency of a PF algorithm is typically measured by its solution time. However, this aspect depends on multiple factors such as the machine used, the language employed, etc. In that sense, all simulations have been run under Matlab R2014b

on a 64-bit i5-8500F Intel Core personal computer (3 GHz, 8 GB of RAM). Tables 4-6 provide the solution times in milliseconds corresponding to the results provided in Tables 1-3, respectively. It is worth mentioning that all reported solution times have been obtained as the average value of 100 simulations. As expected, the developed method was faster than the remainder techniques, since it presents a lower computational burden compared with HOLM and RBS. On the other hand, it outperformed NR, SOPF and BEM due to its higher convergence rate.

Table 4. Solution times [ms] for base cases

System	NR	SOPF	HOLM	RBS	BEM	Developed
case_ACTIVSg500	17.1	24.5	36.3	118.3	43.4	11.6
case2383wp	75.3	95.4	465.0	609.1	194.3	61.2
case2736sp	--	196.9	506.9	1066.5	249.9	68.2
case2737sop	85.2	181.0	353.7	773.7	231.4	68.2
case2746wop	--	187.0	482.0	793.4	236.3	69.4
case2746wp	--	187.0	437.8	807.4	241.3	69.3
case3012wp	--	--	604.7	881.4	260.3	74.9
case3120sp	153.6	203.0	568.1	905.1	269.1	77.0
case3375wp	--	--	691.6	--	326.1	82.2
case6468rte	--	--	1117.4	--	529.3	126.8

Table 5. Solution times [ms] with generator' reactive limits

System	NR	SOPF	HOLM	RBS	BEM	Developed
case_ACTIVSg500	32.8	66.5	67.6	309.1	110.3	18.5
case2383wp	201.5	442.8	855.1	2312.2	690.0	157.1
case2736sp	--	873.6	789.3	2541.4	738.1	151.0
case2737sop	175.7	455.0	545.6	2258.8	677.5	150.0
case2746wop	--	658.6	1006.1	3248.6	963.5	203.6
case2746wp	--	605.2	800.0	2974.1	880.3	178.7
case3012wp	--	--	793.8	2096.2	633.6	135.8
case3120sp	316.3	754.8	1185.7	3649.7	1078.7	199.4
case3375wp	--	--	962.9	--	680.0	180.3
case6468rte	--	--	1645.7	--	1381.8	211.6

Table 6. Solution times [ms] for limit cases

System	NR	SOPF	HOLM	RBS	BEM	Developed
case_ACTIVSg500	--	28.0	74.9	--	--	22.5
case2383wp	--	166.3	623.6	--	226.3	131.8
case2736sp	--	214.8	1755.3	1066.5	249.9	118.4
case2737sop	--	258.1	1011.4	773.7	267.3	205.6
case2746wop	--	187.0	955.0	793.4	274.7	144.3
case2746wp	--	189.0	857.9	807.4	--	149.5
case3012wp	--	--	1235.5	--	301.0	185.5
case3120sp	--	223.3	1079.4	905.1	291.5	198.5
case3375wp	--	--	1721.6	--	--	142.7
case6468rte	--	--	1836.0	--	--	215.2

5.- Conclusions and future works

This paper has been devoted to study some relevant aspects related with the PF solution of ill-conditioned cases using the current injection formulation. In this regard, a novel robust technique using this formulation has been developed. The fixed points of developed technique along with its stability have been proved.

Various numerical results in several large-scale realistic cases have been provided. Firstly, it was analyzed the conditioning of the PF equations in CIF and the conventional power mismatches form. From this analysis, it is deduced that the PF equations tend to be worse conditioned in CIF. Secondly, several conventional and modern robust PF solvers in current injection formulation have been compared. It has been checked that some of available robust PF solvers present an unacceptable performance in current injection formulation. Here, results obtained with the proposed technique were very promising.

As previously commented, solving the PF problem in current injection formulation brings some computational savings, however, it has been shown the instability of this formulation. Consequentially, the developed technique along with a variety of future robust PF techniques should be validated in current injections form.

References

[1] Saadat H. Power system analysis. McGraw-Hill; 2004.

[2] Karimi M, Shahriari A, Aghamohammadi MR, Marzooghi H, Terzija V. Application of Newton-based load flow methods for determining steady- state condition of well and ill-conditioned power systems: A review. Int J Elect Power Energy Syst 2019; 113: 298-309.

[3] da Costa VM, Martins N, Pereira JLR. Developments in the Newton Raphson power flow formulation based on current injections. IEEE Trans Power Syst 1999; 14(4): 1320-26.

[4] da Costa VM, Pereira JLR, Martins N. An augmented Newton–Raphson power flow formulation based on current injections. Int J Elect Power Energy Syst 2001; 23(4): 305-12.

[5] Kulworawanichpong T. Simplified Newton-Raphson power-flow solution method. Int J Elect Power Energy Syst 2010; 23(6): 551-8.

[6] El-Hawary ME. Electrical power systems: design and analysis. Revised printing, vol. 2. John Wiley & Sons; 1995.

[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] Gómez-Expósito A, Romero-Ramos E, Dzafic I. Hybrid real-complex current injection-based load flow formulation. Elect Power Syst Res 2015; 119: 237-46.

[9] 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 Delivery 2004; 19(2): 894-6.

[10] Oliveira CC, Bonini Neto A, Minussi CR, Alves DA, Castro CA. New representation of PV buses in the current injection Newton power flow. Int J Elect Power Energy Syst 2017; 90: 237-44.

[11] Dommel HW, Tinney WF, Powell WL. Further developments in Newton'smethod for power system applications. In: IEEE winter power meeting, conference paper no. 70 CP 161-PWR. New York; 1970.

[12] Stott B. Review of Load Flow Calculation Methods. Proc IEEE 1974; 62: 916-29.

[13] Vinkovic A, Suhadolc M, Mihalic R. Current-based models of FACTS devices for three-phase load-flow calculations using the Newton-Raphson method. Int J Elect Power Energy Syst 2013; 45(1): 117-28.

[14] Garcia PAN, Pereira JLR, Carneiro S, da Costa VM, Martins N. Three-phase power flow calculations using the current injection method. IEEE Trans Power Syst 2000; 15(2): 508-14.

[15] Penido DRR, de Araujo LR, Carneiro S, Pereira JLR, Garcia PAN. Three-Phase Power Flow Based on Four-Conductor Current Injection Method for Unbalanced Distribution Networks. IEEE Trans Power Syst 2008; 23(2): 494-503.

[16] Penido DRR, de Araujo LR, Carneiro S, Pereira JLR. A new tool for multiphase electrical systems analysis based on current injection method. Int J Elect Power Energy Syst 2013; 44(1): 410-20.

[17] Alam MJE, Muttaqi KM, Sutanto D. A Three-Phase Power Flow Approach for Integrated 3-Wire MV and 4-Wire Multigrounded LV Networks With Rooftop Solar PV. IEEE Trans Power Syst 2013; 28(2): 1728-37.

[18] Alves GO, Pereira JLR, La Gatta PO, Passos Filho JA, Tomim MA. A new governor power flow formulation based on the current injections method. Int J Elect Power Energy Syst 2019; 104: 705-15.

[19] Stott B, Alsac O. Fast Decoupled Load Flow. IEEE Trans Power Appar Syst 1974; PAS-93(3): 859-69.

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

[21] Iwamoto S, Tamura Y. A load flow calculation method for ill-conditioned power systems. IEEE Trans Power Appar Syst 1981; PAS-100: 1736-43.

[22] F. Milano. Power System Modelling and Scripting. New York: Springer; 2010.

[23] Tate JE, Overbye TJ. A comparison of the optimal multiplier in polar and rectangular coordinates. IEEE Trans Power Syst 2005; 20(4): 1667-74.

[24] Ferreira CA, da Costa VM. A second order power flow based on current injection equations. Int J Elect Power Energy Syst 2005; 27(4): 254-63.

[25] Sheng H, Chiang H. CDFLOW: A Practical Tool for Tracing Stationary Behaviors of General Distribution Networks. IEEE Trans Power Syst 2014; 29(3): 1365-71.

[26] Yang X, Zhou X. Application of asymptotic numerical method with homotopy techniques to power flow problems. Int J Elect Power Energy Syst 2014; 57: 375-83.

[27] Davoodi E, Hagh MT, Zadeh SG. A hybrid Improved Quantum-behaved Particle Swarm Optimization-Simplex method (IQPSOS) to solve power system load flow problems. Appl Soft Comput 2014; 21: 171-9.

[28] Neuberger JW. Continuous Newton's Method. In: Neuberger JW, editor. Sobolev Gradient and Differential Equations, 2nd ed, vol. 1670. Berlin: Springer-Verlag; 2010, p. 79-83.

[29] Butcher JC. Numerical Methods for Ordinary Differential Equations. Hoboken, NJ: Wiley; 2003.

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

[31] Tostado-Véliz M, Kamel S, Jurado F. A Robust Power Flow Algorithm Based on Bulirsch-Stoer Method. IEEE Trans Power Syst 2019; 34(4): 3081-89.

[32] Milano F. Implicit Continuous Newton Method for Power Flow Analysis. IEEE Trans Power Syst 2019; 34(4): 3309-11.

[33] Xie N, Torelli F, Bompard E, Vaccaro A. Dynamic computing paradigm for comprehensive power flow analysis. IET Gener Transmiss Distrib 2013; 7(8): 832-42.

[34] Torelli F, Vaccaro A. A second order dynamic power flow model. Elect Power Syst Res 2015; 126: 12-20.

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

[36] 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.

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

[38] Liu W, Wu C. A predictor-corrector iterated Tikhonov regularization for linear ill-posed inverse problems. Appl Math Comput 2013; 221: 802-18.

[39] 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.

[40] Sereeter B, Vuik C, Witteveen C. On a comparison of Newton-Raphson solvers for power flow problems. J Comput Appl Math 2019; 360: 157-69.

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

[42] Argyros IK, George S. Regularization Methods for Ill-Posed Problems with Monotone Nonlinear Part. Punjab Univ J Math 2014; 46(1): 25-38.

[43] Huang G, Nochese S, Reichel L. Regularization matrices determined by matrix nearness problems. Lin Alg Appl 2016; 502: 41-57.

[44] Idema R, Lahaye DJP. Computational Methods in Power System Analysis. In: Schilders W, editor. Atlantis Studies in Scientific Computing in Electromagnetics, Paris, France: Atlantis Press; 2014.

[45] Petkovic MS, Neta B, Petkovic LD, Dzunic J. Multipoint Methods for Solving Nonlinear Equations. Cambridge, MA, USA: Academic Press; 2013.

[46] Ezquerro JA, Hernández MA. An optimization of Chebyshev's method. J Complexity 2009; 25: 343-61.

[47] Matpower's database, http://www.pserc.cornell.edu/matpower/ [accessed 8 March 2020].

[48] Peyghami S, Davari P, Fotuhi-Firuzabad M, Blaabjerg F. Standard Test Systems for Modern Power System Analysis: An Overview. IEEE Ind Electron Mag 2019; 13(4): 86-105.

[49] Birchfield AB, Xu T, Gegner K, Shetye KS, Overbye TJ. Grid Structural Characteristics as Validation Criteria for Synthetic Networks. IEEE Trans Power Syst 2017; 32(4): 3258-65.

[50] Tostado-Véliz M, Kamel S, Jurado F. A novel family of efficient Power-Flow methods with high convergence rate suitable for large realistic power systems. IEEE Syst J 2020.

[51] Tostado-Véliz M, Kamel S, Jurado F. Robust and efficient approach based on Richardson extrapolation for solving badly initialised/ill-conditioned power-flow problems. IET Gener Transmiss Distrib 2019; 13(16): 3524-33.