Continuous Newton's Method for Power Flow Analysis

Federico Milano, *Member, IEEE*

***Abstract*—This paper describes the application of the continuous Newton's method to the power flow problem. This method basically consists in formulating the power flow problem as a set of autonomous ordinary differential equations. Based on this formal analogy, we propose an entire family of numerically efficient algorithms for solving ill-conditioned or badly-initialized power flow cases. The paper also shows that the classical Newton–Raphson's method and most robust power flow techniques proposed in the literature are particular cases of the proposed formulation. An example based on a 1254-bus model of the UCTE system is presented and discussed.**

***Index Terms*—Continuous Newton's method, ill-conditioned power flow, Newton–Raphson's method, optimal multiplier, robust power flow, Runge–Kutta formulas.**

I. INTRODUCTION

A. Motivation

THE power flow analysis is one of the most important problems in power system studies. This paper gives a novel perspective on the formulation of the power flow problem and proposes an efficient method for solving ill-conditioned cases.

B. Literature Review

The origins of the formulation of the power flow problem and the solution based on the Newton–Raphson's technique are back to the late 1960s [1]. Since then, a huge variety of studies have been presented about the solution of the power flow problem, addressing starting initial guess [2], computational efficiency [3]–[9], ill-conditioned cases and robustness [10]–[18], multiple solutions [19], [20], and unsolvable cases [21], [22].

It is relevant to classify the power flow problems into the following categories:

1) *Well-conditioned case*. The power flow solution exists and is reachable using a flat initial guess (e.g., all load voltage magnitudes equal to 1 and all bus voltage angles equal to 0) and a standard Newton–Raphson's method. This case is the most common situation.
2) *Ill-conditioned case*. The solution of the power flow problem does exist, but standard solution methods fail to get this solution starting from a flat initial guess. Typically this situation is due to the fact that the region of attraction of the power flow solution is narrow or far away from the initial guess. In this case, the failure of standard power flow solution methods is due to the instability of the numerical method, not of the power flow equations. Robust power flow methods have proved to be efficacious for solving ill-conditioned cases.
3) *Bifurcation point*. The solution of the power flow exists but it is either a saddle-node bifurcation or a limit-induced bifurcation [23].
   a) Saddle-node bifurcations are associated with the maximum loading condition of a system. The solution cannot be obtained using standard or robust power flow methods, since the power flow Jacobian matrix is singular at the solution point.
   b) Limited-induced bifurcations are associated with a physical limit of the system, such as a shortage of generator reactive power. Although limit-induced bifurcation can in some cases lead to the voltage collapse of the system, the solution point is typically a well-conditioned case and does not show convergence issues.

   Several continuation techniques [23], [24] and optimal power flow problems [25]–[27] have been proposed for determining bifurcation points. These methods allow defining the distance between the present power flow solution and the bifurcation points and thus are useful for studying the static stability of the system [23]. However, encountering a case whose solution is exactly a bifurcation point is quite uncommon in the practice.
4) *Unsolvable case*. The power flow solution does not exist. Typically, the issue is that the loading level of the network is too high. As in the case of the bifurcation points, a continuation method or an optimal power flow problem allow defining the maximum loading level that the system can supply. An alternative method to analyze unsolvable cases is given in [21] and [22]. As shown in [21], robust power flow methods provide a solution close to the feasibility boundary rather than diverge.

C. Contributions

This paper addresses ill-conditioned power flow cases. At this aim, we propose a novel approach for formulating the power flow problem based on the vector continuous Newton's method (a scalar version of the continuous Newton's method can be found in [28]). This approach shows that there is a formal analogy between the Newton's method and a set of autonomous ordinary differential equations. This analogy is intriguing, since

Manuscript received April 04, 2008; revised June 09, 2008. First published November 18, 2008; current version published January 21, 2009. This work was supported in part by the Ministry of Science and Education of Spain through CICYT Project DPI2006-08001 and in part by Junta de Comunidades de Castilla-La Mancha, through project PCI-08-0102. Paper no. TPWRS-00257-2008.

The author is with the Department of Electrical Engineering, University of Castilla-La Mancha, Ciudad Real, Spain (e-mail: Federico.Milano@uclm.es).

Digital Object Identifier 10.1109/TPWRS.2008.2004820

it allows unifying the standard Newton–Raphson's method and most robust techniques proposed in the literature in an unique framework. Furthermore, the analogy suggests that any efficient numerical integration method (e.g., Runge–Kutta formulas) can be used for solving the power flow problem. Finally, a byproduct of the continuous Newton's method is the similarity with the Davidenko's homotopy technique [29].

D. Paper Organization

In Section II we provide brief outlines of the fundamentals of standard and robust Newton–Raphson's methods for solving the power flow problem. Section III describes the continuous Newton's method and its application to the power flow problem. Consideration on the stability and the similarity with the Davidenko's homotopy method are also given in this section. Section IV presents a case study based on a 1254-bus model of the UCTE system and shows the advantages of the proposed technique. Finally, in Section V conclusions are duly drawn.

II. Outlines of the Newton–Raphson's Method

The power flow problem is formulated as a set of nonlinear equations, as follows:

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

where $\boldsymbol{g}(\boldsymbol{g} \in \mathbb{R}^n)$ and $\boldsymbol{x}(\boldsymbol{x} \in \mathbb{R}^n)$ are the variables, i.e., voltage amplitudes and phases at load buses, reactive power and voltage phases at generator PV buses and active and reactive power at the slack bus [1]. In the classical power flow formulation, variables and equations are twice the number of the network buses (see also Appendix A).

Since (1) are nonlinear and cannot be explicitly inverted, one has to use a numerical iterative technique for solving the power flow problem. The ith iteration of the classical Newton–Raphson (NR) algorithm for (1) is as follows:

$$\begin{aligned}\Delta\boldsymbol{x}^{(i)} &= -\,[\boldsymbol{g}_{\boldsymbol{x}}^{(i)}]^{-1}\boldsymbol{g}^{(i)}\\ \boldsymbol{x}^{(i+1)} &= \boldsymbol{x}^{(i)} + \Delta\boldsymbol{x}^{(i)}\end{aligned} \tag{2}$$

where $\boldsymbol{g}_{\boldsymbol{x}} = \nabla_{\boldsymbol{x}}^T\boldsymbol{g}$ is the Jacobian matrix of the power flow equations. A *good* initial guess $\boldsymbol{x}^{(0)}$ is needed to start the iterative process. Typically a flat start is an acceptable initial guess [2]. The algorithm stops if the variable increments $\Delta\boldsymbol{x}$ are lower than a given tolerance ϵ or the number of iterations is greater than a given limit ($i > i_{\max}$). In the latter case, the algorithm has likely failed to converge.

For well-conditioned cases, the standard NR technique typically converges in four to five iterations. However, there are idiosyncratic cases for which the NR technique will fail to converge. For this reason a variety of *robust* variations of the basic NR method have been proposed in the literature [10], [11], [13], [15]–[18]. Most of these techniques mainly consists in modifying the first equation of (2) as follows:

$$\Delta\boldsymbol{x}^{(i)} = -\mu[\boldsymbol{g}_{\boldsymbol{x}}^{(i)}]^{-1}\boldsymbol{g}^{(i)} \tag{3}$$

where μ is a factor that improves the convergence properties of the iterative process. In [10], [11], [13], and [15]–[18], several methods for computing adequate values of μ are proposed and/or compared. If μ is the result of an optimization process, μ is called *optimal multiplier* .

III. Continuous Newton's Method

Let us consider a set of autonomous ordinary differential equations (ODE), as follows:

$$\dot{\boldsymbol{x}} = \boldsymbol{f}(\boldsymbol{x}). \tag{4}$$

The simplest method of numerical integrating (4) is the explicit Euler method, as follows:

$$\Delta\boldsymbol{x}^{(i)} = \Delta t\boldsymbol{f}(\boldsymbol{x}^{(i)}) \tag{5}$$

$$\boldsymbol{x}^{(i+1)} = \boldsymbol{x}^{(i)} + \Delta\boldsymbol{x}^{(i)} \tag{6}$$

where Δt is a given time step.

The analogy between (2) and (5) is straightforward if one defines

$$\boldsymbol{f}(\boldsymbol{x}) = -[\boldsymbol{g}_{\boldsymbol{x}}]^{-1}\boldsymbol{g}(\boldsymbol{x}). \tag{7}$$

Equations (2) can thus be viewed as the ith step of the Euler forward method where $\Delta t = 1$ [28]. Furthermore, robust NR techniques (3) are nothing but the ith step of the Euler integration method where $\Delta t = \mu$. In other words, the computation of the optimal multiplier μ corresponds to a variable step Euler method.

The equilibrium point $\boldsymbol{x}_0$ of (4) is

$$\mathbf{0} = \boldsymbol{f}(\boldsymbol{x}_0) = -[\boldsymbol{g}_{\boldsymbol{x}}|_0]^{-1}\boldsymbol{g}(\boldsymbol{x}_0). \tag{8}$$

Thus, assuming that $\boldsymbol{g}_{\boldsymbol{x}}$ is not singular, $\boldsymbol{x}_0$ is also the solution of (1). Observe that assuming a nonsingular Jacobian matrix for the power flow equations is an implicit hypothesis of any power flow analysis (see also the discussion in the following Section III-B).

A. Stability of the Continuous Newton's Method

Differentiating (7) with respect to $\boldsymbol{x}$ leads to

$$\begin{aligned}\boldsymbol{f}_{\boldsymbol{x}} &= \nabla_{\boldsymbol{x}}^T\boldsymbol{f}(\boldsymbol{x})\\ &= -\,[\boldsymbol{g}_{\boldsymbol{x}}]^{-1}\boldsymbol{g}_{\boldsymbol{x}} - (\nabla_{\boldsymbol{x}}^T([\boldsymbol{g}_{\boldsymbol{x}}]^{-1}))\boldsymbol{g}(\boldsymbol{x})\\ &= -\,\boldsymbol{I}_n - (\nabla_{\boldsymbol{x}}^T([\boldsymbol{g}_{\boldsymbol{x}}]^{-1}))\boldsymbol{g}(\boldsymbol{x})\end{aligned} \tag{9}$$

where $\boldsymbol{I}_n$ is the identity matrix of order n. Since the equilibrium point $\boldsymbol{x}_0$ is a solution for $\boldsymbol{g}(\boldsymbol{x}_0)=\boldsymbol{0}$, one has

$$\boldsymbol{f}_{\boldsymbol{x}}|_0 = -\boldsymbol{I}_n. \tag{10}$$

A proof of (10) using tensor notation is given in Appendix B. Equation (10) basically implies that all eigenvalues of $\boldsymbol{f}_{\boldsymbol{x}}$ at the solution point are equal to -1. Thus, (10) means that the solution of (1), if exists, is asymptotically stable. The reachability of this solution depends on the starting point $\boldsymbol{x}(t_0)=\boldsymbol{x}_0$, which has to be within the region of attraction (also called *stability region*) of $\boldsymbol{x}_0$.

This paper focuses on the stability of the numerical methods used to obtain the solution of (1), and thus we assume that $\boldsymbol{x}_0$ is within the region of attraction of $\boldsymbol{x}_0$. At this regard, observe that initial guesses can be of two types:

1) The initial guess is outside the region of attraction of the solution point. Numerical methods typically diverge if one starts with such initial guess. The determination of the region of attraction of the solution of the power flow equations is out of the scope of this paper. Some interesting discussion can be found in [21].
2) The initial guess is inside the region of attraction of the solution point. In this case, a numerical method is expected to converge but can, in some cases, diverge. This is the phenomenon that is addressed in this paper.

The proposed technique is expected to show better ability to converge than other methods presented in the literature if the initial guess is within the region of attraction.

B. Continuous Newton's Method and Homotopy

Let us assume that the power flow (1) depend on a scalar parameter $\lambda(\lambda \in \mathbb{R})$, as follows:

$$\boldsymbol{0} = \boldsymbol{g}(\boldsymbol{x}, \lambda). \tag{11}$$

In most voltage stability studies, this parameter typically multiplies load and generator powers, so that λ is also called *loading parameter* [23]. The study of the behavior of $\boldsymbol{g}$ as λ varies constitutes the well-known continuation power flow analysis, which is basically a homotopy method [30].

Differentiating (11) at a solution point, one has

$$\boldsymbol{0} = \boldsymbol{g}_{\boldsymbol{x}} d\boldsymbol{x} + \boldsymbol{g}_{\lambda} d\lambda \tag{12}$$

where the dependence on $\boldsymbol{x}_0$ has been omitted for the sake of simplicity. Equation (12) leads to the homotopy method proposed by Davidenko for computing the variation of $\boldsymbol{x}$ as a function of the parameter λ [29]

$$\frac{d\boldsymbol{x}}{d\lambda} = -[\boldsymbol{g}_{\boldsymbol{x}}]^{-1}\boldsymbol{g}_{\lambda} \tag{13}$$

where $\boldsymbol{g}_{\lambda} = \nabla_{\lambda}^{T}\boldsymbol{g}$.[1] The Davidenko's method fails at turning points (e.g., saddle-node bifurcations) because of the singularity of $\boldsymbol{g}_{\boldsymbol{x}}$. More details on homotopy techniques can be found in [31].

Equations (13) is equivalent to a set of ODE, where the integration variable is λ. It is relevant to note the similarity of the continuous Newton's method (7) and (13). In fact, let us define the function $\boldsymbol{h}(\boldsymbol{x}, t)$ as follows:

$$\boldsymbol{0} = \boldsymbol{h}(\boldsymbol{x}, t) = e^{t}\boldsymbol{g}(\boldsymbol{x}) \tag{14}$$

where e represents the natural exponential. Then, differentiating $\boldsymbol{h}$, one has

$$\begin{aligned}\boldsymbol{0} &= \boldsymbol{h}_{\boldsymbol{x}} d\boldsymbol{x} + \boldsymbol{h}_t dt \\ &= e^{t}\boldsymbol{g}_{\boldsymbol{x}} d\boldsymbol{x} + e^{t}\boldsymbol{g} dt.\end{aligned} \tag{15}$$

Thus (7) can be rewritten as

$$\frac{d\boldsymbol{x}}{dt} = -[\boldsymbol{h}_{\boldsymbol{x}}]^{-1}\boldsymbol{h}_t. \tag{16}$$

Equation (16) shows that t can be viewed as the continuation parameter for the function $\boldsymbol{h}(\boldsymbol{x}, t)$. As for the Davidenko's method, the continuous Newton's method fails at turning points where the power flow Jacobian matrix is singular.

The main difference between (13) and (16) is that λ is an internal or *forced* continuation parameter (thus leading to a *natural parameter* homotopy), while t is an external or *free-running* (thus leading to an *artificial parameter* homotopy) [32]. Thus only the final equilibrium point of (16) is physically relevant, while the values of $\boldsymbol{x}$ in intermediate iterations lack of interest. It seems more interesting to further investigate the fact that (13) can be viewed as a dynamic system. This is currently an open research topic.

C. Efficient Solution Methods of the Power Flow Problem

It is well-known that the forward Euler method, even with variable time step, can be numerically unstable. Reference [28] suggests that, given the analogy between the power flow (1) and the ODE (4), any well-assessed numerical method can be used to integrate (4). It is thus intriguing to use some efficient integration method for solving (1). Observe that, since the computation of $\boldsymbol{f} = -[\boldsymbol{g}_{\boldsymbol{x}}]^{-1}\boldsymbol{g}$ implies the inversion of the power flow Jacobian matrix, only explicit integration methods are suitable and computationally efficient, since one does not need to compute the Jacobian matrix of $\boldsymbol{f}$.

[1]Observe that (13) can be also obtained from (7). As a matter of fact, differentiating (4) and imposing $d\dot{\boldsymbol{x}} = 0$ leads to

$$\boldsymbol{0} = d\boldsymbol{f} = -\boldsymbol{I}_n d\boldsymbol{x} - [\boldsymbol{g}_{\boldsymbol{x}}]^{-1}\boldsymbol{g}_{\lambda} d\lambda$$

where it is implicitly assumed that $\boldsymbol{g}_{\boldsymbol{x}}$ does not depend on λ [see (23) and the formulation of the power flow problem in Appendix A].

For the sake of example, in the case study described in Section IV, we use a classical fourth order Runge–Kutta formula (RK4). A generic step of the RK4 is as follows:

$$\begin{aligned} \boldsymbol{k}_1 &= \boldsymbol{f}(\boldsymbol{x}^{(i)}) \\ \boldsymbol{k}_2 &= \boldsymbol{f}(\boldsymbol{x}^{(i)} + 0.5\Delta t\boldsymbol{k}_1) \\ \boldsymbol{k}_3 &= \boldsymbol{f}(\boldsymbol{x}^{(i)} + 0.5\Delta t\boldsymbol{k}_2) \\ \boldsymbol{k}_4 &= \boldsymbol{f}(\boldsymbol{x}^{(i)} + \Delta t\boldsymbol{k}_3) \\ \boldsymbol{x}^{(i+1)} &= \boldsymbol{x}^{(i)} + \frac{\Delta t(\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4)}{6}. \end{aligned} \tag{17}$$

The time step Δt can be adapted based on the estimated truncation error of the integration method. An interesting discussion on the Runge–Kutta truncation error estimation can be found in [33]. For example, the RK4 error can be estimated based on the half-step method, as follows:

$$\xi = \max\{\mathrm{abs}(\boldsymbol{k}_2 - \boldsymbol{x}^{(i+1)})\}. \tag{18}$$

Then the time step Δt can be computed based on the following simple heuristic rules:

$$\begin{aligned} &\text{if } \xi > 0.01 \text{ then } \Delta t \leftarrow \max\{0.985 \cdot \Delta t,\ 0.75\} \\ &\text{if } \xi \le 0.01 \text{ then } \Delta t \leftarrow \min\{1.015 \cdot \Delta t,\ 0.75\}. \end{aligned} \tag{19}$$

Based on these rules, the time step is increased if the truncation error is greater than a given threshold and decreased if the truncation error is lower than a given threshold. The minimum value of the time step is limited to 0.75. If the lower value of the time step Δt is not limited, in case of unsolvable unsolvable power flow problems, the proposed algorithm provides a solution close to the feasibility boundary of the power flow equations, as discussed in [21]. All thresholds and tuning parameters in (19) have been determined based on heuristic criteria.

It is important to note that any order and any version of the family of the Runge–Kutta formulas could be used, and any of these methods is numerically more stable than the Euler forward method.

IV. Case Study

Fig. 1 depicts the one-line diagram of the 1254-bus 1942-line model of the UCTE system. An in-depth description of this system is given in [34], while UCTE data in various formats can be found at [35]. In [35], three scenarios are available, namely summer, winter peak, and winter off-peak. Simulations have been solved for the summer scenario, however, similar results can be obtained for the winter cases. All simulations have been solved using the software package PSAT [36], which allows easily prototyping new algorithms. Furthermore, PSAT is open source, thus, the full code of the proposed algorithm is freely available at the author's webpage [37], so that the interested reader can readily reproduce all simulations presented in this section.

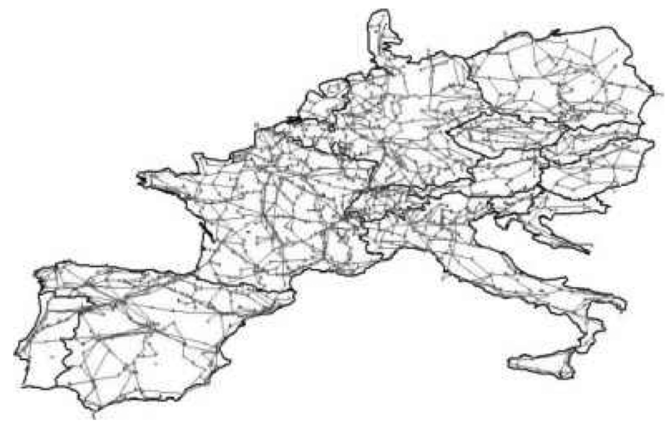

Fig. 1. One-line diagram of the 1254-bus 1942-line model of the UCTE system.

TABLE I
COMPARISON OF METHODS FOR SOLVING THE POWER FLOW OF THE UCTE SYSTEM

Method	# Iter. $\epsilon = 10^{-3}$	# Iter. $\epsilon = 10^{-4}$	# Iter. $\epsilon = 10^{-5}$
Standard NR	-	-	-
Fast Decoupled PF	-	-	-
Iwamoto's method	99	320	1021
Simple robust method	31	39	47
Runge-Kutta method	10	13	16

Table I compares the number of iterations necessary to obtain the solution of the power flow problem for the UCTE system and for a variety of convergence tolerances ϵ, as follows:

$$\epsilon \ge \max\{\mathrm{abs}(\Delta\boldsymbol{x}^{(i)})\}. \tag{20}$$

The methods compared in Table I are the following:

1) Standard NR method.
2) Fast decoupled power flow (FDPF) method [5]. Since the data only provides line reactances, the BX and XB versions of the FDPF produce same results in this case.
3) Iwamoto's method (IM). This method has been presented in [11] and consists in finding the optimal multiplier μ the minimizes the corrector vector $\Delta\boldsymbol{x}^{(i)}$.
4) The continuous Newton's method using a simple forward Euler variable step method [simple robust method (SRM)]. This method consists in comparing the corrector vector of the last two iterations. If $\Delta\boldsymbol{x}^{(i)} > \Delta\boldsymbol{x}^{(i-1)}$, then the multiplier μ is divided by 2.
5) The continuous Newton's method using the RK4 presented in (17). An initial time step $\Delta t = 1$ has been used.

For the sake of completeness, Fig. 2 shows the flowchart of the proposed technique. Observe that the flowchart is the same as the standard Newton–Raphson's technique or other robust methods except for (17) and the criterion used for updating Δt. For example, $\Delta t = 1 =$ constant in the standard Newton–Raphson's method.

The proposed technique does not directly take into account equipment limits and controls, such as reactive power generator limits. However, any technique that is currently used in the

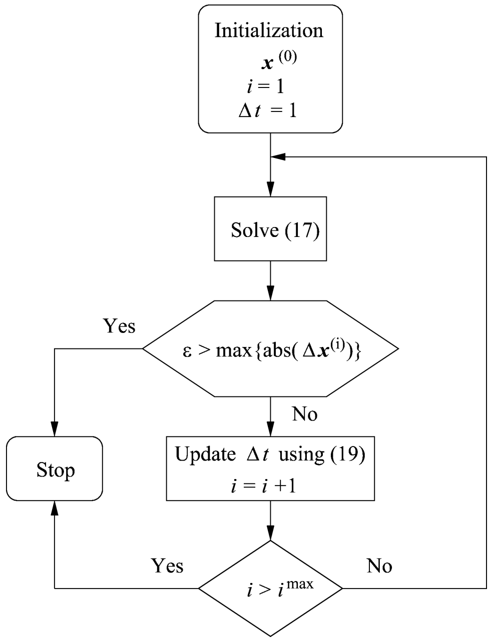

Fig. 2. Flowchart of the proposed RK4-based continuous Newton's method for solving the power flow analysis.

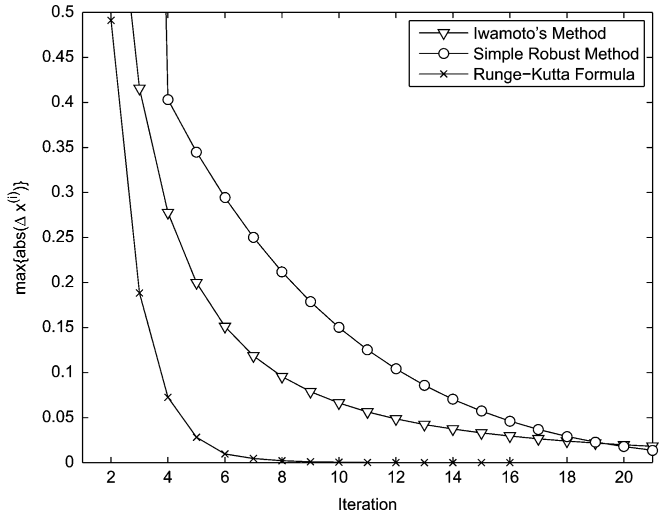

Fig. 3. Comparison of convergence errors obtained with different robust power flow solution methods for the UCTE system.

standard Newton–Raphson's method to take into account these limits can be readily included in the proposed algorithm. For example, one commonly used technique consists in checking at each iteration the value of the reactive power produced at PV buses, and switching the PV bus to a PQ bus if the reactive power limit has been violated.

A. Simulation Results

Table I shows that the standard NR and the FDPF, which in some cases presents better convergence properties than the NR method, fails to reach a solution for the UCTE system. The robust methods that uses a variable multiplier μ reach the solution but with a relatively high number of iterations. Finally, the RK4 applied to the continuous Newton's method converges in a relatively small number of iterations. This result has to be expected, since the RK4 ensures an higher efficiency than the Euler integration method.

Fig. 3 shows a comparison of the convergence error $\epsilon = \max\{\text{abs}(\Delta \boldsymbol{x}^{(i)})\}$ for the Iwamoto's method, the simple robust method and the proposed RK4 method for $\epsilon = 10^{-5}$. The IM provides a smaller error than the SRM in the first iterations. However, the SRM presents a smaller error than IM after iteration 20 and eventually converges before than the IM. The proposed RK4 always gives smaller convergence errors than the other methods.

The reason for the failure of the standard NR method is the initial guess, which in this case is not close enough to the solution and makes the NR map unstable. Thus, it would be reasonable to start with a robust technique and then switch back to the NR method once the corrector vector $\Delta x^{(i)}$ is smaller than a given threshold. This idea is not new in power system analysis [38]. However, it has been applied for time domain integration of power systems with faults not to power flow analysis. In [38], the implicit trapezoidal method is proposed as the workhorse for time domain analysis of power systems. Each step of the implicit trapezoidal method is solved by means of a Newton–Raphson technique. Step variations of the parameters can lead to a "bad" initialization for the next integration step. For this reason, the occurrence of faults can lead to the convergence failure of the trapezoidal method. To overcome this issue, [38] suggests to switch to a Runge–Kutta formula method for the few instants after the occurrence of a fault and then switch back the trapezoidal method when the variations of system variables are sufficiently stabilized.

TABLE II
COMPARISON OF MODIFIED ROBUST METHODS FOR SOLVING THE POWER FLOW OF THE UCTE SYSTEM

Method	# Iter. $\epsilon = 10^{-3}$	# Iter. $\epsilon = 10^{-4}$	# Iter. $\epsilon = 10^{-5}$
Iwamoto's method	32	32	33
Simple robust method	9	9	10
Runge-Kutta method	8	8	8

Table II shows the results for the modified version of the robust power flow methods and the continuous Newton's method. The threshold used to decide if it is convenient to switch to the NR method is $\max\{\text{abs}(\Delta \boldsymbol{x}^{(i)})\} < 10^{-2}$. Furthermore, in the case of the SRM, the multiplier μ is reset to 1 after each iteration. Using this technique, the number of iterations has been drastically reduced, especially in the case of the Iwamoto's method.

Fig. 4 shows a comparison of the convergence error $\max\{\text{abs}(\Delta \boldsymbol{x}^{(i)})\}$ for the modified Iwamoto's method, the simple robust method, and the proposed RK4 method for $\epsilon = 10^{-5}$. The IM provides a smaller error than the SRM in the first iterations. However, once the SRM switches to the standard NR method, it quickly converges. As in the previous simulations, the proposed RK4 always gives smaller convergence errors than the other methods.

B. Computational Burden

The heaviest computational part of any power flow solution technique is the factorization of the Jacobian matrix of the

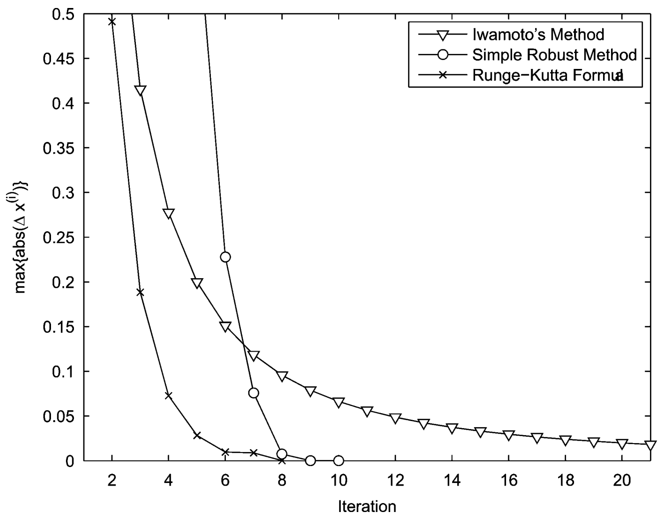

Fig. 4. Comparison of convergence errors obtained with different modified robust power flow solution methods for the UCTE system.

TABLE III
COMPUTATIONAL BURDEN OF THE IM, SRM, AND RK4 SOLUTION TECHNIQUES FOR THE UCTE SYSTEM ($\epsilon = 10^{-5}$)

Method	Original Version CPU time (s)	Modified Version CPU time (s)
Iwamoto's method	106.5	3.4
Simple robust method	3.5	0.8
Runge-Kutta method	3.4	1.4

system. All other computations are matrix and vector sums and products that can be done quite efficiently and we can assume that they do not significantly affect simulation times. In the case of Iwamoto's method, the Jacobian matrix is factorized once per iteration, while, in the case of the RK4 method, four times per iteration. In the case of the simple robust method, the number of factorizations per iteration is not constant, but one can assume that the mean value is about two factorizations per iteration.

Table III presents a comparison of the CPU times of the Iwamoto's method, the SRM, and the RK4 method for the convergence tolerance $\epsilon = 10^{-5}$. CPU times refer to a 2.4-GHz Intel Core 2 Duo processor running Matlab 7.6. Both the original and the modified version of these methods are compared. As expected, the performance of the method is proportional to the number of iterations by the number of factorizations of the power flow Jacobian matrix per iteration. The SRM and the RK4 methods show similar performances and are generally faster than the Iwamoto's method.

V. CONCLUSION

This paper proposes a continuous version of the Newton's method for solving the power flow problem. The paper has two main contributions: 1) a general framework for applying efficient numerical integration techniques for solving ill-conditioned power flow cases; and 2) formal taxonomy of the existing numerical methods for solving the power flow problem.

Future work will concentrate on further developing the analogy between the power flow problem, ODE systems and homotopy methods. The stability and region of attraction of the continuous Newton's method are promising fields of research.

APPENDIX A
POWER FLOW EQUATIONS

For the sake of completeness, this appendix gives the power flow (1) in polar form. For each bus h, one has

$$\begin{aligned} P_h =& V_h^2(g_h + g_{h0}) \\ &- V_h \sum_{\ell \neq h}^{n_\ell} V_\ell (g_{h\ell}\cos(\theta_h - \theta_\ell) \\ &+ b_{h\ell}\sin(\theta_h - \theta_\ell)) \\ Q_h =& -V_h^2(b_h + b_{h0}) \\ &- V_h \sum_{\ell \neq h}^{n_\ell} V_\ell (g_{h\ell}\sin(\theta_h - \theta_\ell) \\ &- b_{h\ell}\cos(\theta_h - \theta_\ell)) \end{aligned} \tag{21}$$

where P_h and Q_h are the real and reactive powers injected at bus h; V and θ are the bus voltage magnitude and phase angle, respectively; n_ℓ is the number of connections departing from bus h and g_h, g_{h0}, b_h, b_{h0}, $g_{h\ell}$ and $b_{h\ell}$ are line parameters, namely conductances and susceptances, as commonly defined in the literature.

Power injections P_h and Q_h at buses are modeled as the sum of generator and load powers connected to the bus h, as follows:

$$P_h = \sum_{i \in \mathcal{I}_h} P_{G_i} - \sum_{j \in \mathcal{J}_h} P_{L_j} \tag{22}$$

where $\mathcal{I}_h$ and $\mathcal{J}_h$ are the sets of generators and loads connected to bus h, respectively. The loading parameter increases linearly the bus power injections, as follows:

$$P_h(\lambda) = \lambda \sum_{i \in \mathcal{I}_h} P_{G_i} - \lambda \sum_{j \in \mathcal{J}_h} P_{L_j}. \tag{23}$$

APPENDIX B
PROOF OF (10)

This appendix proves (10) through tensor notation. Let us define the following quantities:

f_i	element i of the vector function $\boldsymbol{f}(\boldsymbol{x})$;
g_k	element k of the vector function $\boldsymbol{g}(\boldsymbol{x})$;
a_{ik}	element (i,k) of the matrix $[\boldsymbol{g_x}]^{-1}$;
$f_{i,j}$	partial derivative of f_i with respect to the variable x_j;
$g_{k,j}$	partial derivative of g_k with respect to the variable x_j;
$a_{ik,j}$	partial derivative of a_{ik} with respect to the variable x_j

where the dependence of f_i, g_k, a_{ik}, $f_{i,j}$, $g_{k,j}$ and $a_{ik,j}$ on $\boldsymbol{x}$ has been omitted for simplicity. Equations (7) and (9) can be rewritten as follows:

$$f_i = -\sum_{k=1}^{n} a_{ik} \cdot g_k \tag{24}$$

$$f_{i,j} = -\sum_{k=1}^{n} a_{ik} \cdot g_{k,j} - \sum_{k=1}^{n} a_{ik,j} \cdot g_k. \tag{25}$$

Since the matrix $[a_{ik}]$ is the inverse of $\boldsymbol{g_x}$, then

$$\sum_{k=1}^{n} a_{ik} \cdot g_{k,j} = \begin{cases} 1, & \text{if } i = j \\ 0, & \text{if } i \neq j. \end{cases} \tag{26}$$

Thus, (25) can be written in the following compact form:

$$f_{i,j} = -\delta_{ij} - \sum_{k=1}^{n} a_{ik,j} \cdot g_k \tag{27}$$

where δ_{ij} is the well-known Kronecker's operator. Furthermore, if $g_k = 0\ \forall k = 1, \ldots, n$ (which is verified at the solution point $\boldsymbol{x}_0$), then one obtains the final expression:

$$f_{i,j} = -\delta_{ij} \tag{28}$$

that is the tensor version of (10).

This result is straightforward for a scalar $g(x)$, i.e., for $x \in \mathbb{R}$ and $g \in \mathbb{R}$, as follows:

$$\begin{aligned} \dot{x} = f(x) &= -\frac{g(x)}{g_x(x)} \\ \Rightarrow \quad f_x(x) &= -\frac{g_x(x)}{g_x(x)} + \frac{g_{xx}(x)}{g_x^2(x)} g(x) \\ &= -1 + \frac{g_{xx}(x)}{g_x^2(x)} g(x) \end{aligned}$$

thus $f_x(x_0) = -1$ if $g(x_0) = 0$ and $g_x(x_0) \neq 0$.

ACKNOWLEDGMENT

The author would like to thank E. Aranda Ortega and J. Belmonte Beitia from the Applied Mathematics Department of the University of Castilla-La Mancha for the interesting discussions and suggestions on the continuous Newton's method. The author would also like to thank M. Sedlak from the Österreichische Elektrizitätswirtschafts of the Austrian Power Grid for the discussion on the ill-conditioned data set of the UCTE system.

REFERENCES

[1] W. F. Tinney and C. E. Hart, "Power flow solution by Newton's method," *IEEE Trans. Power App. Syst.*, vol. PAS-86, pp. 1449–1460, Nov. 1967.

[2] B. Stott, "Effective starting process for Newton-Raphson load flows," *Proc. Inst. Elect. Eng.*, vol. 118, no. 8, pp. 983–987, Aug. 1971.

[3] B. Stott and O. Alsac, "Fast decoupled load flow," *IEEE Trans. Power App. Syst.*, vol. PAS-93, pp. 859–869, Jun. 1974.

[4] B. Stott, "Review of load-flow calculation methods," *Proc. IEEE*, vol. 62, pp. 916–929, Jul. 1974.

[5] R. van Amerongen, "A general-purpose version of the fast decoupled loadflow," *IEEE Trans. Power Syst.*, vol. 4, no. 2, pp. 760–770, May 1989.

[6] G. X. Luo and A. Semlyen, "Efficient load flow for large weakly meshed networks," *IEEE Trans. Power Syst.*, vol. 5, no. 4, pp. 1309–1316, Nov. 1990.

[7] A. Semlyen, "Fundamental concepts of a Krylov subspace power flow methodology," *IEEE Trans. Power Syst.*, vol. 11, no. 3, pp. 1528–1537, Aug. 1995.

[8] A. J. Flueck and H. D. Chiang, "Solving the nonlinear power flow equations with an inexact Newton method using GMRES," *IEEE Trans. Power Syst.*, vol. 13, no. 2, pp. 267–273, May 1998.

[9] Y. Chen and C. Shen, "A Jacobian-free Mewton-GMRES(m) method with adaptive preconditioner and its application for power flow calculations," *IEEE Trans. Power Syst.*, vol. 21, no. 3, pp. 1096–1103, Aug. 2006.

[10] S. Iwamoto and Y. Tamura, "A fast load flow method retaining nonlinearity," *IEEE Trans. Power App. Syst.*, vol. PAS-97, pp. 1586–1599, Sep./Oct. 1978.

[11] S. Iwamoto and Y. Tamura, "A load fow calculation method for ill-conditioned power systems," *IEEE Trans. Power App. Syst.*, vol. PAS-100, pp. 1736–1743, Apr. 1981.

[12] S. C. Tripathy, G. D. Prasad, O. P. Malik, and G. S. Hope, "Load-flow solutions for ill-conditioned power systems by a Newton-like method," *IEEE Trans. Power App. Syst.*, vol. PAS-101, pp. 3648–3657, Oct. 1982.

[13] M. D. Schaffer and D. J. Tylavsky, "A nondiverging polar-form Newton-based power flow," *IEEE Trans. Ind. Appl.*, vol. 24, pp. 870–877, Sep./Oct. 1988.

[14] Z. Feng and W. Xu, "Fast computation of post-contingency system margins for voltage stability assessments of large-scale power systems," *Proc. Inst. Elect. Eng., Gen., Transm., Distrib.*, vol. 147, no. 2, pp. 76–80, Mar. 1990.

[15] D. J. Tylavsky, P. Crouch, L. F. Jarriel, and R. Adapa, "Improved power flow robustness for personal computers," *IEEE Trans. Ind. Appl.*, vol. 28, pp. 1102–1108, Sep./Oct. 1992.

[16] L. M. C. Braz, C. A. Castro, and C. A. F. Murari, "A critical evaluation of step size optimization based load flow methods," *IEEE Trans. Power Syst.*, vol. 15, no. 1, pp. 202–207, Feb. 2000.

[17] P. R. Bijwe and S. M. Kelapure, "Nondivergent fast power flow methods," *IEEE Trans. Power Syst.*, vol. 18, no. 2, pp. 633–638, May 2003.

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

[19] K. Iba, H. Suzuki, M. Egawa, and T. Watanabe, "A method for finding a pair of multiple load flow solutions in bulk power systems," *IEEE Trans. Power Syst.*, vol. 5, no. 2, pp. 582–591, May 1990.

[20] T. J. Overbye and R. P. Klump, "Effective calculation of power system low-voltage solutions," *IEEE Trans. Power Syst.*, vol. 11, no. 1, pp. 75–82, Feb. 1996.

[21] T. J. Overbye, "A power flow measure for unsolvable cases," *IEEE Trans. Power Syst.*, vol. 9, no. 3, pp. 1359–1365, Aug. 1994.

[22] T. J. Overbye, "Computation of a practical method to restore power flow solvability," *IEEE Trans. Power Syst.*, vol. 10, no. 1, pp. 280–287, Feb. 1995.

[23] C. A. Cañizares, Voltage Stability Assessment: Concepts, Practices and Tools, IEEE/PES Power System Stability Subcommittee, Final Document, 2002.

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

[25] S. Granville, J. C. O. Mello, and A. C. G. Melo, "Application of interior point methods to power flow unsolvability," *IEEE Trans. Power Syst.*, vol. 11, no. 4, pp. 1096–1103, Nov. 1996.

[26] G. D. Irisarri, X. Wang, J. Tong, and S. Mokhtari, "Maximum loadability of power systems using interior point nonlinear optimization method," *IEEE Trans. Power Syst.*, vol. 12, no. 1, pp. 162–172, Feb. 1997.

[27] C. A. Cañizares, "Applications of optimization to voltage collapse analysis," in *Proc. IEEE-Power Eng. Soc. Summer Meeting*, San Diego, CA, Jul. 1998.

[28] S. M. Hetzler, "A continuous version of Newton's method," *College Math. J.*, vol. 28, no. 5, pp. 348–351, Nov. 1997.

[29] D. F. Davidenko, "On a new method of numerical solution of systems of nonlinear equations," *Math. Rev.*, vol. 14, p. 906, 1953.

[30] M. Ilić and J. Zaborszky, *Dynamic and Control of Large Electric Power Systems*. New York: Wiley-Interscience, 2000.

[31] R. Seydel, *Practical Bifurcation and Stability Analysis: From Equilibrium to Chaos*, 2nd ed. New York: Springer-Verlag, 1994.

[32] M. Ilić and J. Zaborszky, *Dynamic and Control of Large Electric Power Systems*. New York: Wiley-Interscience, 2000.

[33] L. F. Shampine, "Some practical Runge-Kutta formulas," *Math. Comput.*, vol. 46, no. 173, pp. 135–150, 1986.

[34] Q. Zhou and J. W. Bialek, "Approximate model of European interconnected system as a benchmark system to study effects of cross-border trades," *IEEE Trans. Power Syst.*, vol. 20, no. 2, pp. 782–788, May 2005.

[35] Approximate Model of European Interconnected System. [Online]. Available: http://www.see.ed.ac.uk/~jbialek/Europe_load_flow/.

[36] F. Milano, "An open source power system analysis toolbox," *IEEE Trans. Power Syst.*, vol. 20, no. 3, pp. 1199–1206, Aug. 2005.

[37] PSAT ver. 2.0.0. [Online]. Available: www.uclm.es/area/gsee/Web/Federico.

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

Federico Milano (M'03) received the M.S. and Ph.D. degrees from the Università di Genova, Genova, Italy, in 1999 and 2003, respectively.

In 2001–2002, he worked at the University of Waterloo, Waterloo, ON, Canada, as a Visiting Scholar. He is currently an Associate Professor at the Universidad de Castilla-La Mancha, Ciudad Real, Spain. His research interests include voltage stability, electricity markets, and computer-based power system modeling.