RESTORING SOLUTION FOR UNSOLVABLE POWER FLOW: AN APPROACH USING THE AUGMENTED LAGRANGIAN ALGORITHM

Luciano V. Barboza  
UCPel – CEFET/RS  
Pelotas, Brazil  
luciano@ucpel.tche.br

Juliano B. Francisco  
UFSC  
Florianópolis, Brazil  
 juliano@mtm.ufsc.br

Mário C. Zambaldi  
UFSC  
Florianópolis, Brazil  
zambaldi@mtm.ufsc.br

Abstract – This paper aims to present and discuss a new and robust approach to restore the network electric equations solvability. The unsolvable power flow is modeled as a constrained optimization problem. The cost function is the least squares of the real and reactive power mismatches sum. The equality constraints are the real and reactive power mismatches at null injection buses and/or at those buses that must have their power demands totally supplied for technical or economical criteria. The mathematical model is solved using an algorithm based on Augmented Lagrangian methods considering the particular structure of the problem. The inner iterations of the proposed methodology are solved using Levenberg-Marquardt (LM) algorithm. Numerical results for both IEEE test systems and a real equivalent system from the Brazilian South-Southeast region are presented in order to assess the performance of the proposed approach.

Keywords: Unsolvable power flow, Restoring solvability, Augmented Lagrangian method, Levenberg-Marquardt algorithm

1 INTRODUCTION

Nowadays we observe that the electric power systems operate more and more close to their capacity limits. Several factors can be remarked, among them:

- a disordered increase of the power demands;
- lack of investment in the enlargement of the transmission systems;
- environment problems resulting to the construction of new generation units;
- the required time for that the electric system improvement are planned and executed;
- electric industries attempt to obtain larger profits.

Considering the former reasons, it is not unusual that the electric power systems present load levels that can not be supplied by the electric network. In such cases, the power systems can be subject to voltage instability phenomenon [1]. Thus, it is necessary to act upon the system controls or/and, as last attempt, to perform a system load shedding in order to restore the power flow equations solvability.

In a general way, the regions of the power flow equations solutions can be stated as [2]:
- no solution region: in which the power flow equations no present real solution. This case can occur due to either high load level or lack of important system transmission lines.
- emergency region: in which the power flow equations present at least two real solutions, one of which is used as an operational point. In this region, some operational limits can be violated. Hence, the electric system can be operated in this region for short periods of time.
- normal region: the power flow equations present real solutions and no operational limit is violated. The major system operator achievement is to operate the electric power system in this region.

A review on the literature show us several proposed methods to restore the power flow solvability. In [3][4][5] simple extensions of the conventional Newton load flow are proposed. A step length control strategy is aggregated to Newton algorithm in order to avoid the divergence of the iterative process. When the power flow equations no have real solutions, the step length factor tends to zero.

In [6] is proposed a strategy to restore the solvability combining the step control power flow in cartesian coordinates proposed in [4] with the left eigenvector associated to the null eigenvalue of the Jacobian matrix. Such approach is based on [7]. The main goal is to obtain an operational point as close as possible to the specified power demands.

In [8] is proposed an optimization technique based on Interior Point algorithm. In this case, although the mathematical model more complex, it allows to attain a more realistic solution.

On the other hand, in [9] is proposed an optimization technique based on Newton algorithm in which only equality constraints are modeled. This approach also uses a step control strategy associated to the Gauss-Newton method applied to the initial iterations.

In this work we propose to model the minimum load shedding problem as in [9]. In this, the special structure of the problem is exploited. It is based on the Augmented Lagrangian method and the Levenberg-Marquardt algorithm to handle with the inner iterations. Besides of the advantage to solve a constraint problem by a sequence of unconstrained problems (or at least simpler problems than the original), the Augmented Lagrangian method presents good convergence property. Another advantage is that the solution process reduces to the Levenberg-Marquardt method in cases in which the power flow equations present real solution.
The paper is organized as follows. Next section discusses the mathematical model used to deal with the minimum load shedding problem. Section 3 presents the Augmented Lagrangian method and Levenberg-Marquardt algorithm. The numerical results obtained are shown in section 4. Conclusions and final remarks are presented in section 5.

2 MINIMUM LOAD SHEDDING PROBLEM

The solvability restoration problem can be modeled as a minimum load shedding one stated as

\[
\text{Min } \frac{1}{2} f(x)^T f(x) \tag{1}
\]

\[\text{s.t. } g(x) = 0 \]

where \( f \) is the vector with real and reactive power mismatches; \( g \) is the vector with real and reactive power mismatches at all null injection buses and the mismatches at buses that have their power demands totally supplied by the electric system; and \( x \) is the vector with the optimization variables.

It is important to observe that the optimization model stated in (1) is typically a constrained nonlinear least squares problem. This fact leads to the use of mathematical techniques that takes advantage of this special feature.

Considering \( P_{d_i} \) and \( Q_{d_i} \) the generated real and reactive powers and \( P_{a_i} \) and \( Q_{a_i} \) the real and reactive power demands, the power mismatches in vectors \( f \) and \( g \) are

\[
P^\text{calc} \left( V, \delta \right) - P_{d_i} + P_{a_i} = 0 \]

\[
Q^\text{calc} \left( V, \delta \right) - Q_{d_i} + Q_{a_i} = 0 \tag{2}
\]

where \( P^\text{calc} \) and \( Q^\text{calc} \) are the real and reactive power injections, respectively. All variables specified at bus \( i \). In this study, we use

\[
x = \left[ V \quad \delta \right] \tag{3}
\]

where \( V \) is the voltage magnitudes at PQ buses and \( \delta \) is the phase angles of the complex voltages at all buses, except at slack bus.

Regarding to the vector \( g \), all real and reactive power mismatches at null injection buses must be set in it by technical reasons. At the iterative process, mismatches in this vector are impossible to be handled. In this case, we can add to the vector \( g \) other real and reactive power mismatches. Consider the case in which the firm contract of energy has been agreed between two electrical agents. In this case, the power demands at specified buses must be totally supplied. Thus, the real and reactive power mismatches corresponding to those buses can be set in vector \( g \) too.

3 THE AUGMENTED LAGRANGIAN ALGORITHM

The problem (1) can be modified to

\[
\text{Min } \frac{1}{2} \| f(x) \|^2 \tag{4}
\]

\[\text{s.t. } g(x) = 0 \]

where \( \| \cdot \| \) means the Euclidian norm.

Observe that if the power flow equations have a real solution, problem (4) resumes simply to determine the vector \( x^* \) so that

\[
\begin{bmatrix}
  f(x^*) \\
  g(x^*)
\end{bmatrix} = 0 \tag{5}
\]

The procedure used to solve (4) is based on advancements in the Augmented Lagrangian theory [10][11][12][13], including the use of Levenberg-Marquardt method [14][15] to solve the nonlinear equations in the inner iterations.

The Augmented Lagrangian function associated to (4) is

\[
\mathcal{L}(x, \lambda, \rho) = \frac{1}{2} \| f(x) \|^2 + \lambda^T g(x) + \frac{\rho}{2} \| g(x) \|^2 \tag{6}
\]

where \( \lambda \) is the vector with Lagrange multipliers and \( \rho > 0 \) is a penalty parameter.

3.1 Augmented Lagrangian – Outer Iteration

The method depends on a set of parameters that are necessary to ensure the convergence to stationary points. Usually, they do not vary too much for different tests (see [11] and [13] for more details). A usual set of parameters for this problem is given in Section 4. Thus, the Augmented Lagrangian outer iteration can be established as shown in the following algorithm. Consider \( \varepsilon_{(k)} \) and \( \upsilon_{(k)} \) nonnegative real number sequences that converge to zero, where \( k \) is the iteration index.

**Step 0:** let \( k = 0 \), \( \varepsilon_{(0)} > 0 \), \( \upsilon_{(0)} > 0 \), \( \rho_{(k),+} > 0 \), \( x_0 \), \( \lambda_{(0),+} \), \( 1 < \xi_{(k)} < \zeta_{(k)} < +\infty \), \( \gamma \in (0,1). \)

**Step 1:** let \( \tau_{(k+1)} = \min \left\{ \varepsilon_{(k),+}, \upsilon_{(k),+} \right\} \| g(x_{(k)}) \| \)

**Step 2:** calculate \( x_{(k+1)} \) so that

\[
\left\| V_x \mathcal{L} \left( x_{(k+1)}, \lambda_{(k+1),+}, \rho_{(k+1),+} \right) \right\| \leq \tau_{(k+1)} \tag{7}
\]

**Step 3:** if

\[
g \left( x_{(k+1)} \right) = 0 \text{ and } \left\| V_x \mathcal{L} \left( x_{(k+1)}, \lambda_{(k+1),+}, \rho_{(k+1),+} \right) \right\| = 0:
\]

- stop the algorithm and consider \( x_{(k+1)} \) a stationary point of (4) with Lagrange multipliers \( \lambda_{(k+1),+} \) associated to.
- otherwise, go to step 4.
Step 4: update the Lagrange multipliers vector by
\[ \hat{\lambda}_{(i+1)} = \hat{\lambda}_{(i)} + \rho_{(i)} g(x_{(i)}) \]
Step 5: if \[ \| g(x_{(i)}) \|_2^2 \leq \gamma \| g(x_{(i)}) \|_2^2 : \]
- \[ \rho_{(i+1)} \leftarrow \rho_{(i)} \]
- otherwise, \[ \rho_{(i+1)} \in [\zeta_1 \rho_{(i)}, \zeta_2 \rho_{(i)}] \]
Step 6: let \( k = k + 1 \), go to step 1.

One of the main features of the Augmented Lagrangian method is to present good convergence property. Let \( \{x_{(i)}\} \) be the solution sequence generated by the former algorithm and \( x^* \) one of its accumulation points, then the following results can be verified \[ \[10\][11].

(i) \( x^* \) is a stationary point from
\[ \text{Min} \quad \| g(x) \|_2^2 \]
\[ \text{s.t.} \quad x \in \mathbb{R}^n \]

(ii) if \( x^* \) is feasible, that is, \( g(x^*) = 0 \), and the Jacobian \( G(x^*) \) has full rank, then \( x^* \) satisfies the Karush-Kuhn-Tucker conditions from (4), that is, \( x^* \) is a stationary point from the original problem.

3.2 Levenberg-Marquardt Algorithm – Inner Iteration

At each iteration from Augmented Lagrangian method, called outer iteration, it is necessary to compute a vector \( x \) that satisfies (7), here called inner iteration. This can be performed minimizing the Lagrangian function (6). The proposed approach uses the special problem structure, generating an oriented algorithm for the inner iteration.

Observe that
\[ \mathcal{E}(x, \lambda, \rho) = \frac{1}{2} \| f(x) \|_2^2 + \lambda^T g(x) + \frac{\rho}{2} g(x)^T g(x) \]
\[ = \frac{1}{2} \| f(x) \|_2^2 + \frac{1}{2} \sqrt{\rho} g(x) + \frac{\lambda}{\sqrt{\rho}} \| g(x) \|_2^2 \]
\[ = \frac{1}{2} \left[ \left( f(x) \right)^T \sqrt{\rho} g(x) + \frac{\lambda}{\sqrt{\rho}} \right] - \frac{1}{2} \rho \| g(x) \|_2^2 \]

Thus, since \( \lambda \) and \( \rho \) are fixed, we have that
\[ \text{Min} \quad \mathcal{E}(x, \lambda, \rho) \quad \text{(9)} \]
is the same that
\[ \text{Min} \quad \frac{1}{2} \left[ f(x) \right]^T \sqrt{\rho} g(x) + \frac{\lambda}{\sqrt{\rho}} \quad \text{(10)} \]

that is a nonlinear least squares problem.

We propose to use the Levenberg-Marquardt method \[ [14][15][16] \] to solve (10) because no second order information is necessary.

Let us define the function \( h(x) \) as
\[ h(x) = \left[ f(x) \right] + \frac{\lambda}{\sqrt{\rho}} \quad \text{(11)} \]

and denote the Jacobian of \( h \) evaluated at \( x \) by \( H(x) \). As in the outer iteration it is necessary to evaluate (7), it is convenient to express this gradient in terms of \( H \), that is,
\[ \nabla_x \mathcal{E}(x, \lambda, \rho) = [H(x)]^T h(x) \quad \text{(12)} \]

In order to describe the procedure, let us consider \( y_{(i)} \) an approximation to the inner iteration solution. The Levenberg-Marquardt method uses a search direction that is a cross between the Gauss-Newton direction and the steepest descent. At each iteration of the LM algorithm, a direction \( d_{LM} \) is determined by the minimization of the squared Gauss-Newton model added by a regularization term, that is,
\[ d_{LM} = \text{Min} \left[ H(y_{(i)}) d + h(y_{(i)})^T + \mu_{(i)} \| d \|_2^2 \right] \quad \text{(13)} \]

where \( \mu > 0 \), called Levenberg-Marquardt parameter, is a scalar suitably chosen. Thus, the method continues performing a linear search in the direction \( d_{LM} \) beginning from \( y_{(i)} \).

Using the first order necessary conditions to (13), we have that the direction \( d_{LM} \) is the solution of the linear system
\[ \left[ H(y_{(i)}) + \mu_{(i)} U \right] d_{(i)} = -H(y_{(i)})^T h(y_{(i)}) \quad \text{(14)} \]

where \( U \) is an identity matrix.

Notice that to solve (14) is necessary to obtain the matrix \( H^T H \). From literature it is well known that both the condition number of this matrix corresponds to the squared condition number of \( H \) \[ [17] \] and as the power system becomes more loaded, this condition number increases a lot. Thus, the linear system (14) is ill-conditioned and conventional techniques to factorize this matrix can usually produce bad results. An effective approach to deal with this problem is to consider (13) like a linear least squares minimization, that is,
\[ \text{Min} \left[ H(y_{(i)}) \right] d + h(y_{(i)})^T \quad \text{(15)} \]
With the formulation proposed in (15), we can use suitable procedures making good use of the structure of the matrix

\[
\begin{bmatrix}
    H(y_{(i)}) \\
    \sqrt{\mu_{(i)}} U
\end{bmatrix}
\]

and performing, for instance, special QR factorization.

### 3.2.1 The Choice of \( \mu \) Parameter

The parameter \( \mu \) in the LM method is fundamental in the process convergence. When the inner iteration sequence converges to a point \( y \), that is a root of the nonlinear system (11), a reasonable choice to \( \mu \), at iteration \( k \), is

\[
\mu_{(i)} = \min \left\{ \beta \| h(y_{(i)}) \|, \mu_{\text{min}} \right\}
\]

with the factor \( \mu_{\text{min}} \) determined based on [13]. For fixed \( 0 < \sigma_{\text{min}} < \sigma_{\text{max}} \),

\[
\sigma_{\text{max}} = \max \left\{ \sigma_{\text{min}}, \min \left( \sigma_{\text{max}}, \sigma_{(i)} \right) \right\}
\]

where \( \sigma_{(i)} \) is the solution of

\[
\bar{m} \rightarrow \min_{\sigma \in \mathbb{R}} \left\{ \| H(y_{(i)})^T H(y_{(i)}) + \sigma U \| s_{(i)} - z_{(i)} \| \right. \}
\]

with

\[
z_{(i)} = H(y_{(i)})^T h(y_{(i)}) - H(y_{(i-k-1)})^T h(y_{(i-k-1)})
\]

\[
s_{(i)} = y_{(i)} - y_{(i-k-1)}
\]

where \( y_{(i-k-1)} \) and \( y_{(i)} \) are two successive solutions of the inner iteration.

From (18), we obtain a closed expression to parameter \( \sigma_{(i)} \) which is

\[
\sigma_{(i)} = \frac{z_{(i)}^T s_{(i)} - \| H(y_{(i)}) s_{(i)} \|}{s_{(i)}^T s_{(i)}}
\]

### 3.2.2 The LM algorithm

The Levenberg-Marquardt algorithm can be resumed in the following steps. Consider the outer iteration \( j \) and, thus, \( x_{(i)}, r_{(i)}, \lambda_{(i)} \) and \( \rho_{(j)} \) are known.

**Step 0:** Let \( k \leftarrow 0, \bar{m}, \bar{p} > 0, 0 < \sigma_{\text{min}} < \sigma_{\text{max}} < \infty \).

**Step 1:** Build \( h \) as in (11) and evaluate \( h(y_{(i)}) \) and \( H(y_{(i)}) \). If \( \| H(y_{(i)})^T h(y_{(i)}) \| \leq r_{(i)} \):

- stop the inner iterative process and let \( x \leftarrow y_{(0)} \) (that satisfies (7)).
- otherwise, go to step 2.

**Step 2:** If \( k = 0 \):

- let \( \sigma_{(i)} = 1 \).
- otherwise, determine \( z_{(i)} \) and \( s_{(i)} \) as in (19) and (20), respectively, and evaluate \( \sigma_{(i)} \) as in (21).

**Step 3:** Choice of \( \mu_{(i)} \). If \( k = 0 \):

- let \( \mu_{(i)} = \bar{m} \| h(y_{(i)}) \| \).
- otherwise, determine \( \mu_{\text{max}} \) using (17) and \( \mu_{(i)} \) using (16).

**Step 4:** Determine \( d_{(i)}^M \) as in (13).

**Step 5:** Perform a line search to determine the step size \( t_{(i)} \) in order to satisfy Armijo’s condition.

**Step 6:** let \( y_{(i+1)} = y_{(i)} + t_{(i)} d_{(i)}^M \).

**Step 7:** let \( k \leftarrow k + 1 \) and return to step 1.

In order to guarantee convergence to stationary points of (10), we perform a line search along the direction \( d_{(i)}^M \) at step 5 [14][15]. Hence, we assure that (7) will be satisfied after a finite number of inner iterations.

### 4 NUMERICAL RESULTS

This section assesses the performance of the proposed methodology. Thus, we applied the method to two electric power systems, one from IEEE and another one from Brazilian South-Southeast region (BSS). Table 1 shows the main characteristics of the test systems. In this table, \( nb, nc, ng \) and \( nlb \) are number of buses, circuits (transmission lines and transformers), generators and load buses, respectively.

<table>
<thead>
<tr>
<th>System</th>
<th>( nb )</th>
<th>( nc )</th>
<th>( ng )</th>
<th>( nlb )</th>
</tr>
</thead>
<tbody>
<tr>
<td>IEEE</td>
<td>118</td>
<td>179</td>
<td>34</td>
<td>74</td>
</tr>
<tr>
<td>BSS</td>
<td>340</td>
<td>684</td>
<td>53</td>
<td>184</td>
</tr>
</tbody>
</table>

Table 1: Main features of test systems.

Three tests were performed in each power system:
- **Test A:** the base case. At this loadability level, the power flow equations have real solution and thus no load shedding is necessary. A conventional Newton-Raphson power flow presents solution to this case.
- **Test B**: the loadability level was increased so that a conventional power flow does not converge. Nevertheless, when a suitable redispatch of generation is performed, it is possible to supply all loads.

- **Test C**: the loadability level was increased up to the value in which even performing a redispatch of generation the total scheduled load is not possible to be supplied. In this case, it is necessary to make a load shedding.

In order to verify the quality of the results provided by the proposed approach, we compare them to those provided by the algorithm suggested in [9], whose approach is based on the well known Newton-Raphson method [12] applied to the KKT conditions of problem (4). We use “Algorithm A” for our method and “Algorithm B” to that suggested in [9]. When a load shedding is necessary, we use a load shedding index ($lsi$) to measure the unsolvability degree formulated as

$$lsi = \left( \frac{P_{d_{sch}} - P_{d_{sup}}} {P_{d_{sch}}} \right) \cdot 100\%$$

where $P_{d_{sch}}$ and $P_{d_{sup}}$ are the total real power demands scheduled and effectively supplied by the system, respectively.

The fixed parameters that appear in both outer and inner iterations are specified as shown in Table 2. Despite the wide range of choice, we take the ones suggested in [13] and [14] whenever possible.

### Table 2: Fixed parameters for the proposed method. The number inside the parenthesis is the number of iteration.

<table>
<thead>
<tr>
<th>Outer iteration</th>
<th>Inner iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\rho_{(0)} = 10$</td>
<td>$\mu = 0$</td>
</tr>
<tr>
<td>$\zeta_1 = 10$</td>
<td>$\mu_1 = 10$</td>
</tr>
<tr>
<td>$\zeta_2 = 10$</td>
<td>$\sigma_{\text{min}} = 5.10^{-3}$</td>
</tr>
<tr>
<td>$\gamma = 0.25$</td>
<td>$\sigma_{\text{max}} = 2.10^{-1}$</td>
</tr>
<tr>
<td>$\epsilon_{(0)} = 10^{-1}$</td>
<td>$\eta_1 = 0.99999$</td>
</tr>
<tr>
<td>$\eta_{(0)} = 10^{-1}$</td>
<td>$\eta_2 = 10^{-2}$</td>
</tr>
<tr>
<td>$\beta = 0.7$</td>
<td></td>
</tr>
</tbody>
</table>

4.1 **IEEE Test System**

For this power system, the tests were performed at follow loadability levels:

- **Test A**: 6,187.5 MW and 2,159.1 MVAR (base case – total values).
- **Test B**: loadability in base case increased by 10% (6,806.2 MW and 2,375.0 MVAR).
- **Test C**: loadability in base case increased by 15% (7,115.6 MW and 2,483.0 MVAR).

For all tests, the scheduled real power generation was set in 6,882.4 MW.

Firstly, Tables 3, 4 and 5 present the results obtained for each test in terms of powers. In these tables, $Q^{\text{supp}}_{d_{sup}}$ is the total reactive power demand effectively supplied by the electric system, and $P^{\text{red}}_{d_{sch}}$ is the total real power re-depatch.

### Table 3: Quality of the results – IEEE – Test A.

<table>
<thead>
<tr>
<th></th>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P^{\text{supp}}<em>{d</em>{sup}}$ (MW)</td>
<td>6,187.5</td>
<td>6,187.5</td>
</tr>
<tr>
<td>$Q^{\text{supp}}<em>{d</em>{sup}}$ (MVAR)</td>
<td>2,159.1</td>
<td>2,159.1</td>
</tr>
<tr>
<td>$P^{\text{red}}<em>{d</em>{sch}}$ (MW)</td>
<td>6,882.4</td>
<td>6,882.4</td>
</tr>
<tr>
<td>$lsi$ (%)</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### Table 4: Quality of the results – IEEE – Test B.

<table>
<thead>
<tr>
<th></th>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P^{\text{supp}}<em>{d</em>{sup}}$ (MW)</td>
<td>6,803.7</td>
<td>6,806.2</td>
</tr>
<tr>
<td>$Q^{\text{supp}}<em>{d</em>{sup}}$ (MVAR)</td>
<td>2,374.7</td>
<td>2,375.0</td>
</tr>
<tr>
<td>$P^{\text{red}}<em>{d</em>{sch}}$ (MW)</td>
<td>8,155.8</td>
<td>8,176.2</td>
</tr>
<tr>
<td>$lsi$ (%)</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

### Table 5: Quality of the results – IEEE – Test C.

<table>
<thead>
<tr>
<th></th>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$P^{\text{supp}}<em>{d</em>{sup}}$ (MW)</td>
<td>6,986.6</td>
<td>6,986.6</td>
</tr>
<tr>
<td>$Q^{\text{supp}}<em>{d</em>{sup}}$ (MVAR)</td>
<td>2,465.3</td>
<td>2,465.2</td>
</tr>
<tr>
<td>$P^{\text{red}}<em>{d</em>{sch}}$ (MW)</td>
<td>8,543.6</td>
<td>8,543.6</td>
</tr>
<tr>
<td>$lsi$ (%)</td>
<td>1.81</td>
<td>1.81</td>
</tr>
</tbody>
</table>

We observe that in Test C it is necessary a load shedding index about 1.81%. Therefore, the total scheduled power demand can not be supplied by the network even that a power redispatch is performed.

Tables 3, 4 and 5 show the good accuracy of the proposed approach.

Tables 6, 7 and 8 show the results of computational effort. In these tables, $NI$ is the number of iterations and $CT$ is the computational time. In the proposed algorithm the number outside the parenthesis is the number of outer iterations and the number inside the parenthesis corresponds to the number of inner iterations at each outer iteration.

### Table 6: Computational effort – IEEE – Test A.

<table>
<thead>
<tr>
<th></th>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>$NI$</td>
<td>2 (4, 3)</td>
<td>8</td>
</tr>
<tr>
<td>$CT$ (seg)</td>
<td>0.29</td>
<td>0.39</td>
</tr>
</tbody>
</table>
From Tables 6, 7 and 8, we conclude that the proposed algorithm presents an acceptable behavior. In all tests, algorithm A has presented a speedup when compared to algorithm B. The speedup reaches 25.6% in Test A, 47.2% in Test B and 4.8% in Test C.

4.2 BSS Test System

This system is a real equivalent one from Brazilian South-Southeast region. The three tests performed were:
- **Test A**: loadability of 42,407.5 MW and 657.3 MVAr (base case).
- **Test B**: base case increased by 1.4% (43,001.1 MW and 666.5 MVAr).
- **Test C**: base case increased by 3% (43,679.7 MW and 677.0 MVAr).

It is worthwhile to point out that the algorithm proposed in [9] using only the Newton optimization method did not converge for all tests. To assess the quality of the results we use one Gauss-Newton iteration at the beginning of the iterative process in Algorithm B. Tables 9, 10 and 11 show the results for BSS system.

<table>
<thead>
<tr>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>NI</td>
<td>1 (13)</td>
</tr>
<tr>
<td>CT (seg)</td>
<td>0.38</td>
</tr>
</tbody>
</table>

Table 7: Computational effort – IEEE – Test B.

<table>
<thead>
<tr>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>NI</td>
<td>2 (11, 7)</td>
</tr>
<tr>
<td>CT (seg)</td>
<td>0.6</td>
</tr>
</tbody>
</table>

Table 8: Computational effort – IEEE – Test C.

Observing Tables 9, 10 and 11, we can conclude that the results provided by the proposed algorithm have good accuracy again.

Tables 12, 13 and 14 present the computational behavior of the methodologies.

<table>
<thead>
<tr>
<th>Algorithm A</th>
<th>Algorithm B</th>
</tr>
</thead>
<tbody>
<tr>
<td>NI</td>
<td>1 (12)</td>
</tr>
<tr>
<td>CT (seg)</td>
<td>0.95</td>
</tr>
</tbody>
</table>

Table 11: Quality of the results – BSS – Test C.

For this test system, all tests presented speedup again, except in Test C. Test A reaches speedup of 5% and Test B of 56.8%. However, the proposed approach increased the computational time by about 129.5% in Test C.

Despite the widely varying speedups in Test C, it is worthwhile to mention that to forcing convergence towards a stationary point, we performed in the Algorithm B some Gauss-Newton type iterations before Newton ones, which is not always a fair comparison in terms of both robustness and reliability of the algorithms.

5 CONCLUSIONS

This paper proposed an algorithm to restore the solvability of the power flow equations when these ones do not have a real solution. Such cases can occur due to high loadability levels or/and loss of important equipment in the electric power network. The proposed methodology is an Augmented Lagrangian-type method. This approach leads to two iterative loops to be solved. To solve the inner iterations it was proposed the use of Levenberg-Marquardt algorithm.
The numerical results prove the efficacy of the methodology. In cases where the power flow equations have a real solution, the proposed method does not interfere in the iterative process and consequently in the determination of the operational point.

In terms of computational performance, the algorithm showed to be efficient. The computational times obtained encourage us in continuing the researches. It is important to point out that the implemented methodology is a basic one. Other features can be added to the algorithm in order to let it more efficient.

REFERENCES


