An Optimal Power Flow with User-Defined Objective Functions and Constraints

Zulmar S. Machado Jr, Glauco N. Taranto, Djalma M. Falcão
Federal University of Rio de Janeiro
Rio de Janeiro, Brazil
zulmar@coep.ufrj.br, tarang@coep.ufrj.br, falcao@nacad.ufrj.br

Abstract - This paper presents an optimal power flow (OPF) methodology with user-defined objective functions and constraints. First and second-order automatic differentiation techniques are applied, allowing flexibility to the program. The algorithm is based on an interior-point method with an embedded full Newton-Raphson solver. The computational framework is structured with advanced concepts of object oriented modeling (OOM) using the C++ language. This framework readily allows modeling generalizations, making it possible to incorporate detailed models of generators and their controls. Numerical results are presented for a 45-bus equivalent of the Southern Brazilian power system.

Keywords - Optimal Power Flow, User-defined OPF, Automatic Differentiation, Object-Oriented Modeling

1 INTRODUCTION

IN present day operation of electric power systems, where market-driven rules have led the systems to operate with reduced security margins, it is of paramount importance the availability of robust and flexible computational tools for system analysis. In this context, the object-oriented modeling encompasses these desirable features and puts itself as an alternative to becoming a framework for development of new power system tools.

Generally, in conventional Optimal Power Flow (OPF) programs, models are pre-defined and cannot be modified by users, unless changing the source code. Some programs allow users to choose objective functions and constraints, among a pre-defined set. Although being able to fulfill most of the usual analysis needs, they limit, to some extent, the desires of users that would like to analyze unusual objectives and constraints.

This paper proposes an OPF able to have user-defined objective functions and constraints, without the need to change and recompile the program source code.

The objective function and constraints are constructed from an user-defined format utilizing a structure called *Model*, which aggregates parameters, equations and information flows of a given power system apparatus. This technique readily allows for generalizations in the formulations of objective functions and constraints. Furthermore, it allows for example, the utilization of more detailed generator models, making it possible to have internal generator variables as part of the objective function. Models for round-rotor and salient-pole synchronous machines are options available in the program. The representation is possible because the OPF formulation is based on augmented algebraic power system equations.

The proposed OPF methodology utilizes automatic differentiation techniques, and calculates the first and second-order derivatives of the objective function and of the equality and inequality constraints, thus obtaining the Hessian and Jacobian matrices of the lagrangian function. This capability of automatic differentiation aggregates new features to the program, making readily possible the incorporation of any differentiable mathematical function in the OPF formulation.

It is presented an OPF algorithm with the Karush-Kuhn-Tucker (KKT) optimality conditions solved via a full Newton-Raphson method. The algorithm is based on an interior-point method implemented with object-oriented modeling in C++.

Numerical results are presented for a 45-bus equivalent of the Southern Brazilian power system.

2 MODELING

For transient and steady-state stability analysis, a power system is represented by a set of nonlinear differential and algebraic equations as given by (1) below.

$$\begin{aligned} \dot{x} &= f(x,y) \\ 0 &= g(x,y) \end{aligned} \tag{1}$$

where x is the vector of state variables, such as rotor speed and angle, y is the vector of algebraic variables, such as the complex nodal voltages, and f and g are vectors of non-linear functions describing, respectively, the differential equations modeling the system dynamical elements (generators and their controllers, FACTS devices, induction motors, etc.) and the algebraic equations modeling the network.

Conventional OPF formulations are only based in the set of algebraic equations given in (1), which basically represents the load flow equations. Steady-state or small-signal stability analysis are based in the equilibrium point of (1) yielding the following representation:

$$\begin{aligned} \dot{x} = \quad 0 &= f(x,y) \\ 0 &= g(x,y) \end{aligned} \tag{2}$$

The set of algebraic equations given by (2) is then used as a new set of equality constraints in an OPF formulation, similarly to the methodology used in the quasi steady-state simulation presented in [1] and originally introduced in [2].

The addition of the equation $f(x,y) = 0$ to the set of equality constraints opens a whole range of opportunities for new OPF formulation, such as, comprehensive representation of generator capability curves and its control systems, assessment of electromechanical dynamic analysis, etc.

The generator model adopted in this paper is based in a detailed representation of the machine and its control systems, i.e., the automatic voltage regulator (AVR) and the speed governor [3, 4]. Figure 1 shows how the machine, the AVR and the speed governor are built in the computational framework. The output of the model can be either a current injection or a power injection.

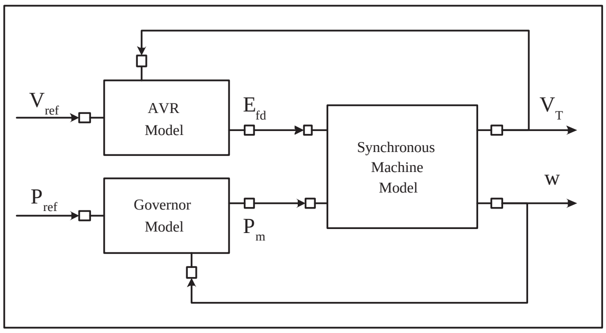

Figure 1: Functional representation of the synchronous machine and its control systems

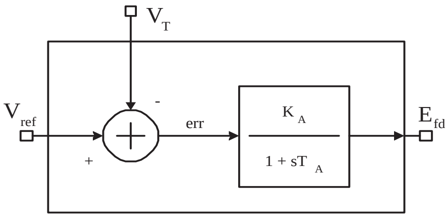

Figure 2: Automatic voltage regulator model

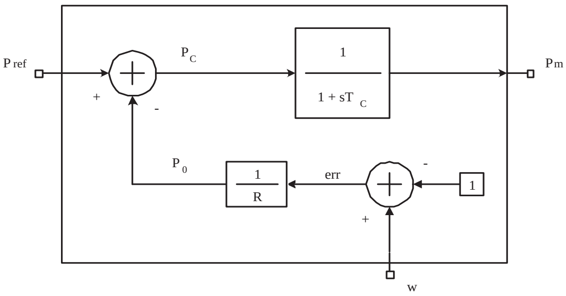

Figure 3: Speed governor model

Figure 2 and Figure 3 show the simple models that are being utilized for the AVR and for the speed governor, respectively. The AVR model has as input variables the terminal voltage V_T and the reference voltage V_{ref}, and as output variable the field voltage E_{fd}. The speed governor model has as input variables the rotor speed w and the reference mechanical power P_{ref}, and as output variable the mechanical power P_m.

It is noted that these models have variable reference values as controlling variables, in contrast to either load flow or time simulation programs, where those references are kept constant. The salient-pole machine, the AVR and the speed governor models used in this work are described in [5].

3 COMPUTATIONAL STRUCTURE FOR MATHEMATICAL MODELING

In this work, the building of the equations and control actions modeling devices and control schemes, as well as the objective functions and constraints in the optimization problem, are dealt with by a computational structure based on the technique of object oriented modeling.

The modeling technique main characteristic is to encompass in its structure all the parameters, equations, and control actions modeling a device regarding to a particular application (power flow, for example). Some of this information may be shared by several applications. The models features may be described as:

1. **Data and Equations Storage**: parameters, variables, and equations that describe how the models are managed by the computations structure;

2. **Initial Conditions Determination**: from the information on the device status, it is possible to calculate the initial conditions of the state variables and the model set points as described by their equations;

3. **Equations Differentiation and Solution**: responsible for the calculation of the partial derivatives in relation to the variables of interest (automatic differentiation) and to solve the model set of equations;

4. **Specific Features**: capacity to introduce particular actions for a given network device.

The standardization and generalization in the models development allow that several applications may have access to common features as well as to their own features. For instance, if an optimal power flow and a time simulation programs share the same computational structure, new models for any of the two applications can be incorporated to the class structure with no need for code alteration of any of the application programs.

In the optimal power flow software, the objective function and constraints are dealt with in a high level of generalization. These functions are built automatically by the computational structure from information provided by the users by combining parameters, equations, and control actions of each system device.

The general procedure adopted for building the objective function and constraints is to compose the equations in the form of a block diagram in which each block represents an elementary mathematical operation. Figure 4 shows the representation in block diagram of the hypothetical objective function described in (3) which is composed of the weighted sum of the active output and the squared reactive output of all the generators in the system. In this diagram, the variables (P_i, Q_i) and parameters (C_{p_i}, C_{q_i}) associated with each generator (Gen $\#i$, $i = 1, ..., n$) are combined using the elementary mathematical operations *Square* and *Gain*.

$$f_{obj} = C_{p_1} \cdot P_1 + \frac{1}{2}\, C_{q_1} \cdot Q_1^2 + ... + C_{p_n} \cdot P_n + \frac{1}{2}\, C_{q_n} \cdot Q_n^2 \tag{3}$$

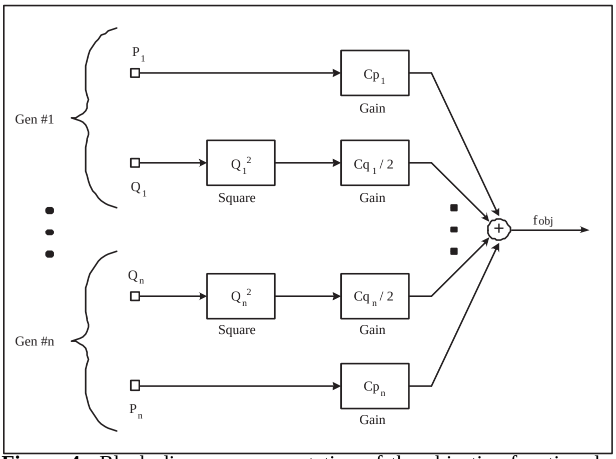

Figure 4: Block diagram representation of the objective function described in equation 3

The procedure adopted for the treatment of functions in the block diagrams allows the connection of an output variable of a block to several input variables of different blocks, but a block input variable can be connected to only one output variable from another block.

The modeling procedure adopted in this work is based on a generic block to describe any mathematical function without association to particular elementary operations, i.e., this generic block is mainly used to manage the connectivity among the elementary blocks. The number of inputs and outputs of the block depends on the kind of operation that the block is going to perform. Figure 5 presents one generic block connected to others blocks allowing the building of different mathematical functions.

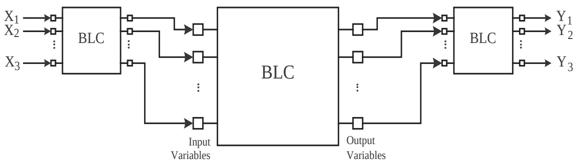

Figure 5: Representation of a generic block

The computational structure allows the building of sub-models which can be assembled to build the model of a more complex device. This operation is performed by combining the blocks of each part according to the physical coupling of the components. An example of this kind of composed model building is shown in Figure 1, representing the model of a generator which is composed by the sub-models representing the synchronous machine, the voltage regulator, and the governor. Figures 2 and 3 represent the block diagrams of the voltage regulator and the governor, respectively. This sub-model assembling process gives great flexibility to the computational structure by allowing a sub-model attached to a device to be substituted, removed, or added to a device to satisfy the need of a particular application program.

4 OPTIMAL POWER FLOW FORMULATION

The optimization problem can be written, in its standard formulation, as

$$\begin{aligned} \min_{s.\ a.}\ & f_{obj}(z) \\ h_{eq}(z) &= 0 \\ g_{ds}(z) &\leq 0 \end{aligned} \tag{4}$$

where f_{obj} is the objective function, h_{eq} are the equality constraints, and g_{ds} are the inequality constraints, and z is the vector of state (x) and algebraic (y) variables.

The set of equations given in (1) constitutes the equality constraint set h_{eq} , as shown in (5) below.

$$z = \left(\begin{smallmatrix} x \\ y \end{smallmatrix}\right) \Rightarrow h_{eq}(z) = \begin{pmatrix} f(x,y) \\ g(x,y) \end{pmatrix} = 0 \tag{5}$$

As the OPF formulation consider adjustments in the control variables u, which correspond to physical controls available in the system, like transformer tap position, voltage control in generation busses, etc., then z can be expanded as follows:

$$z = \begin{pmatrix} x \\ y \\ u \end{pmatrix} \tag{6}$$

In the classical OPF formulation, the objective function $f_{obj}(z)$ is a non-linear function representing some economic or security goal. The more usual objective functions are:

- Minimum generation cost;
- Minimum active losses;
- Minimum deviation from an operating point;
- Maximum power transfer between areas;
- Maximum loadability;
- Minimum load shed.

The equality constraints $h_{eq}(z)$ contain the power balance equations modeling the power flow in the network. In the extended formulation described in this work, equations derived from the detailed modeling of generators may be incorporated to the equality constraints set.

Figure 6 presents an example system for the illustration of how the constraints associated to the power balance equations are inserted in the FPO. In this figure, a generator, a shunt element, and a series element are connected to bus $\#1$. Each one of these devices have associated models in which it is defined the contribution of the device to the power injected into the bus. Thus, the power balance equation is formed by the sum of the contributions of each device connected to it, as shown in (7) below.

$$S_{gen} + S_{serie} + S_{shunt} = 0 \tag{7}$$

Substituting each term of (7) by the expression corresponding to the respective device injection model, equation (8) is obtained which represents two equality constraints (active and reactive parts) in the FPO.

$$(P + jQ) + V_1[(V_1 - V_2)Y_{ser} + V_1 Y]^* + V_1(V_1 Y_{sht})^* = 0 \tag{8}$$

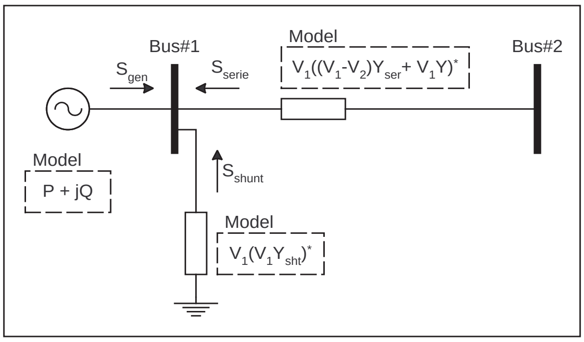

Figure 6: Power balance equations

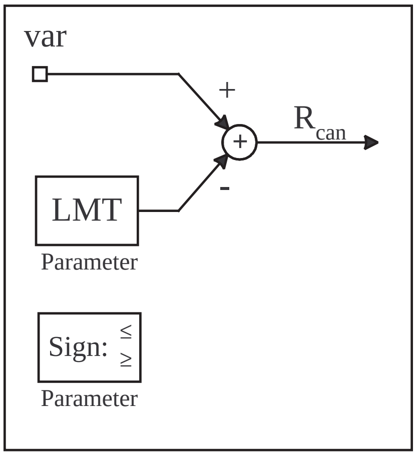

Figure 7: Model of bounded constraints

The power balance equations come from the network configurator. The network configurator must have a complete description of each substation and how transmission lines are attached to the substation equipment, such as switches and circuit breakers. Once the network topology is determined the nodal power injection equations are established, as shown by the arrows at bus$\#1$ in Figure 6. Then, the power balance equations are the only ones that can not be defined by the user.

The inequality constraints set $g_{ds}(z)$ represent the operational constraints like limits on voltages, power flows, control variables, etc. As in the case of equality constraints, the OPF formulation described in this paper allows the introduction of constraints representing more realistic limits on the generators like, for instance, constraints derived from the capability curve of the machines [6, 7].

Operating and control limits are usually represented by bounded (or box) constraints like the ones shown in (9) which represent maximum and minimum limits on the voltage, active power, and reactive power.

$$\begin{array}{lcl} V_{MIN} \leq & V & \leq V_{MAX} \\ P_{MIN} \leq & P & \leq P_{MAX} \\ Q_{MIN} \leq & Q & \leq Q_{MAX} \end{array} \tag{9}$$

In general, a bounded constraint may be characterized as an state variable limited by maximum and minimum values, as shown in (10),

$$MIN \leq var \leq MAX \tag{10}$$

which can be rewritten as

$$\begin{array}{r} var - MAX \leq 0 \\ var - MIN \geq 0 \end{array} \tag{11}$$

Based on (11), it is possible to build a model that represents both the maximum and minimum limits, as this type of constraints represents the difference between a variable and its limit (minimum or maximum). Thus, figure 7 presents a general model for this type of constraint in which the parameter *LMT* represents the maximum or minimum limit value and the parameter *Sign* stores the constraint sign ($\leq$ or $\geq$) which determines how the constraint is going to be treated by the OPF.

Functional constraints, like the ones in transmission lines power flow, power factor limits, or limits on the power flow on groups of lines chosen by the user, are usually more difficult to deal with by the OPF programs. For illustration purpose, suppose the building of a constraint on the sum of the power flow in three branches of the network. This constraint is shown in (12).

$$S_1 + S_2 + S_3 \leq S_{MAX} \tag{12}$$

where, S_1, S_2 and S_3, represent the power flow from one area of the system and S_{MAX} represent its maximum value. As the S_i, $i = 1, 2, 3$, are not explicit variables of the problem, they have to be written as function of the variables P and Q, as shown in (13).

$$S = \sqrt{P^2 + Q^2} \tag{13}$$

Substituting (13) in (12)

$$\sqrt{P_1^2 + Q_1^2} + \sqrt{P_2^2 + Q_2^2} + \sqrt{P_3^2 + Q_3^2} \leq S_{MAX} \tag{14}$$

From (14), the block diagram shown in figure 8 can be built.

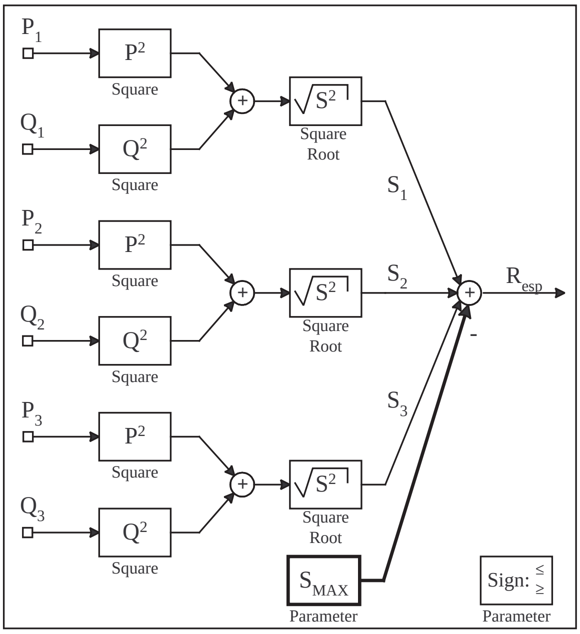

Figure 8: Model of functional constraint

5 INTERIOR POINT METHOD

In this work, the methodology proposed in [8] is used, in which the inequality constraints defined in (4) are transformed into equality constraints by the introduction of the slack variables s and by the application of penalty factors on the slack variables, as shown as follows

$$\begin{aligned} \min_{s.\ a.} \quad & f_{obj}(z) + \mathscr{P}(s) \\ & h_{eq}(z) = 0 \\ & g_{ds}(z) + s = 0 \end{aligned} \qquad (15)$$

The penalty function $\mathscr{P}(s)$, applied to the slack variables s, is a logarithmic function used to establish the interior point method [8].

Once the optimization problem is reduced to equality constraints only, then it is possible to write the lagrangian function as shown in (16).

$$\mathscr{L}(z,\lambda,\pi,s) = f_{obj}(z) + \lambda h_{eq}(z) + \pi[g_{ds}(z)+s] + \mathscr{P}(s), \qquad (16)$$

where λ and π are vectors of dual variables associated to the equality and inequality constraints, respectively.

Using the function defined in (16), it is possible to establish the first order Karush-Kuhn-Tucker optimality conditions [9], which are shown in (17).

$$\begin{aligned} \nabla_z \mathscr{L}() &= 0 = && \nabla_z f_{obj}(z) + \lambda \nabla_z h_{eq}(z) + \pi \nabla_z g_{ds}(z) \\ \nabla_\lambda \mathscr{L}() &= 0 = && h_{eq}(z) \\ \nabla_\pi \mathscr{L}() &= 0 = && [g_{ds}(z) + s] \\ \nabla_s \mathscr{L}() &= 0 = && \pi + \nabla_s \mathscr{P}(s) \end{aligned} \qquad (17)$$

The set of equations given in (17) can be solved by the Newtom-Raphson method trough the successive solution of the following set of linear equations:

$$\nabla^2 \mathscr{L}(z,\lambda,\pi,s) \times \Delta(z,\lambda,\pi,s) = -\nabla \mathscr{L}(z,\lambda,\pi,s) \qquad (18)$$

The computation of the first and second order derivatives of the lagrangian function in (18) is performed automatically using a technique based on the Chain Rule [10] and in the composition of elementary operations to built equations trough block diagrams already described in this paper.

6 RESULTS

The methodology described in this paper was applied to an equivalent of the Brazilian Southern power system with 45 buses, 56 transmission lines, 17 transformers, and 10 generators, as shown in the figure 15. The results were obtained using a software developed using advanced concepts of Object-Oriented Modeling (OOM) and the C++ language.

The results were obtained from a base case where the control variables considered were the terminal voltage at the generating buses and the generated active power. The objective function was the one given by equation (3) and built by block diagram as shown in Figure 4.

Figure 9 shows the convergence process for the voltage magnitude of the generating units of J.Lacerda A and J.Lacerda B.

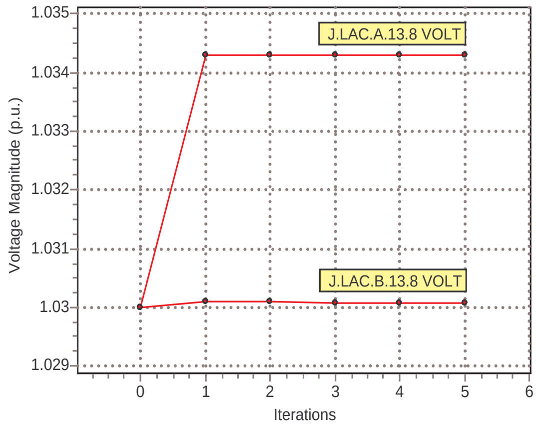

Figure 9: J.Lacerda voltage magnitude

Figure 10 shows the convergence process for the field current of the generating units of J.Lacerda A. The optimal value obtained was 1.39 pu.

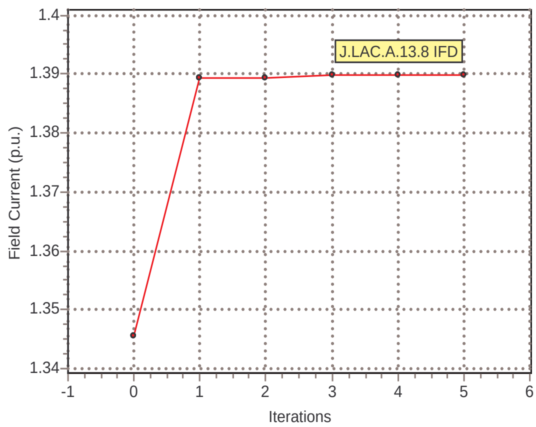

Figure 10: J.Lacerda field current

In order to show the flexibility in defining a new objective function, not "hardly" coded in the OPF program, we will consider that the optimal solution of the J.Lacerda A field current is violating its upper bound considered to be $1.374\ pu$.

Two approaches can be used to tackle this problem:

1. Include the field current variable in the objective function, weighted with a positive cost;

2. Represent upper and lower bounds for the field current variable.

Figure 11 shows the addition of the field current in the objective function.

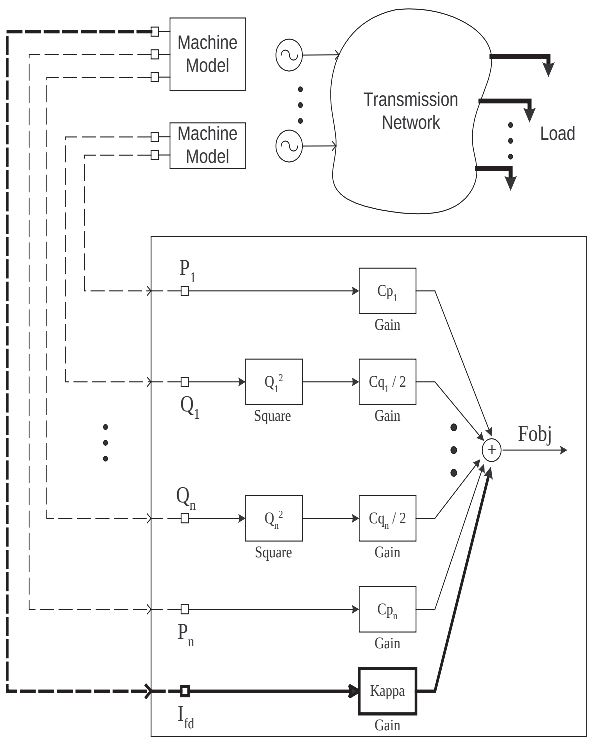

Figure 11: Objective function and synchronous machine models

Figure 11 also shows that the variables appearing in the objective function are connected to the synchronous machine models, such that any internal machine variable can be readily available for the objective function and constraints.

Figure 12 show the convergence process for the voltage magnitude of the generating units of J.Lacerda A and J.Lacerda B, utilizing the first approach.

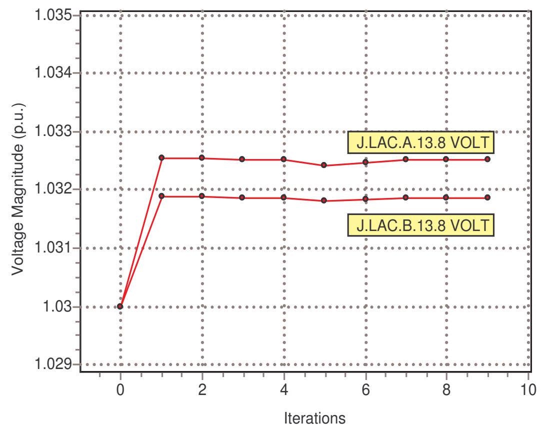

Figure 12: J.Lacerda voltage magnitude

Figure 13 shows the convergence process for the field current of the generating units of J.Lacerda A. The optimal value obtained was $1.357\ pu$, which lies below the upper bound.

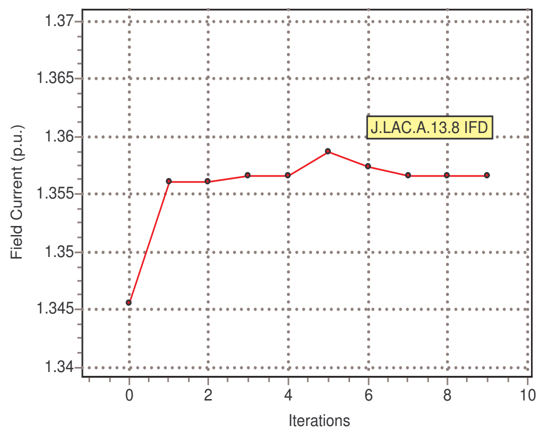

Figure 13: J.Lacerda field current

For the second approach the field current variable is constrained by upper and lower bounds as show in (19).

$$0.0 \leq I_{fd} \leq 1.374 p.u. \qquad (19)$$

Figure 14 shows the convergence process for the J.Lacerda A field current, whose value at the optimal solution is bind in the upper limit of $1.374\ pu$.

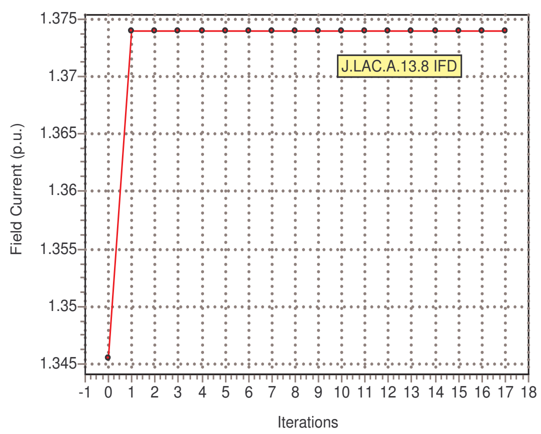

Figure 14: Limits on field current

The OPF can be formulated either as current injection or power injection balance equations. Another important feature of the formulation is the possibility for modal analysis, since the method is based on the differential-algebraic equations. Furthermore, the system eigenvalues and the lagrange multipliers can be obtained and used as sensitivities to remedial actions.

7 CONCLUSIONS

The paper presented an OPF formulation with user-defined objective functions and constrains. Despite slightly degrading the computational performance, the proposition is appealing due to the great flexibility offered by the computational framework. The optimization algorithm is based on well-known interior-point methods. The computational framework is structured with advanced concepts of object-oriented modeling.

REFERENCES

[1] A. Manzoni, G. N. Taranto, and D. M. Falcão, "A comparison of power flow, full and fast dynamic simulations," in *Proceedings of the* 14^{th} *PSCC*, Session 38, Paper 2, (Sevilla, Spain), Power Systems Computation Conference, June 2002.

[2] T. Van Cutsem and C. D. Vournas, "Voltage stability analysis in transient and mid-term times scales," *IEEE Transactions on Power Systems*, vol. 11, pp. 1173–1182, 1996.

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

[4] J. Arrillaga, C. P. Arnold, and B. J. Harker, *Computer Modeling of Electric Power Systems*. New York: Wiley, 1983.

[5] Z. S. Machado Jr, G. N. Taranto, and D. M. Falcão, "An optimal power flow formulation including detailed modeling of generators," in *Proceedings of the PSCE ('04)*, (New York City, New York, USA), 2004 IEEE PES Power Systems Conference and Exposition, October 2004.

[6] G. N. Taranto, Z. S. Machado Jr, and D. M. Falcão, "Detailed representation of capability curves of synchronous machines and control systems in optimal power flow formulation," in *Proceedings of the IX SEPOPE*, (Rio de Janeiro, Brazil), IX Symposium of Specialists in Electrical Operational and Expansion Planning, May 2004.

[7] P. A. Löf, G. Anderson, and D. J. Hill, "Voltage dependent reactive power limits for voltage stability studies," *IEEE Transactions on Power Systems*, vol. 10, pp. 220–228, February 1995.

[8] S. Granville, "Optimal reactive dispatch through interior point methods," *IEEE Transactions on Power Systems*, vol. 9, no. 1, pp. 136–146, February 1994.

[9] M. Minoux, *Mathematical Programming: Theory and Algorithms*. New York: Wiley, 1986.

[10] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer Series in Operations Research, New York: Springer-Verlag, 1999.

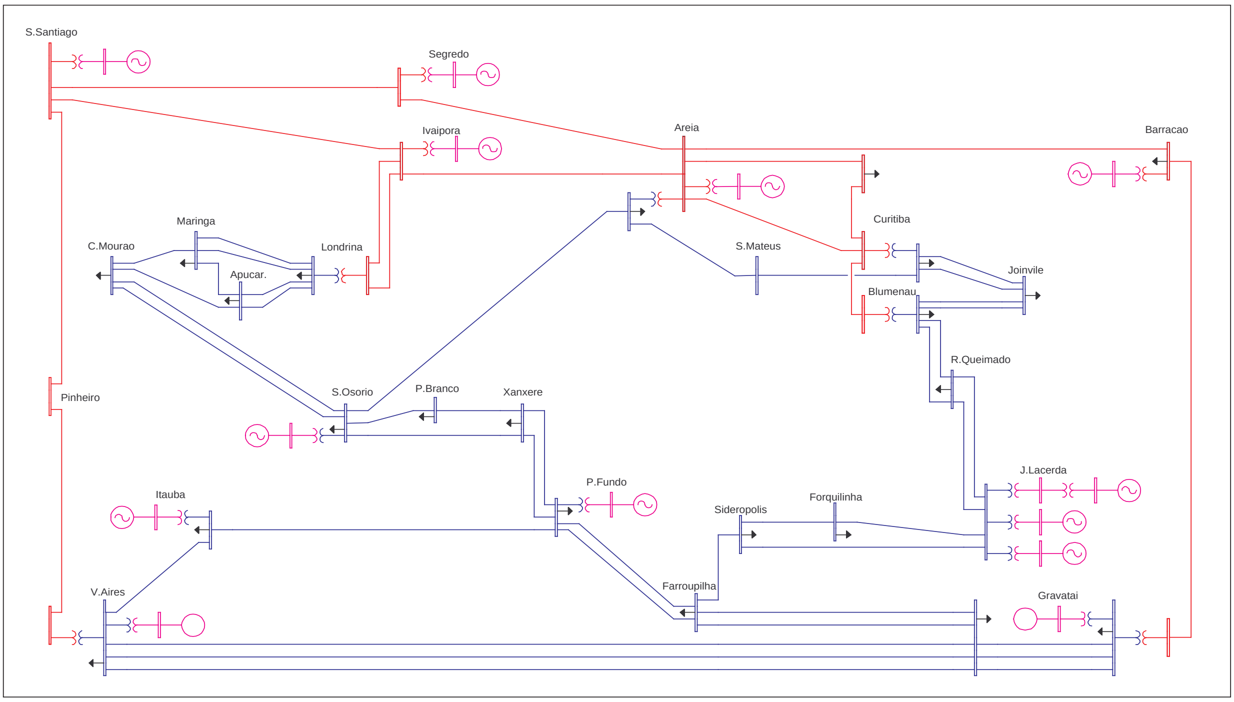

Figure 15: One line diagram of the Brazilian Southern System