

# REAL TIME PREVENTIVE ACTIONS FOR TRANSIENT STABILITY ENHANCEMENT WITH A HYBRID NEURAL NETWORK - OPTIMIZATION APPROACH

Vladimiro Miranda, J. N. Fidalgo, J. A. Peças Lopes  
INESC - Instituto de Engenharia de Sistemas e Computadores  
and DEEC/FEUP - Fac. de Engenharia da Universidade do Porto  
Largo Mompilher 22 , P- 4000 Porto, PORTUGAL  
Fax: +351.2.318692 email: vmiranda@inescn.pt

L. B. Almeida  
INESC - Instituto de Engenharia de Sistemas e Computadores  
and DEEC/IST - Instituto Superior Técnico  
Lisboa, PORTUGAL

*Abstract:* This paper reports a new approach in defining preventive control measures to assure transient stability relatively to one or several contingencies that may occur separately in a power system. Generation dispatch is driven not only by economic functions but also with the derivatives of the transient energy margin value; these derivatives are obtained directly from a trained Artificial Neural Network (ANN), using real time monitorable system values. Results obtained from computer simulations, for several contingencies in the CIGRE test system, confirm the validity of the developed approach.

Later, Yamashiro [10] proposed the definition of a preventive control strategy by embedding in the optimal dispatch problem an additional transient security constraint described by a function obtained by a perceptron pattern recognition technique.

Djukanovic et al. [11] presented an application of an ANN to be used in support of the decision making process towards the fast stabilization of multi-machine systems through the determination of generator shedding requirements when instability was detected.

## INTRODUCTION

Fast security assessment is one of the most important needs in dispatch centers during power system operation. However, a complete answer about the degree of system security requires an evaluation of the dynamic system behavior following some foreseen perturbations. Transient stability is concerned with system behavior following any large change in system operating conditions. Short-circuits, in particular, can give rise to large electromechanical oscillations among generator rotors, which may lead to system collapse.

Transient instability detection/evaluation is, however, an incomplete task and it is most important to complement it by defining fast and efficient control measures in order to assure system security.

Several methods for fast transient stability evaluation have been proposed during the last decades, by adopting namely direct methods [1,2], pattern recognition techniques [3,4], inference inductive methods using decision trees [5], and artificial neural networks (ANN) [6,7,8]. To the authors knowledge, the first attempt to define preventive control measures for the transient stability problem may be found in Saito et. al. [9], who used a trial and error approach combined with a fast transient stability evaluation based on a perceptron pattern recognition method.

Recently, we have proposed a new method for the determination of preventive control measures based in generation load redispatch driven by the derivatives of a transient energy function. These derivatives, relative to the generated powers, are estimated directly from backpropagating the derivatives of an ANN output relative to the output of each previous ANN unit. In paper [12] we demonstrated the feasibility of this technique, showing that, for a single contingency, redispatch measures could be derived driving a system into a security zone.

In this paper, the following new advances are described:

- a) a scheme to take care of several possible contingencies at the same time has been devised;
- b) one is no longer only concerned with security, but instead the problem may now be viewed as an economic dispatch problem with security constraints.

The concept "preventive measures" acquires a new meaning in the present paper: one is no longer facing a sequential process where: i) economic dispatch decisions are taken; ii) contingency analysis performed and possibilities of instability detected; and then iii) preventive actions taken to avoid such system states. Instead, the dispatch decision process presented in this paper includes now security constraints; therefore, the "preventive actions" emerge as direct consequence of the solution of the dispatch problem.

The results obtained from computer simulations, with the CIGRE test system, confirm the feasibility and demonstrate the interest of the developed approach .

## ANNs FOR FAST ASSESSMENT OF TRANSIENT STABILITY MARGINS

The need to achieve not only fast accurate measures of the transient stability degree, but also an easy way to determine preventive control measures led us to adopt ANNs. This approach was used to derive a stability index as an emulation of the transient energy margin ( $\Delta EV$ ), as described in [1,2].

The description of ANN basics is not within the scope of this paper. Nevertheless, in the next sections we refer to some characteristics of the type of ANN used, the training algorithm adopted and results obtainable.

#### ANN and Training algorithm

Feedforward ANNs and the Adaptive Backpropagation (ABP) training algorithm were used. The ABP [12,13] is based in the traditional Backpropagation (BP) [14], but instead of a fixed and unique learning rate it uses an individual adaptive learning rate for each weight. This algorithm provides a faster learning process; in some complex mapping cases, the authors experienced a reduction of up to about 1/1000 of the training epochs relative to the traditional BP, with the same initial conditions. Other techniques adopted to accelerate training are described in [12].

#### Assessment of transient stability margins

All pattern recognition approaches require the following sequence of operations: feature selection, training set generation, training procedure and performance evaluation tests.

The feature selection phase needs a special attention. From the immense number of variables that may characterize a power system, we required that the set chosen to represent system states would have the following properties:

- i) it should adequately characterize a system state - at the same time, this set should be small enough to avoid a large number of ANN parameters;
- ii) its variables should be independent;
- iii) it should include a subset of monitorable and controllable variables so that it might be possible to exercise control actions in the system.

Having in mind the known properties of certain system variables and after evaluating, in several experiments, its discriminating power relatively to a hard stable/unstable classification (based on the transient energy margin sign), the selected set of system parameters and measures was the following:

- generated active power outputs ( $P_i$ );
- emfs ( $E_i$ );
- machine inertia constants ( $M_i$ );

where  $i=1,\dots,NG$  with  $NG$  - number of generation buses.

The first two subsets are directly monitorable or available from SCADA, and the third subset is available from an EMS data base. This specification lead to discarding other variables from consideration (such as accelerating powers,

kinetic energies, etc. - transient stability variables) that would not be directly available. This engineering judgement dismissed the need to adopt feature extraction methods in selecting inputs, for small and medium scale systems.

However, the set of chosen inputs contains implicitly information needed to derive the values of these transient stability variables. If properly trained, one expects that an ANN will internally have somehow established relationships that will lead to a discriminating power similar to the one usually displayed by such now discarded variables. As reported in [12], this hypothesis has been confirmed.

Besides, experiences conducted with ANNs that used, as input, also transient stability variables have shown us no improvement in the quality of results, compared to the ones reported in this and previous papers [12]. The only noticeable (but not too important) difference has been the computing effort needed to train the ANNs.

Generated active powers and emfs are variables in any dispatch environment. However, we have used machine inertia constants also as variables to take in account different possible unit commitments: this successful approach allowed us to develop a general tool .

A great number of feasible operating situations has then been generated by considering, for a specific contingency, several load levels and different unit-commitment alternatives. For each system scenario, several feasible dispatch alternatives were generated, in a neighbourhood of the optimum. However, one must bear in mind that the basic topological configuration was kept fixed (no different or diverse network structures were examined).

A load flow has been performed for each alternative. A selected contingency has then been simulated and system stability analyzed with the Transient Energy Function method [1,2] assigning to each operating point a transient energy margin value  $\Delta EV = V_{cr} - V_{tel}$  (where  $V_{cr}$  is the transient energy function value at the stability boundary and  $V_{tel}$  is the energy function value at the fault elimination moment).

For each contingency, this set of operating situations has been divided in two: one constituting the Training Set (circa 1400 points), and the other the Test Set (circa 400 points).

For each operating point included in the training set,  $\Delta EV$  is used as the target output when training the ANN. When properly trained, the difference between the ANN output  $\phi$  and the desired target  $\Delta EV$  tends to be small. Ref. [12] presents values between 0.006 and 0.031 for the  $\Delta EV$  root mean square error obtained in the test sets, which are very good results.

#### Sensitivities

Neural networks provide a simple and efficient way to evaluate the sensitivities of the output  $\phi$  relative to the input

variables. In this case, this is equivalent to the estimation of the derivatives of the transient energy margin with respect to the chosen control variables.

As shown in [12], it is possible to calculate the derivative of the output activation  $x_j$  of an ANN neuron relative to the output activation  $x_k$  of each unit of the previous layer as:

$$\frac{-x_j}{-x_k} = f'_j w_{jk} \quad (1)$$

where  $k$  refers to any of the units that connects directly to the unit  $j$ , and:

- $w_{jk}$  - weight of the connection between units  $j$  and  $k$
- $f'_j$  - the derivative of the activation function of unit  $j$

Now, a chain rule can be applied, calculating the derivatives from the output layer back to the input. Note that if a unit feeds two or more units in the next layer, those influences must be added in the calculation. So, a recursive process to calculate the partial derivative of the ANN output transient margin  $\phi$  to ANN generated power input  $P_i$  is based on:

$$\frac{-\phi}{-x_i} = \sum_k f'_k w_{ki} \frac{-\phi}{-x_k} \quad (2)$$

where  $k$  refers to all the units in the following layer that are directly fed by unit  $i$ . Of course, when one gets to the first layer, one has  $\frac{-\phi}{-P_i}$ .

### MINIMUM INCREMENTAL COST STRATEGY

The general dispatch problem can be formalized as a typical non-linear constrained optimization problem:

$$\min C_c(\mathbf{P}) \quad (3)$$

$$\text{subj. } \mathbf{P} \in \mathbf{D} \quad (4)$$

$$\phi_i > \phi_{sec} \quad (5)$$

where  $\mathbf{P}$  is a vector of generated powers,  $\mathbf{D}$  is the domain of the feasible power flow solutions,  $C_c(\mathbf{P})$  is the generation cost function and  $\phi_{sec}$  is the stability margin to be reached, corresponding to a target transient stability margin  $\Delta EV_i > 0$ , for a selected perturbation  $i$ . As this is a general formulation, equation (5) represents a vector of  $p$  different pre-specified perturbations or contingencies.

An initial solution may be found by any optimizing dispatch technique, neglecting constraints (5). In [12], a feasible solution to the global problem, when the optimal dispatch solution violates  $\phi_i > \phi_{sec}$ , was found by a technique similar to the reduced gradient method: the gradient of the violated constraint was used to move a solution into the

feasible domain. We are now interested not only in finding feasible solutions, but also in moving to the least cost one by small changes in the optimal economic dispatch.

We have dealt with this problem (3-5) by adopting a penalty approach. The problem can be transformed into the minimization of a global cost function

$$\min C = C_c + \sum_{i=1}^p \alpha C_{\phi_i} \quad (6)$$

$$\text{subj. } \mathbf{P} \in \mathbf{D} \quad (7)$$

where  $C_c$  represents the production cost,  $C_{\phi_i}$  is a penalty function composed with the security constraint (5i) and  $\alpha$  is a weight factor (if  $\alpha$  is great, the stability margin acquires a increased importance; if  $\alpha$  is small the priority is to satisfy the production cost term).

The production cost term is well known and depends only on the sum of the individual generation costs of each generator (neglecting losses). The security penalty function had to be defined in such a way that it could satisfy the following requests:

1. it had to be continuous and derivable;
2. it should present high values for  $\phi_i < \phi_{sec}$  (unstable states) and small values elsewhere.

As we were working with ANNs, we took the elegant step of choosing, for this purpose, the symmetrical of a sigmoid:

$$C_{\phi} = \frac{1}{1 + e^{b(\phi - \phi_{sec})}} \quad (8)$$

where:

- $b$  - factor for curve smoothness control
- $\phi_{sec}$  - shifter of the center of the function

In fact, we adopted two different representations of a sigmoid, by choosing two different smoothing values  $b^-$  and  $b^+$  such that  $b^+ \gg b^-$  (typically,  $b^- = 0.05$  and  $b^+ = 10^3$ ). Therefore

$$\text{if } \phi < \phi_{sec} \text{ then } b = b^-; \quad \text{if } \phi \geq \phi_{sec} \text{ then } b = b^+$$

The effect is depicted in figure 1. In algorithmic terms, we managed to obtain the following result:

1. for an unfeasible solution, the problem behaves as an external penalty method, and solutions in successive iterations are pushed into the feasible domain;
2. as soon as a security constraint is no longer violated, the process now shows characteristics of an interior penalty method, and solutions in successive iterations are allowed to get very close to the border - but then suddenly the penalty factors grow very fast, if they move too close.

This strategy worked well with the adopted gradient iterations (see in Fig. 1 how the derivatives of the penalty

function behave), in a process similar to the one described in [12]: each algorithm step is based on the redistribution of the generated powers in every busbar.

Following a gradient technique, for each generating bus, a correction  $\Delta P_i$  is given in general by

$$\Delta P_i = h \beta_i, \quad i=1, 2, \dots, NG-1 \quad (9)$$

where

$$\beta_i = \frac{C}{-P_i} - \frac{C}{-P_{NG}} \quad (10)$$

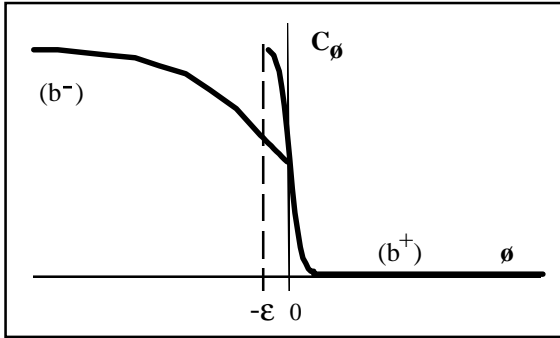


Fig. 1 - Penalty function for the security constraints

The coefficient 'h' is a step size control parameter, changed according to the evolution of the algorithm. NG is chosen as a reference bus.  $\Delta P_{NG}$  is calculated using the constraining equation derived from an incremental DC approach to the power flow problem:

$$\sum_i \Delta P_i = 0 \quad (11)$$

When differentiating equation (6), one assumes that traditional well behaved generation cost functions  $C_c$  are adopted. The derivative of  $C_{\phi i}$  is calculated via a chain rule:

$$\frac{-C_{\phi i}}{-P_i} = \frac{-C_{\phi i}}{-\phi} \frac{-\phi}{-P_i} \quad (12)$$

The value of  $\frac{-C_{\phi i}}{-\phi}$  is calculated from (8);  $\frac{-\phi}{-P_i}$  is obtained from the ANN, as described in a previous chapter.

The constraints concerning the limits of power to be generated in each bus ( $P_i^{\min}$  and  $P_i^{\max}$ ) are checked and dealt with during the iterative procedure, so that the dispatch corrections always lead to feasible load flow solutions.

The gradient of the primitive objective function and the gradients of the penalty functions are used to conduct the iterations. In order to enhance this effect, for small excursions of a feasible solution into the unfeasible domain, in a neighborhood  $\epsilon$  of the border, we kept the value  $b^+$  instead of changing at once to  $b^-$  (see figure 1).

In any case, like in other problems that include compromising terms, the value of parameter  $\alpha$  had to be carefully tuned, in order to get better and faster achievement of

the desired stable economic state. Ideally, one should choose a decreasing sequence of  $\alpha$  values (or an increasing sequence of  $b^+$  values) for the cases when  $\phi > \phi_{\text{sec}}$ , assuring the convergence of the process. However, we found that, with the typical  $b^+$  values mentioned above, this was not needed because the improvements were negligible; besides, this means that the operating point will not lie exactly on the border of the stability domain, but just slightly inside, which is rather convenient.

## DEALING WITH MULTI-CONTINGENCIES

When one considers only one contingency ( $p=1$ ), it is likely that one will easily find a solution for the problem (6-7). However, one may be looking for an operating point that may be classified as secure for two or more contingencies: this raises the problem of having eventually an empty feasible stability domain.

If this situation arises, one must have at hand a set of criteria in order to take a decision which inevitably will imply a certain level of risk: no matter what operating point is chosen, it will always be unstable for some events.

Simple criteria could be as follows: choose the operating point with lower cost that, for instance:

1. maximizes the number of contingencies for which the system is classified as secure, or
2. assures stability for the most probable contingencies.

For criteria such as these, the stability domain now considered may be disjoint, composed of unconnected parts (for example, the stability domain for  $p$  pre-specified contingencies may be empty; however, one may find two non-empty stability domains for two sets of  $p-1$  out of those  $p$  events). This raises up algorithmic difficulties in reaching a final solution.

A deeper analysis would also lead us, in this case, to deal with concepts of trade off between stability levels and operation cost, assuming that, in general, higher stability levels imply dispatch decisions with increasing costs (but not always). This is a rich field for further developments and will allow the specification of new operation policies.

### Solving a two contingency security problem

We present an algorithm to deal with the problem of considering two contingencies in the formulation of (3-5). In a pseudo code, it can conceptually be described by

```

FIND the feasible security domains  $D_1$  and  $D_2$  of each of the two
contingencies.
If  $D_1 = D_2 = \{\}$  then NO SECURE STATE;
else if  $D_1 = \{\}$  and  $D_2 \neq \{\}$  then OPTIMIZE for  $D_2$ ;
else if  $D_1, D_2 \neq \{\}$  then FIND  $D_1 \cap D_2$ ;
if  $D_1 \cap D_2 \neq \{\}$  then OPTIMIZE for  $(D_1, D_2)$ ;

```

```

else    OPTIMIZE for D1;
        OPTIMIZE for D2;
        CHOOSE one of the two solutions,
            according to a specified criterion.

```

When implementing this algorithm in a program, the FIND task was performed by the optimization algorithm described earlier, starting with one of the contingencies; then, if stability was reached, the second contingency would be added.

For a small number of contingencies (say, up to 10), this algorithm is easily generalized. For instance and referring to the above algorithm, whenever  $D_1 \cap D_2 \neq \emptyset$ , this combined set may be entered in the algorithm and searched for a new security domain in conjunction with  $D_3$  from a new contingency. The general principle is to search for a state that may maximize the number of contingencies for which secure dispatch solutions may be found and, at the same time, take in account the trade off between operating cost and increased security.

For a large number of contingencies, a simple exhaustive search may become a heavy computing burden and therefore require a more complex approach, not dealt with in this paper. A search procedure (yet to be developed) should be devised, guided by trade off values, that would avoid simple enumeration of the combination of contingencies.

## NUMERICAL RESULTS

### Example data

The CIGRE test system is used to illustrate the results obtained with the new approach. Figure 2 shows the single line diagram of the system, also adopted by several other authors, allowing comparison of results. Numerical data can be found in Peças Lopes et al. [4].

In the following examples, we set  $\phi_{sec} = 0$ , i.e., we accept a solution as being secure for a contingency when its stability index is positive. This is acceptable here, as a Transient Energy Margin approach was adopted, which is considered a conservative criterion. However, this is not limitative of the technique and, of course, positive  $\phi_{sec}$  values could have been used if considered adequate.

Table 1 describes two events whose simulation results are presented and discussed below.

In order to obtain the corresponding training sets several load levels were considered, with the total system load varying linearly between 1190 MW and 1850 MW. Each operating point was thus initially characterized by a set of 21 variables:  $P_i$ ;  $E_i$ ;  $M_i$  ( $i=1, \dots, 7$  for the 7 generating buses).

Table 1 - Simulated contingencies

contingency nr.	3phase fault on bus	elimin time (s)	line switched
1	3	0.37	3-2
2	5	0.31	no

### Neural network architecture

The ANN used for single contingency studies has 4 layers:

- an input layer with 21 units
- a first hidden layer with 4 units
- a second hidden layer with 2 units
- an output layer with 1 unit.

This structure was chosen as the one with the best results among several tried. The quality of the results is enough guarantee that an adequate choice has been made, for this case.

For two contingency studies, we have merged two ANN for two separate contingencies by making them share the units in the first (input) layer. This merged ANN has now two units in the output layer, giving  $\phi$  for each contingency.

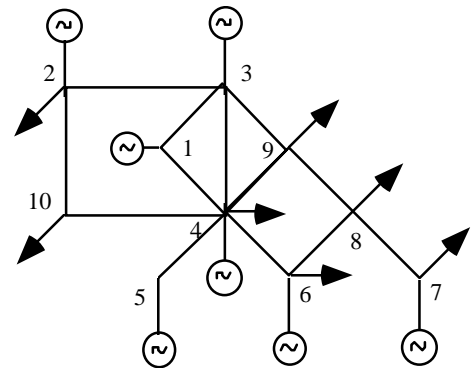


Fig. 2 - Single line diagram of CIGRE test system  
Single contingency studies

Table 2 presents numerical data and results obtained for contingency nr. 1. The last three columns show the original optimal generated powers ( $P_o$ ), the new values for the same powers after the preventive control algorithm ( $P_{con}$ ) from ref [12] and the new values obtained with the approach described in this paper ( $P_{opt}$ ).

Table 2 - ANN inputs and changed P (after the preventive control algorithm application)

bus	E(pu)	M(pu)	$P_o$ (pu)	$P_{con}$ (pu)	$P_{opt}$ (pu)
1	1.171	.060	2.879	2.926	2.915
2	1.182	.060	2.183	2.189	2.192
3	1.051	.076	3.198	2.883	2.919

4	1.117	.095	1.985	2.030	2.035
5	1.113	.064	3.146	3.181	3.207
6	1.120	.068	2.254	2.310	2.294
7	1.084	.066	2.446	2.572	2.528

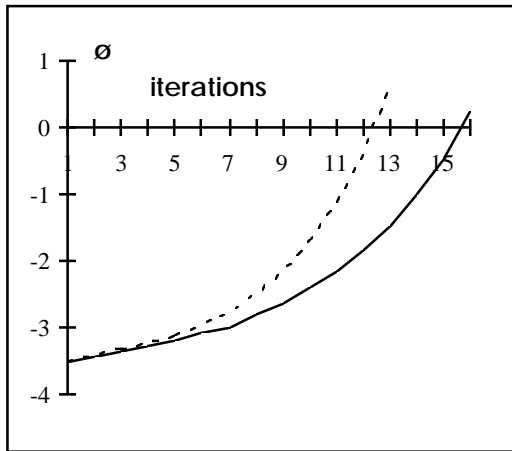


Fig. 3 - Evolution of the stability index  $\phi$  along the algorithm iterative process, with (continuous line) and without (dashed line)  $C_c$  in the objective function

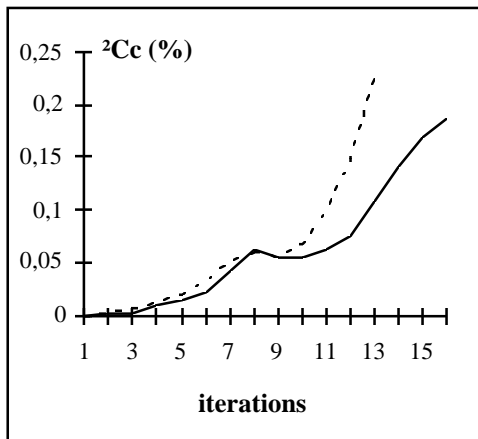


Fig. 4 - Evolution (along the algorithm iterative process) of the generation cost increment in relation to the optimal dispatch, with (continuous line) and without (dashed line)  $C_c$  in the objective function

The security index values obtained in these three situations are:

$$\phi_0 = -3.51; \quad \phi_{con} = 0.67; \quad \phi_{opt} = 0.24$$

Figures 3 and 4 display a comparison in the behavior of those two algorithms, in terms of security index  $\phi$  and generation cost  $C_c$ . The final state reached by the new algorithm has a lower cost and a security index  $\phi$  closer to 0, although some few more iteration steps were required.

#### Two-contingency example

The following example illustrates the simultaneous consideration of contingencies nr. 1 and 2. The departure point is, as usual, the optimal economic dispatch. Continuous lines represent the solution trajectory obtained with the application of the proposed algorithm, while dashed lines represent the solution trajectory when the cost part of the objective function is neglected (i.e., security is the only concern).

It is interesting to notice, in fig. 5 - dashed line, that when one tried to increase the stability index exclusively for  $\phi_1$ , in the beginning of the process, the index  $\phi_2$  became first better and then worse, which means that security conditions for two contingencies are not necessarily always divergent.

The presence of a cost component in the gradient vector clearly changed the path followed by the algorithm. Figure 6 shows that it was possible to find a secure solution for both contingencies, with only a marginal increase in the overall operation cost. Although in security index terms the final solutions are not very different, a large relative difference in the generation cost has been found.

For illustration purposes, only a two contingency example has been presented. However, the authors report that they have applied the same technique to the same system, considering an optimization with four security constraints for four contingencies, with the same kind of results, confirming the robustness of the approach.

### CONCLUSIONS

The work reported in the paper demonstrates how improved and simplified the power system dynamic security analysis can turn into, in small or medium sized networks, when combining neural networks with some classical optimization approaches. Straight but effective methods can be implemented not only for transient stability index evaluation but also for the development of preventive control policies embedded in dispatch decisions.

The algorithm presented shows a way of reaching system stable states from unstable ones. Moreover, the suggested approach allows not only to reach security but also do it with a minimal incremental cost.

In all examples studied, a very small change in the economic dispatch has proven sufficient to provide a new stable state.

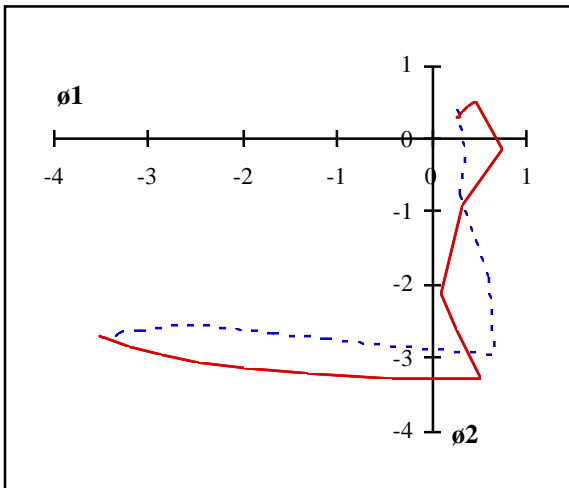


Fig. 5 - System security evolution trajectory under the multi-correction algorithm, with (continuous line) and without (dashed line)  $C_c$  in the objective function

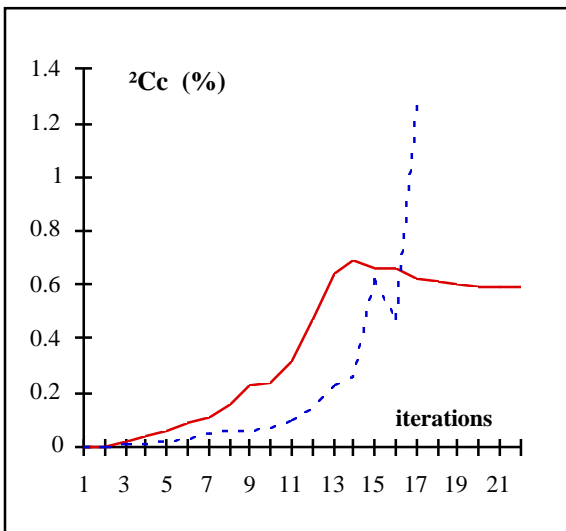


Fig. 6 - Evolution of the cost increment in relation to the optimal dispatch during the iterative process, with (continuous line) and without (dashed line)  $C_c$  in the objective function

The process generalization for the case of several contingencies has originated very interesting results, showing the feasibility of the adopted multi-correction approach.

Notice that the authors are not defending that the reduced gradient technique should be the one to be adopted: one is aware of the drawbacks of simple gradient techniques. However, the work does prove that the approach is both feasible and computationally interesting, therefore opening a path of further research on which optimization algorithms are best suited to deal with the problem, as formulated:

*one has analytical expressions for the objective function and for the domain of feasible power flows, and although not having an analytical expression for the security*

*constraints, one may nevertheless obtain, via the neural network, good numerical estimations of their gradient.*

The main contributions of this work can thus be summarized as follows:

- a) Preventive measures in terms of generation redispatch, in case of instability detection, can be suggested by exploiting the possibility of getting, directly from a trained ANN, the sensitivities of the stability margin relative to the input variables - and this is done in an integrated dispatch environment, enabling one to obtain a secure solution at minimal incremental operation cost; as a result of using ANNs, computing time is negligible allowing one to have available a real time tool.
- b) The generalization of the procedure for the multi-contingency case is possible and quite straightforward for a small number of contingencies.
- c) New ways of looking at the stability enhancement problem are now at hand - one must start discussing matters such as trade offs between security levels and operation costs and how to provide decision aid in the case of multiple acceptable solutions, that can now be calculated.

This technique is already being applied to small sized isolated networks, such as in islands with wind/diesel generation (Lemnos, Greece) in a project funded by the EC under the JOULE program [15]. Its application to large scale networks needs to be further studied.

### REFERENCES

- [1] M. A. Pai, *Energy function analysis for power system stability*, Kluwer Academic Publishers, 1989
- [2] T. Athay et al., "Transient stability analysis", *Conf. on Systems Engineering for Power: Emergency Operating State Control - Section IV*, Davos, Switzerland, 1980
- [3] H. Hakimmashadi, *Fast Transient Security Assessment*, Ph. D. Thesis, Purdue University, 1982
- [4] J. A. Peças Lopes et al., "Fast transient security assessment by pattern recognition techniques", *Proc 9th PSCC*, Lisbon, Portugal, 1987
- [5] L. Wehenkel et al., "An artificial intelligence framework for on-line transient stability assessment of power systems", *IEEE Trans. PWRs*, Vol. 4, nr. 2, May 1989
- [6] D. J. Sobajic and Y. H. Pao, "Artificial neural-net based security assessment for electric power systems", *IEEE Trans. on PWRs*, Vol. 4, nr. 1, February 1989
- [7] Y. H. Pao and D. J. Sobajic, "Combined use of unsupervised and supervised learning for dynamic

- security assessment", *Proc. PICA' 91*, Baltimore, USA, 1991
- [8] A. B. R. Kumar et al., "Neural network for dynamic security assessment of large-scale power systems: requirements overview", *Proc. 1st Int. Forum on App. of Neural Networks to Power Systems*, Seattle, July 1991
- [9] O. Saito et al., "Security monitoring systems including fast transient stability studies", *IEEE Trans. on PAS*, Vol. 94, n° 5, Sept/Oct. 1975
- [10] S. Yamashiro, "On line secure-economic preventive control of power systems by pattern recognition", *IEEE Trans. on PWRs*, vol PWRs-1, n.3, Aug 1986
- [11] M. Djukanovic et al., "Neural net based determination of generator shedding requirements in electric power systems", *IEE Proc. C - Vol 139*, nr. 5, Sep 1992
- [12] J.N. Fidalgo et al., "Fast assessment of transient stability margins by a neural network approach", *Proc. 11th PSCC*, Avignon, France, Aug 1993
- [13] F. M. Silva, L. B. Almeida, "Acceleration Techniques for the Backpropagation Algorithm", in *Neural Networks*, Almeida and Wellekens eds., Springer-Verlag, 1990.
- [14] D.E. Rumelheart et al, "Learning Internal Representations by Error Propagation", in *Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, Rumelheart and MacClelland eds, MIT Press, Cambridge, MA, 1986
- [15] J.A. Peças Lopes et al., "Dynamic Security Assessment Using Pattern Recognition and Neural Networks - Results in the Lemnos Power System", *Intermediate Report to the CEC*, project JOU2-CT92-0053, Nov 1993

### **BIOGRAPHIES**

**Vladimiro Miranda** was born in Oporto, Portugal, on March 11, 1955. He received his Licenciado and Ph.D. degrees from the Faculty of Engineering (FEUP) of the University of Oporto in 1977 and 1982, all in electrical engineering. In 1981 he joined FEUP and currently holds the position of Associate Professor. In 1985 he joined also INESC - a Research Institute for Systems and Computers, and holds currently the position of Project Manager - Head of Information and Decision in Energy Systems Area (IDEIA).

**J. N. Fidalgo** was born in Oporto, Portugal, on August 30, 1961. He received his Licenciado degree from FEUP in 1984, in electrical engineering, when he joined INESC as a researcher. He is presently preparing his PhD. thesis. His research interests are neural networks and their application to power systems.

**J. A. Peças Lopes** was born in Sertã, Portugal, on May 23, 1958. He received his Licenciado and Ph.D. degrees from FEUP in 1981 and 88, in electrical engineering. In 1981 he joined FEUP and currently holds the position of Assistant Professor. He is also a researcher at

INESC, in the IDEIA group. His research interests are power system security assessment and dispersed generation system analysis. He is a member of the CIGRE TF 38-06-06 on "Application of Artificial Neural Networks to Power Systems", and is an IEEE member.

**J. B. Almeida** was born on April 1950, in Lisbon. He received his graduation degree in Electrotechnic Engineering in 1972 from Technical University of Lisbon and the Ph.D. degree in 1983 from Technical University of Lisbon. Dr. Almeida is presently Associate Professor in the Technical Superior Institute (IST) in Lisbon and researcher at INESC where he holds currently the position of Project Manager. His current areas of interest are: Neural Networks and Speech Processing.