

# SECURITY BOUNDARY VISUALIZATION FOR SYSTEMS OPERATION

James D. McCalley Shimo Wang  
Qianglin Zhao Guozhong Zhou  
Iowa State University  
Ames, Iowa

Roger T. Treinen  
Consultant  
San Francisco, California

Alex D. Papalexopoulos  
Electric Supply Systems  
Pacific Gas and Electric Company  
San Francisco, California

**Abstract:** This paper presents a security assessment approach for operational planning studies that provides the operator with accurate boundary visualization in terms of easily monitored precontingency information. The approach is modeled after traditional security assessment procedures which result in use of a *nomogram* for characterizing the security boundaries; these procedures are common among many North American utilities today. Therefore, the approach builds on what is already familiar in the industry, but it takes advantage of computer automation and neural networks for generating and understanding large data bases. The appeal of the approach is threefold:

it provides increased accuracy in boundary representation, it reduces the labor traditionally required in generating security boundaries, and the resulting boundaries, encoded in fast, flexible C subroutines, can be integrated into energy management system software to provide the operator with compact, understandable boundary illustration in real time. These improvements are of particular interest in securely operating transmission systems close to their limits so as to fully utilize existing facilities.

**Keywords:** Electric power systems, operations planning, automatic security assessment, neural networks, boundary visualization.

## 1.0 INTRODUCTION

In the past, North American transmission systems, owned by regulated, vertically integrated utility companies, have been designed and operated so that conditions in close proximity to security boundaries were not frequently encountered. One reason for this was that the load patterns and consequently the flow directions were fairly predictable and not significantly different from that for which they were originally designed. Another reason is that companies could usually justify construction of new facilities that could alleviate operating constraints if they could show reliability would be compromised otherwise. In the future's "open access" environment, operating conditions will be more frequently in close proximity to security boundaries. This is because transmission usage is increasing in sudden and unplanned directions, and competition, coupled with regulatory and environmental requirements, has significantly inhibited construction of new transmission facilities.

In the future, buyers and sellers will be operating autonomously instead of under centralized dispatch and will be able to access markets that are beyond the borders of the local utility's control area. The flow directions will

consequently not be dependent on only local load patterns. One implication of the change in the nature and amount of transmission usage is that security assessment must be accurate, and it must be presented to the operator clearly and compactly to allow secure operation in close proximity to system boundaries.

We have developed an approach to performing security assessment studies that builds on a common procedure used by many utility companies in North America [1, 2, 3]. In Section 2, we review the traditional approach to boundary characterization, point out its deficiencies, give an overview regarding our proposed improvements, and formalize the boundary characterization problem. Section 3 describes criteria for selecting precontingency information for characterizing the boundary. Section 4 describes software developed for automating the computer simulations required in performing security studies and discusses a neural network design and training procedure used for mapping precontingency information into a postcontingency performance level. Section 5 describes an automatic boundary visualization algorithm; an example is given in Section 6, and conclusions are drawn in Section 7.

Artificial intelligence techniques which have been applied with some success to security assessment include decision trees, neural networks, and expert systems. Wehenkel and colleagues have significantly contributed to security assessment techniques via their work in applying decision trees [4, 5, 6, 7]. This work has resulted in an approach which allows fast and accurate classification of an operating point (secure, not secure). Similar investigations by others are reported in [8, 9, 10, 11, 12]. Many researchers have also used neural networks for security assessment; a representative sample of this work includes [13, 14, 15, 16, 17]. When using neural networks in security studies, it is required, at least for large systems, that the computer simulations be automated in order to generate the necessary training data set. Therefore, many of the previous references also make mention of some form of an "expert system;" in addition, [18] reports on a more advanced expert system of this nature, and [19] provides a literature and industry survey of expert system applications and practices in power engineering. Finally, we mention that a few papers have reported on integration of various AI techniques for security assessment studies; among these are [20, 21, 22].

## 2.0 SECURITY BOUNDARY CHARACTERIZATION

In practice, security assessment is performed in two different time frames. In the short-term (minutes to hours ahead), real-time assessment software performs contingency analysis for the current operating point; the primary goal is to *classify* the operating point as secure or insecure. In the far-term (days to months ahead), computer studies are conducted off-line by an analyst with the primary goal being to identify and illustrate *boundaries* (or limits) of operation associated with well-understood security problems,

for use in real time by the operator. The resulting boundary illustrations are two dimensional graphs called nomograms. They enable the operator to (a) classify an operating point as secure or insecure, and (b) provide information on boundary proximity (how close the operating point is to the boundary) and on control actions (how much to adjust one or both operating parameters corresponding to the axes of the nomogram). The price to pay for this additional information, as we shall see, is that boundary representation is approximate. The inaccuracy associated with these approximations have been acceptable in the past but will not be acceptable in the new competitive environment.

## 2.1 Procedure for Nomogram Development

Nomogram development results in identification and illustration of the boundary between secure and insecure regions of operation for the single most limiting (most restrictive) contingency. Illustration of this boundary is done graphically using coordinate axes where each axis corresponds to a *critical parameter*. A critical parameter, selected specific to a particular contingency and resulting security problem, is a precontingency parameter such as a voltage magnitude, MW flow, generation level, or load level, which can be monitored by the operator in real time, and which is a good predictor of the postcontingency performance level of the system for the specified security problem should the contingency occur.

The postcontingency performance level is quantified by a *performance measure* for the security problem and the specific contingency. Selection of the performance measure is dependent on the type of security problem being studied. Typical performance measures for the most common security problems are given below:

- Thermal Overload: amperes or MVA on the overloaded circuit;
- Voltage Out of Limits: voltage at the violated bus;
- Voltage Instability: MVAR or MW “distance” to the bifurcation (nose) point of a QV or PV curve;
- Transient Instability: voltage dip, energy margin, or critical clearing time; and
- Oscillatory Instability: damping ratio or real part of eigenvalue corresponding to unstable mode.

In developing a nomogram, the analyst first chooses two critical parameters (at least one of which is controllable), the values of which will be represented by the nomogram coordinate axes; we denote these as  $x_1$  and  $x_2$ . All other critical parameters are then set to selected values within a typical operating range. Some precontingency parameters, not included in the critical parameter set used to characterize the boundary, may, however, influence the postcontingency system performance. Therefore, these parameters are set to constant values biased to be conservative with respect to the influence on the performance measure. Other noninfluential parameters are set according to the season and loading conditions assumed for the study, e.g., summer peak, winter partial peak, etc.

Points on the nomogram curve are identified by repeating computer simulations, varying one critical parameter,

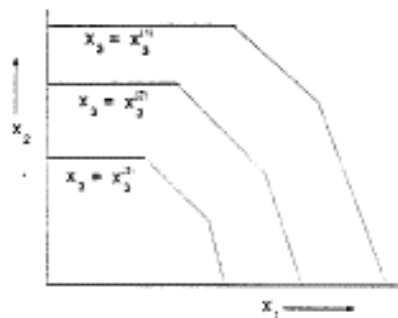


Figure 1: Nomogram curves for three critical parameters

say  $x_2$ , while holding constant the second variable,  $x_1$ . If the relationship between the performance measure and  $x_1$  is fairly linear, interpolation and/or extrapolation helps to “zero-in” on a boundary point using just three or four simulations: highly nonlinear relationships may require more simulations. Repeating this procedure for selected values of  $x_1$  provides enough boundary points to draw the nomogram curve. In the case where there is a third critical parameter, a different nomogram curve is drawn for each value of this third critical parameter so that the result is a family of nomogram curves, as illustrated in Figure 1. Inclusion of a fourth critical parameter requires a distinct family of nomogram curves (i.e., a new “page”) for each new value of the fourth critical parameter. Inclusion of a fifth critical parameter requires a distinct family of pages for each new value of the fifth critical parameter, and so on.

## 2.2 Approximations in Nomogram Development

There are two main approximations made in nomogram development which can ultimately result in inaccurate boundary characterization when the nomogram is used by the operator.

*Linear Interpolation Between Points:* Because of labor requirements, the simulation procedure described in the last section is normally used to obtain only a very few points on the boundary; indeed, often only the “corner points” are obtained. The remaining portions of the boundary are obtained by drawing a straight line through the computed points. We improve on this approximation by automating the security assessment process so that the number of points generated is only limited by computer availability. We also use an interpolation tool that has the capability to recognize and model nonlinear portions of the boundary very well - the artificial neural network.

*Insufficient Information Contained in Critical Parameters:* Nomogram development usually limits the number of critical parameters to five or less, even when there are other parameters known to be influential, for two reasons. First, having more critical parameters requires performing more simulations. Second, it is difficult to compactly represent the information to the operator if there are more than five critical parameters. For example, if the fourth and fifth parameters have six different levels of interest, the operator would require 36 pages of three parameter nomogram curves. Limiting the number of critical parameters may

mean that the information content of the parameters that are used is insufficient for accurately predicting the performance measure. In our approach, we have the capability to perform large number of simulations with little attention from the analyst, and the neural network provides compact boundary characterization for any number of critical parameters.

### 2.3 Formalization of Boundary Characterization

We denote the critical parameters using the vector  $\underline{x} = [x_1, \dots, x_M]^T$ , the performance measure by  $R$ , and we choose a *threshold* value  $R_0$  that delineates between acceptable and unacceptable performance levels. For example,  $R_0$  would be the thermal rating of the circuit at risk for overloading. For a given security problem caused by a particular contingency, we desire to obtain a relationship between the critical parameters and the performance measure  $R = f(x)$  that provides prediction of the contingency effects using knowledge of only precontingency conditions. The security boundary is given when  $R = R_0$ .

The security boundary can be identified in terms of the critical parameters  $\underline{x}$  as the solution to the equation  $f(x) - R = 0$  for  $R = R_0$ . Solutions to the equation for other values of  $R$  are not of interest, because once the boundary is identified, security assessment for any operating point not on the boundary can be given as the “distance,” in terms of precontingency parameters, between the current operating point and the boundary. Assessment in terms of precontingency parameters, as opposed to assessment in terms of the performance measure (a postcontingency quantity), is more meaningful to the operator.

### 3.0 SELECTION OF CRITICAL PARAMETERS

Critical parameter selection has relied traditionally on experience, judgement, and engineering insight into the problem under study on the part of the analyst, perhaps enhanced by simulation-based sensitivity analysis where one varies a critical parameter candidate and observes the effect on the performance measure. A tool to assist the analyst in this selection is needed<sup>1</sup>, and possibilities include applying statistical methods such as those used in [4, 5, 6, 7] and genetic algorithms to large databases of the type described in Section 4.0. Such a tool would require a set of criteria for defining a satisfactory critical parameter set for a particular security problem. We set forth the criteria used in our work to date; however, in this paper, we rely on traditional engineering judgement in applying it.

For the purpose of critical parameter selection, operating parameters may be classified as *independent* or *dependent*. A parameter is independent if it is included in the input data to a power flow program: examples include MW injection or voltage magnitude at a type PV bus or load level (MW or MVAR injection) at a type PQ bus. A parameter is dependent if it is computed as a result of a power flow program solution: examples include bus voltages at a type PQ bus or line flows.

Independent critical parameters may be further subdivided according to operator controllability. MW injections and voltage magnitudes at PV type buses are controllable independent parameters; load levels at PQ type buses are noncontrollable independent parameters.

The critical parameter set must satisfy the following criteria:

- *Set Sufficiency*: The parameters must contain sufficient information to allow prediction of the postcontingency performance measure within a desired accuracy for all operating conditions within the study scope.
- *Set Cardinality*: The critical parameter set should be chosen as the set of minimum size which satisfies the set sufficiency criterion.
- *Operating Point Controllability*: At least one critical parameter within the set must be controllable by the operator so that the operating point can be adjusted, with respect to the boundary, using preventive actions.

In addition, each parameter included in the set must satisfy the *information contribution requirement*, i.e., each parameter must contain some information, with respect to the performance measure  $R$ , not contained by any other parameter or set of parameters in the critical parameter set. If the information content of a critical parameter is completely redundant with that of the remaining parameters, its inclusion in the set is not necessary; further, its inclusion could inappropriately desensitize neural network output to changes in other parameters. The amount of additional information contained within a dependent parameter, with respect to the performance measure, is an indication of whether this parameter is a good candidate for inclusion in the critical parameter set.

### 4.0 SECURITY ASSESSMENT AUTOMATION AND NEURAL NETWORK TRAINING

We have developed an Automated Security Assessment Software (ASAS), illustrated in Figure 2, which we use to generate a large database containing data characterizing precontingency operating conditions and corresponding system performance for one specific contingency. The simulation tool interfaces with the other ASAS software in a modular way so that one can replace it with software appropriate for analysis of the problem under study; in the thermal overload example described in Section 6, the simulation tool was a power flow program.

We describe two salient features of ASAS pertaining to choosing the operating points for which contingency simulations are performed. In describing these features, we assume that there are  $M_z$  independent critical parameters  $\underline{z}$  (a subset of  $\underline{x}$ ) chosen *a priori*. For each operating condition simulated, values for  $M_z - 1$  parameters  $z_1, z_3, \dots, z_{M_x}$  are selected in a *structured randomized* fashion, and the value for one parameter  $z_2$  is selected in a manner which results in operating points being centered, although not clustered, along the boundary. We define a *state* as a unique choice of values for  $z_1, z_3, \dots, z_{M_x}$ .

*Boundary Centered Data Generation*: For each state, a series of simulations are performed by adjusting  $z_2$  according to a secant root finding method [23, pp. 248-251] so that

<sup>1</sup> Such a tool should *enhance* understanding of the security problem on the part of the analyst and not replace it. In fact, it should be emphasized that no part of this work is intended to relieve the analyst from conceptually understanding the influences of the various operating conditions on the postcontingency system performance.

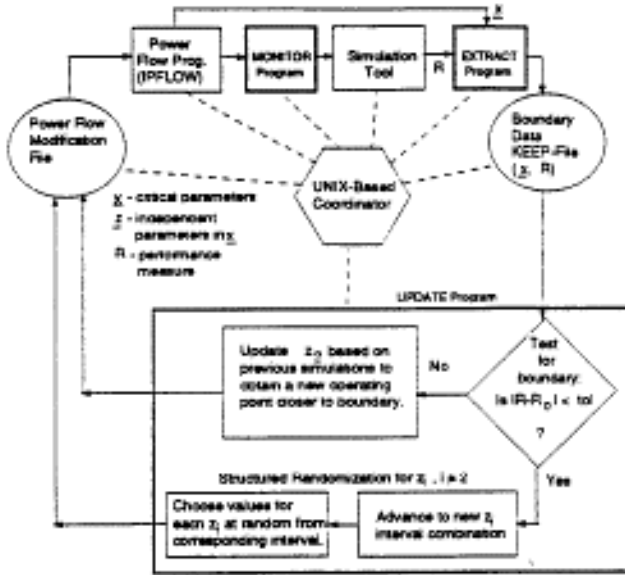
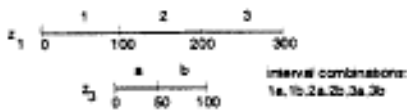


Figure 2: Automatic Security Assessment Software Package; dark blocks are C-programs.

the last simulation results in a performance measure that is within a certain tolerance of the threshold. In other words, for each state,  $z_2$  is adjusted successively to move the operating conditions closer to the boundary. From experience, 3 to 5 simulations are usually required; we define the average number of simulations required per state as  $K_{avg}$ . This action, represented by the inner loop in Figure 2, causes the operating conditions and corresponding performance measures in the database to center, although not cluster, about the boundary. This ultimately causes the neural network mapping described below to retain high accuracy for operating points close to the boundary but to slightly decrease in accuracy as operating points become more distant from the boundary. Loss of accuracy for points distant from the boundary is not of concern because only solutions on the boundary are revealed to the operator (See Section 2.3).

*Structured Randomization:* A value for each independent critical parameter  $z_i$ ,  $i = 1, 2$  characterizing a state is chosen at random from a specified interval in the range  $z_{i,min} \leq z_i \leq z_{i,max}$ , where there are  $n_i$  designated intervals for each  $z_i$ , with each interval spanning  $(z_{i,max} - z_{i,min})/n_i$ . The number of states is the number of interval combinations, given by  $N_c = \prod_{i=1, i=2}^{M_z} n_i$ . A simple case is illustrated below, where  $n_1 = 3$  and  $n_2 = 2$  such that there are  $3 \times 2 = 6$  interval combinations. Corresponding states, where the values for each parameter are chosen at random from the designated interval, might be (11,41), (85, 71), (130, 12), (174,58), (288, 30), (202, 75).



A “step by step” advancement through all interval combinations is deployed, corresponding to the outer loop of Figure 2. This approach captures two benefits. First, the “step by step” advancement through all interval combinations ensures that simulations are conducted for a uniform sample of operating conditions. Second, random selection of parameter values from each interval, for a given number of points, provides for higher resolution for each parameter. Without randomization, i.e., in using a “step-by-step” advancement through combinations of parameter values, which amounts to predefining the states instead of the intervals, we have found that, for a given number of simulations, neural network accuracy is diminished. The number of intervals  $n_i$  for each parameter  $z_i$  are chosen to ensure that the total

running time is not excessive:  $N = K_{avg}N_c < N_{max}$  where  $N$  is the number of simulations to be conducted, and  $N_{max}$  is a threshold determined by allowable computer run time.

The database generated by ASAS provides the data used in training an artificial neural network (NN) to predict postcontingency performance given precontingency information, for a single contingency. Therefore, NN inputs are the critical parameters, and the NN output is the post-contingency performance measure  $R$ . We use a commercial software package called *Predict* [24] to train the NN. In this software, network structure (number of layers and neurons per layer) is optimized during training using a cascade method of network construction where hidden nodes are added one at a time [25, 26]. Following each addition of a neuron, the network is trained using a back propagation learning rule.

*Predict* has a feature that provides the user with C-code characterizing the input-output mapping performed by a trained neural network. Given that the critical parameters are denoted by the vector  $x$ , the postcontingency performance measure by  $R$ , this C-code evaluates  $R = f(x)$  and may easily be called from another C or Fortran routine.

## 5.0 BOUNDARY VISUALIZATION

We define the independent critical parameters as  $\underline{z}$  and the dependent critical parameters as  $\underline{y}$  so that  $\underline{x} = (\underline{z}, \underline{y})$ ,  $\underline{z} = [z_1, \dots, z_{M_x}]$  and  $\underline{y} = [y_1, \dots, y_{M_y}]$ . In what follows, we describe an algorithm that uses the NN mapping function

$$f(\underline{x}) - R_0 = 0$$

to produce an illustration of the boundary in the plane defined by  $z_{1,min} \leq z_1 \leq z_{1,max}$  and  $z_{2,min} \leq z_2 \leq z_{2,max}$  with the remaining independent critical parameters  $z_i$ ,  $i = 3, \dots, M_z$  held constant<sup>2</sup>.

The algorithm is initialized from the current operating point<sup>3</sup>  $\underline{x}^{(0)} = (\underline{z}^{(0)}, \underline{y}^{(0)})$ . If dependent parameters are used in characterizing the boundary, then variations in  $z_1$  and  $z_2$  that are made when drawing the boundary must be used to update  $\underline{y}$ . We assume that simple, approximate expressions may be developed to perform this update. We denote these expressions as  $\underline{g}_y(\underline{y}^{(0)}, \Delta z_1, \Delta z_2)$ <sup>4</sup>, where  $\Delta z_i$  is the change

<sup>2</sup> The algorithm can be used to characterize the boundary in terms of any of the independent critical parameters by redefining  $z_1$  and  $z_2$ . However, at least one of these should be controllable to provide the operator with corrective action guidance.

<sup>3</sup> When used in the control room, the current operating point is retrieved from the state estimation routine. The algorithm would also be useful in operations planning; in this case, the algorithm is initiated from a power flow solution.

<sup>4</sup> Most commonly, independent critical parameters are bus injections and dependent critical parameters, if included at all, are either voltage magnitudes or circuit real power flows. Update functions may be developed in either case using linear sensitivity factors that can be obtained from the Jacobian matrix of a power flow program.

in  $z_i$  from the value  $z_i^{(0)}$  from which the algorithm was initialized, i. e.,

$$\Delta z_i = z_i^{(k)} - z_i^{(0)}$$

so that

$$y^{(k)} = g(y^{(0)}, \Delta z_1, \Delta z_2) \quad (2)$$

Substitution of eqt. 2 into eqt. 1 allows the NN mapping function to be expressed as a function of  $z$  only, i.e.,

$$f(z_1^{(k)}, z_2^{(k)}, z_3^{(0)}, \dots, z_{M_z}^{(0)}, g(y^{(0)}, \Delta z_1, \Delta z_2)) - R_0 = 0$$

The basic steps of the algorithm are:

1. Let  $k=1$  and  $z_1^{(k)} = z_{1,\min}$
2. Solve for  $z_2^{(k)}$  in
 
$$f(z_1^{(k)}, z_2^{(k)}, z_3^{(0)}, \dots, z_{M_z}^{(0)}, g(y^{(0)}, \Delta z_1, \Delta z_2)) - R_0 = 0 \quad (3)$$
3. Move right:  $z_1^{(k+1)} = z_1^{(k)} + \text{stepsize}$
4. Test for stopping: if  $z_1^{(k+1)} > z_{1,\max}$  stop, else go to 5.
5. Increment  $k$
6. Update  $y$  due to the change in  $z_1$  in step 3 according to eqt. 2, with  $\Delta z_2 = 0$ .
7. Return to step 2.

Use of dependent critical parameters depends on whether they can provide information more compactly than if independent parameters are used alone and on whether simple update expressions  $g_y$  can be developed. Update expressions  $g_y(y^{(0)}, \Delta z_1, \Delta z_2)$  that include approximations are acceptable because their effect on the accuracy of a boundary point is negligible when the boundary is close to the current operating point due to  $\Delta z_1$  and  $\Delta z_2$  being small. Approximate update functions may have significant influence on accuracy for boundary points "far away" from the current operating point. However, this part of the boundary would not be of great interest to the operator unless the system was moving towards it, a situation that could be handled by frequent reinitialization.

## 6.0 EXAMPLE

The procedures described in the previous sections were applied to a security problem within the PG&E system in a subarea of California<sup>5</sup> [27]. Figure 3 illustrates the affected region in much simplified form. The three generators, Gen A (GA), Gen B (GB), and Gen C (GC), represent the bulk of the generation in the subarea. Subarea MW load (L), distributed among several buses, is represented in Figure 3 at a single bus. The tie lines represent interconnections between the subarea and the remaining portion of PG&E's system. Flows on all tie lines are into the subarea for most conditions of concern. We apply the procedure to

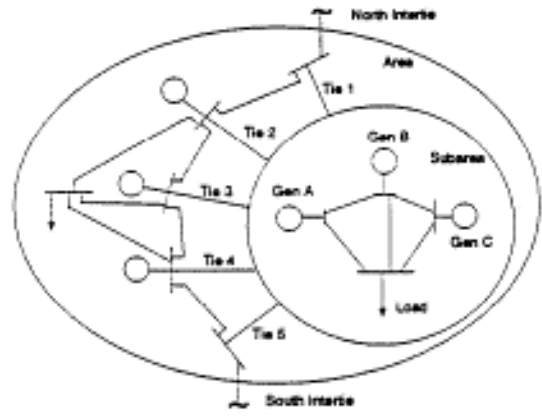


Figure 3: Simplified System One Line Diagram for the Example

a thermal overload security problem because the analysis is relatively simple and serves well to illustrate the concepts involved. The procedure is general, however, in that it also applies well to other security problems. The security problem, which occurs as a result of tie line 3 outage, is thermal overload on tie line 5; thus the postcontingency performance measure is the thermal rating of tie line 5.

### Critical Parameter Selection

A nomogram for this problem could be illustrated very simply using the precontingency flows on tie lines 3 ( $T_3$ ) and 5 ( $T_5$ ) as nomogram axes, since these are the out-aged and overloaded circuits, respectively, and the resulting boundary would provide a very effective predictive capability, i.e., the critical parameter set ( $T_3, T_5$ ) would satisfy the sufficiency criterion<sup>6</sup>. However, it would not provide the operator with corrective action guidance because neither critical parameter satisfies the controllability criterion. Therefore we search for a critical parameter set which contains information (with respect to  $R$ ) equivalent to ( $T_3, T_5$ ) to satisfy sufficiency; the set must also contain a parameter to satisfy controllability. Noting that

$$T_5 = GA + GB + GC - L - \sum_{i=1}^4 T_i \quad (4)$$

we see that choice of the critical parameter set as ( $GA, GB, GC, L, T_1, T_2, T_3, T_4$ ) is equivalent to ( $T_3, T_5$ ) since  $T_5$  may be obtained from these parameters<sup>7</sup>. In addition, set cardinality is minimal since if any one parameter is omitted from the set, sufficiency is no longer satisfied. Of the 8 critical parameters, 4 are independent.

### Data Generation Using ASAS

The ASAS methodology was applied to vary the four independent critical parameters,  $z_1, z_2, z_3,$  and  $z_4$  in the operating ranges  $z_{i,\min} \leq z_i \leq z_{i,\max}$  in  $n_i$  intervals according to the following table:

<sup>5</sup> This subarea is defined as such by geography and not by control, i.e., its "tie lines" to the remainder of PG&E's system are not controlled.

<sup>6</sup> Bus voltage magnitudes could also be included in the set, but this would increase set cardinality without substantially increasing accuracy.

<sup>7</sup> We cannot choose both  $T_3$  and  $T_5$  to be in this set because this would violate the information contribution requirement in that all other parameters add no new information, with respect to  $R$ .

	$z_i$	Range (MW)		$n_i$
		$z_{i,min}$	$z_{i,max}$	
Load	$z_1$	2100	3000	8
Gen A	$z_2$	0	1212	NA *
Gen B	$z_3$	0	400	8
Gen C	$z_4$	0	155	7

\*ASAS uses  $z_2$  to search for the boundary for each state

There are  $7 \times 8 \times 8 = 448$  interval combinations, so that the total number of simulations performed, with  $K_{avg} = 4$  simulations per state to find the boundary, was 1792. These simulations required about 1.5 days of CPU time on a SUN Sparstation LX.

In order to ensure realistic variability on the tie lines 1-4, the dispatch at 3 load following units in PG&E's system external to the subarea and the south intertie flow were chosen at random for each state, with generation at the north intertie, modeled as the swing bus, taking the slack so as to satisfy power balance.

#### NN Training and Testing

The resulting database was used to train a NN using *Predict's* cascade learning algorithm. The resulting NN contained a single hidden layer with six neurons interconnecting the eight inputs to a single output. The NN accuracy was tested using the results of 50 power flow simulations. Accuracy, in terms of average absolute error, was 2.5% of the performance measure threshold  $R_0$ , i.e., the tie line 5 current rating. It is of interest that this implies security *classification* would only be susceptible to error if the operating point was within the  $\pm 2.5\%$  error band around the boundary, a conclusion which we have verified by experiment.

#### Boundary Visualization

The boundary is illustrated using Gen A as one axis, because it represents a large plant with high ramp rates and is therefore easily controllable. Illustration of other security problems in the area [27], not addressed here, make it attractive to choose Subarea MW load as the other axis. We define the dependent critical parameters as

$$y_1: \text{tie line 1 MW flow} \quad y_3: \text{tie line 3 MW flow}$$

$$y_2: \text{tie line 2 MW flow} \quad y_4: \text{tie line 4 MW flow}$$

To apply the visualization algorithm, it is necessary to develop the update functions; these functions yield tie line flows  $y_i$ ,  $i = 1, 4$  as a function of changes in Gen A MW output  $z_2$  or in subarea load  $z_1$ . Development of these functions requires the following definitions:

- $A_m$  is the percentage of the power imbalance caused by  $Dz_1$  or  $Dz_2$  that is redispatched to external generator  $m$  and may be obtained from economic dispatch base points and participation factors [28, pp. 44-46]. External dispatch variation according to these factors constitutes an "allocation rule." Other allocation rules may be used if desired.
- $K_{i,m1}$  is the increase in tie line  $i$  MW flow when external generator  $m$  compensates for a 1 MW increase in  $z_1$ .
- $K_{i,m2}$  is the increase in tie line  $i$  MW flow when external generator  $m$  compensates for a 1 MW decrease in  $z_2$ .

Op. Pt. No.	Independent Parameters (MW)				Dependent Parameters (MW)			
	Load $z_1$	GenA $z_2$	GenB $z_3$	GenC $z_4$	Tie1 $y_1$	Tie2 $y_2$	Tie3 $y_3$	Tie4 $y_4$
11	2600	700	200	155	142	214	70	676
12	2600	500	200	155	193	253	92	756
13	2200	600	0	155	109	182	558	648
14	2200	200	0	155	214	261	103	808
15	2050	600	0	0	109	182	59	653
16	2050	200	0	0	214	261	104	813
17	2600	700	200	0	173	250	80	743
18	2600	500	200	155	144	222	51	869

The factors  $K_{i,mj}$  are obtained assuming the usual DC load flow approximations. These factors are employed to compute the new flow on circuit  $i$  due to a change in MW injection at bus  $m$  countered by an equal and opposite change in MW injection at bus  $j$ , as follows:

$$y_i^{(k)} = g_{yi} (y_i^{(0)}, \Delta z_1, \Delta z_2)$$

$$= y_i^{(0)} + \sum_{m=1}^{M_g} [A_m K_{i,m1} \Delta z_1 - A_m K_{i,m2} \Delta z_2]$$

$$= y_i^{(0)} + \sum_{m=1}^{M_g} [b_{i,m1} \Delta z_1 - b_{i,m2} \Delta z_2]$$

where  $b_{i,mj} = A_m K_{i,mj}$ ,  $m = 1, \dots, M_g$ , and  $M_g$  the number of load-following external generators. We have used  $M_g = 3$  large fossil-fired units in PG&E's area, but this number can be increased as desired.

The visualization algorithm was tested by using it to generate boundaries initialized by the operating points given in Table 1, some of which are secure and some of which are insecure. Power flow simulations of the contingency were conducted for operating conditions corresponding to various test points on the resulting boundary to determine the true value of the performance measure  $R$ . External dispatch differed between initial points I1, I2, and I7, between I3 and I4, and between I5 and I6 only for the three designated load following generators, with allocation defined by the  $A_m$  factors. However, the external dispatch between initial points I2 and I8 differed via a 1000 MW shift in injection at the north intertie to the south intertie.

These boundaries (numbered lines,  $B_i$ ), the initial points (numbered small circles,  $I_i$ ), and the associated boundary error at each boundary test point (the X's) are illustrated in Figures 4 for initial points I1-I7. The secure region is above each boundary. Important observations are

- Comparison of boundaries B1 to B2, B3 to B4, and B5 to B6 indicate that a small difference exists between each pair; this difference in each case is due to the approximations inherent in the functions  $g_{yi}$ . Otherwise, these comparisons indicate that the boundary is independent of the initial values of  $z_1$  and  $z_2$  when variation in initial operating point is compensated by an external redispatch which adheres to the allocation rule.
- Comparison of boundary B1 to B7 illustrates how the boundary changes when an independent critical parameter ( $z_4$ ) varies.

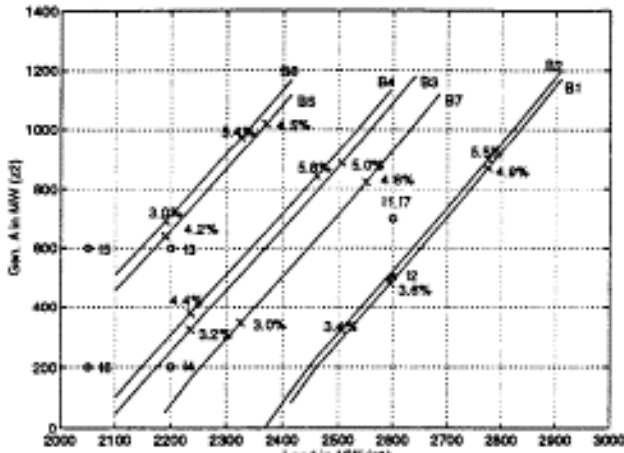


Figure 4: Sample Illustration of Boundaries for Initial Points I-7 (Secure region is above each boundary)

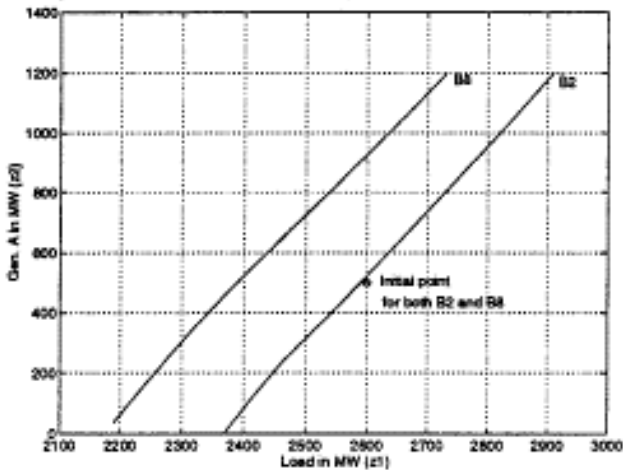


Figure 5: Comparison of Boundaries for Initial Points 2 and 8 (Secure region is above each boundary)

- Error is bracketed between 2.5% and 5.5% (of threshold value,  $R_0$ , in amperes) for all points on the boundaries. The additional error above the average error found in neural network testing (2.5%) is attributable to approximate nature of the update functions.

In addition, Figure 5 compares boundary B2 to boundary B8; this comparison illustrates the importance of the dependent parameters  $y$  since the independent parameters  $z$  are the same for these two boundaries (the tie line information is different because of the 1000 MW north to south shift in external dispatch). Without the dependent parameters  $y$ , the boundaries for initial points I2 and I8 would be identical; use of boundary B2 under the 1000 MW shift scenario would lead to a *hidden* insecure operating condition.

## 7.0 CONCLUSIONS

This work was motivated by the need to more fully utilize existing facilities by operating closer to security boundaries. We have presented an approach to developing security boundaries and providing real time boundary visualization to transmission system operators. The approach extends traditional operational planning techniques by relying on automated security assessment for producing large training databases coupled with the use of neural networks to model the relationships characterized by the data. In addition, it represents a significant improvement over traditional techniques by reducing the labor-intensive study time required by the engineers and analysts performing these studies. It also increases the accuracy of the boundary

characterized using neural network-based nonlinear interpolation in high dimensions. The approach has been illustrated using a thermal overload problem; however, we believe the proposed methodology generalizes very well to other types of security problems. Ongoing efforts include application of the approach to a voltage instability problem.

## REFERENCES

- [1] P. Shanahan and S. Naumann, "Evaluation of Simultaneous Transfer Capabilities," *Proc of the 1995 American Power Conference*, April, 1995, Chicago, III. pp. 1463-1468.
- [2] J. Kanetkar and S. Ranade, "Compact Representation of Power System Security - A Review," *Proc. of the North American Power Symposium*, 1992.
- [3] R. Farmer, "Present Day Power System Stability Analysis Methods in the Western United States," *Proc. of the International Symposium on Power System Stability*, Ames, Iowa, May 13-15, 1985, pp. 39-44.
- [4] L. Wehenkel, T. Van Cutsem, and M. Pavella, "An Artificial Intelligence Framework for On-line Transient Stability Assessment of Power Systems," *IEEE Transactions on Power Systems*, Vol. 4., No. 2, May 1989.
- [5] L. Wehenkel, M. Pavella, E. Euxibie, B. Heilbronn, "Decision Tree Based Transient Stability Method: A Case Study," *IEEE Transactions on Power Systems* Vol. 9., No. 1, Feb. 1994, pp. 459-469.
- [6] L. Wehenkel and M. Pavella, "Decision tree approach to power systems security assessment," *Electrical Power and Energy Systems*, Vol. 15, No. 1, Feb., 1993, pp. 13-36.
- [7] T. Van Cutsem, L. Wehenkel, M. Pavella, B. Heilbronn, and M. Goubin, "Decision tree approaches to voltage security assessment," *IEE Proceedings-C*, Vol. 140, No.3, May 1993, pp. 189-198.
- [8] B. Jayasurya and S. Venkata, "A Knowledge-Based Approach for Power System Dynamic Security Assessment," *Proceedings of the 1990 Association for Computing Machinery Conference*.
- [9] S. Rovnyak, S. Kretsinger, J. Thorp, and D. Brown, "Decision Trees for Real-Time Transient Stability Prediction," *IEEE Transactions on Power Systems*, Vol. 9., No. 3, Aug. 1994, pp. 1417-1426.
- [10] N. Hatziaargyriou, C. Contaxis, and N. Sideris, "A Decision Tree Method for On-line Steady-State Security Assessment," *IEEE Trans. Pwr. Sys.*, Vol. 9, No. 2, May 1994, pp. 1052-1061.
- [11] N. Hatziaargyriou, S. Papathanassiou, and M. Papadopoulos, "Decision Trees for Fast Security Assessment of Autonomous Power Systems with a Large Penetration from Renewables," presented at the 1994 PES Summer Meeting, San Francisco, paper 94 SM 368-1 EC.
- [12] C. Yang and Y. Hsu, "Estimation of Line Flows and Bus Voltages Using Decision Trees," *IEEE Trans. Pwr. Sys.*, Vol. 9, No. 3, August 1994, pp. 1569-1574.
- [13] D. Sobajic and Y. Pao, "Artificial Neural-Net Based Dynamic Security Assessment for Electric Power Systems," *IEEE Trans. Pwr. Sys.*, Vol. 4, Feb., 1989, pp. 220-228.
- [14] Q. Zhou, J. Davidson, and A. Fouad, "Application of Artificial Neural Networks in Power System Security and Vulnerability Assessment," *IEEE Trans. Pwr. Sys.*, Vol. 9, No. 1, Feb., 1994, pp. 525-532.
- [15] M. Aggoune, M. El-Sharkawi, D. Park, M. Damborg, and R. Marks, "Preliminary Results on Using Artificial Neural Networks for Security Assessment," *IEEE Trans. Pwr. Sys.*, Vol. 6, No. 2, May 1991.

- [16] M. El-Sharkawi, R. Marks, and S. Weerasooriya, "Neural Networks and Their Application to Power Engineering," *Control and Dynamic Systems 41*, Academic Press, 1991.
- [17] V. Miranda, J. Fidalgo, J. Pecas-Lopes, and L. Airneida, "Real Time Preventive Actions for Transient Stability Enhancement with A Hybrid Neural Network - Optimization Approach," *IEEE Trans. Pwr. Sys.*, Vol. 10, No. 2, May, 1995, pp. 1029-1035.
- [18] R. Marceau, R. Mailhot, and F. Galiana, "A Generalized Shell for Dynamic Security Analysis in Operations Planning," *IEEE Trans. Pwr. Sys.*, Vol. 8, No. 3, Aug., 1993, pp. 1098-1832.
- [19] M. Huneault, C. Rosu, R. Manoliu, and F. Galiana, "A Study of Knowledge Engineering Tools in Power Engineering Applications," *IEEE Trans. Pwr. Sys.*, Vol. 9, No. 4, Nov., 1994, pp. 1825-1832.
- [20] L. Wehenkel and V. Akella, "A hybrid decision tree-neural network approach for power system dynamic security assessment," *Proc. of the 4th Int. Symp. on Expert Systems application to Power Systems*, Melbourne, Australia, pp. 285-291, Jan. 1993.
- [21] L. Wehenkel, T. Van Cutsem, M. Pavella, Y. Jacquemart, B. Heilbronn, and P. Pruvot, "Machine learning, neural networks, and statistical patterns recognition for voltage security: a comparative study," *Engineering Intelligent Systems*, Vol. 2, No. 4, December 1994.
- [22] N.D. Hatziaargyriou, S. Papathanassiou, J. Pecas-Lopes, J. Fidalgo, and V. Van Acker, "Fast Dynamic Security Assessment of Autonomous Power Systems with a Large Penetration from Wind-The Lemnos Study Case," European Wind Energy Association Conference and Exhibition (EWEC), Thessaloniki, October 1994.
- [23] W. Press, B. Flannery, S. Teukolsky, and W. Vetterling, "Numerical Recipes: The Art of Scientific Computing," Cambridge University Press, 1986, New York.
- [24] *Neural Works Predict Manual*, NeuralWare, Inc., 1995, Pittsburgh, PA.
- [25] S. Fahlmann and C. Lebiere, "The Cascade-Correlation Learning Architecture," *Advances in Neural Information Processing Systems 2*, Morgan Kaufmann, February 1990.
- [26] J. Ehrlich and R. Eymard, "An Efficient Implementation of Conjugate Gradient Method for Multilayer Neural Networks," *Proc. of Neural Times*, EC2, 1993.
- [27] Z. Alaywan, T. Petrich, T. Reece, and J. McIntosh, "Voltage Collapse: Operating Philosophy at Pacific Gas & Electric," *Proc. of the Bulk Power System Voltage Phenomena III Seminar on Voltage Stability, Security, and Control*, Aug. 22-26, 1994, Davos, Switzerland, pp. 187-196.
- [28] A. Wood and B. Wollenberg, "Power Generation Operation and Control," John Wiley and Sons, 1984.

## ACKNOWLEDGEMENTS

The research reported in this paper was supported by Pacific Gas and Electric Company. We express appreciation to Eddie Dedashti of PG&E's R&D Department for supporting this work, to Christine Vangelatos, Robert Sparks, and Ziad Alaywan of PG&E's System Operation Department for data provision and consultation regarding its use, to James F. Luini, private consultant and formerly of PG&E's Transmission Planning Department, for helpful discussions during project inception, and to Louis Wehenkel of the Department of Electrical Engineering, University of Liege, Belgium, and to J. Pecas-

Lopes of University of Porto, Portugal, both of whom provided considerable insight regarding data generation and NN usage during separate visits to Iowa State University.

## BIOGRAPHIES

**James D. McCalley** is Assistant Professor of Electrical and Computer Engineering Department at Iowa State University, where he has been employed since 1992. He worked for Pacific Gas and Electric Company from 1986 to 1990. Dr. McCalley received the B.S. (1982), M.S. (1986), and Ph.D. (1992) degrees in Electrical Engineering from Georgia Tech. He is a registered professional engineer in California and a member of the IEEE.

**Alex D. Papalexopoulos** received the Electrical and Mechanical Engineering Diploma from the National Technical University of Athens, Greece in 1980 and the M.S. and Ph.D. degrees in Electrical Engineering from the Georgia Institute of Technology, Atlanta, Georgia in 1982 and 1985 respectively. Upon graduation Alex joined PG&E where he spent several years working on the development of advanced applications for PG&E's new Energy Management System. Alex is currently responsible for the development and implementation of methodologies, software, databases and information systems for operations, operations planning, transmission planning and transmission and power contracts. Alex is also responsible for the development of models and software for supporting PG&E's efforts in the regulatory arena and in electric industry restructuring. His primary research interests include applications of large-scale theory to the real-time control of power systems, dynamic simulation of power systems and electromagnetic transient analysis. Alex is a senior member of IEEE and a member of Sigma Xi and the Technical Chamber of Greece.

**Roger T. Treinen** received the BSEE (1989), MSEE (1992) and Ph.D. (1993) degrees from Iowa State University. He is now a consultant with Pacific Gas and Electric Company in San Francisco, California, with interests in power system dynamics and voltage security.

Shimo Wang has been a postdoctoral researcher at Iowa State University since 1994. Dr. Wang graduated (1977) from Department of Electrical Engineering, Tsinghua University, Beijing, China. He received the M.S. (1982) and Ph.D. (1986) degrees in electrical engineering from Xi'an Jiaotong University, Xi'an, China. He was employed by Northwest Power Design Institute of Energy Industry Ministry as a consulting engineer and by Xi'an Jiaotong University as an associate professor. Dr. Wang's areas of interest are power system security and reliability, dynamics and control. EMS applications, and artificial intelligence.

**Qianglin Zhao** is a graduate student in the Computer Science Department at Iowa State University. He was an assistant teacher of Automation Department of Beijing Institute of Chemical Technology from 1990 to 1992. He received the B.S.(1987) degree in Chemical Engineering from Beijing Institute of Technology and the M.S.(1990) degree in System Engineering from Beijing Institute of Chemical Technology.

**Guozhong Zhou** received the Bachelor's degree (1985) and Master's degree (1988) in electrical engineering from Tianjin University, PRC. From 1988 to 1992, Mr. Zhou was employed as a teacher and researcher at Tianjin, and from 1992 to 1994, as an engineer for Northeast China Electric Power Administration, and from 1994 to 1995 as a visiting researcher at the University of Porto in Portugal. Mr. Zhou is currently working towards his doctoral degree at Iowa State University.