Border Identification For Power System Security Assessment Using Neural Network Inversion: An Overview

I. N. Kassabalidis, M.A. El-Sharkawi, R.J.Marks II
Dept. of Electrical Eng., Box 352500 University of Washington Seattle, WA 98195 USA
[ikas, elsharkawi, marks]@ee.washington.edu

Abstract – **Knowledge of the dynamic security border can provide an operator with valuable information on how to safely steer the power system away from vulnerable operating regions. The large set of non-linear differential equations that describe modern large-scale power systems makes it difficult to determine the security border either analytically or numerically in real time. As an alternative, neural networks trained off line on emulator data can provide a commensurate representation of the system transfer function, while significantly decreasing evaluation time. Using neural network inversion, sets of input points corresponding to a fixed output can be evaluated quickly. Different inversion procedures and their properties are reviewed here. We also review various metrics used for determining whether sufficient coverage of the border is achieved. Finally, we illustrate the use of border identification for preventive control when feature selection is initially performed to reduce the dimensionality of the input space.**

I. INTRODUCTION

Border identification is a valuable visualization and computational tool for dynamic power system assessment. Knowledge of the security border allows the system operator to steer the system away from vulnerable states. However, for large-scale complex power systems, the security boundary cannot be computed analytically. Instead, the border has to be computed in an approximate fashion by identifying points on the border and interpolating between them [1,2].

This inversion procedure can be computationally intensive if the inversion is performed using complex simulation software. However, the use of neural networks (NNs) [1-8] can greatly increase the convergence speed of the inversion algorithms. The inversion of the NN can be performed by a plethora of techniques (see e.g. [9-16]).

In some cases, the points on the border need to be distributed uniformly and/or close to an operating state. Evolutionary algorithms are ideally suited for such constrained cases of NN inversion. Reed and Marks [1] propose such an evolutionary algorithm, where the constraint is maximal spread of the points on a border. Kassabalidis et al. [2] propose an enhanced *particle swarm optimization* (PSO) [17] algorithm. In this algorithm, the effects of imposing varying constraints on the distribution of the points are investigated. Jensen *et al.* [3] propose another constrained inversion approach based on the nearest inversion algorithm by Reed and Marks [14] and the alternating projections algorithm [18]. Jensen's algorithm seeks to locate the point on the border that is closest to the operating state and satisfies the power flow constraints.

In many cases the dimensionality of the input space is very high, thereby contributing to the computational intensity of the problem. One remedy is feature selection [2, 19 and 20]. Feature selection uses only the most important features of the input and the contour is identified in the reduced-dimensionality selected-feature space.

II. BORDER IDENTIFICATION FOR POWER SYSTEM PREVENTIVE CONTROL

In security or vulnerability assessment of power systems, knowledge of the distance of the current operating point to the security or vulnerability border is required. Such information gives the operator valuable information on how to steer the system away from vulnerable states and into more secure operating regions.

As an example, consider the case when a neural network assesses the security index of the power system. A neural network output value exceeding 0.5 indicates a secure state while a reading below 0.5 denotes vulnerability. The boundary separating secure from vulnerable states, as illustrated in Figure 1, is the boundary specified by the locus of points for which the security index is 0.5. All the coordinates on this border, when supplied as the system input, produce outputs of 0.5. Suppose, for a given input operating point $\vec{x}_0$, the security index is 0.6. We desire to know the *security margin* for this operating point defined as the distance of the current operating point to the border. This is illustrated in Fig.1.

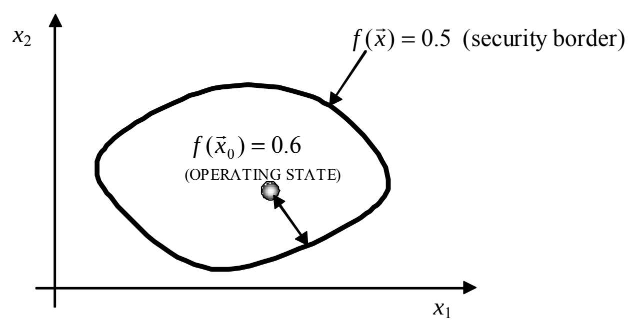

Fig.1. Distance between of the power system operating state and the security border

The border in Fig.1 is a closed loop. Such is the case when the function to be inverted is continuous everywhere. When the function is not continuous, the contour need not be closed. An example of this is shown in Fig.2. Here the distance of the operating state from the security border may not be a good security margin indicator. If the border is a closed loop, the distance of the operating state from the security border is always smaller than the distance of the operating point from a vulnerable region. However, if the border is not a closed loop, as shown in Fig.2, the operating state can be much closer to a vulnerable region than to the security border. Thus, the distance from the border is no longer a good security margin indicator.

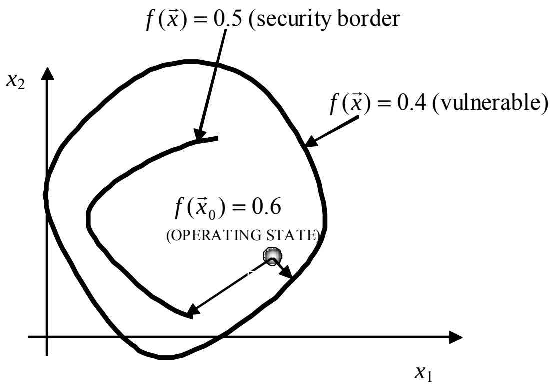

Fig.2. Example where contour-based preventive control provides false security indicators.

Consider training a layered perceptron with data generated from a computationally intensive emulator. Ideally, the neural network will have a nearly identical input-output relationship as the emulator. The use of smooth sigmoidal nonlinearities, however, transforms discontinuities of the emulations into steep slopes in the neural network. The system transfer function is smoothed so that all discontinuities are eliminated [16]. All borders are therefore closed loops. If the result of this smoothing is still a viable approximation, where very little accuracy is lost, border identification can be used for preventive control.

III. Border Identification As An Inversion Problem

Power systems security assessment can be modeled by a many-to-one mapping $y = f(\vec{x})$, where y is a security index such as the *critical clearing time,* the *energy function*, the *second kick or load shedding* and $\vec{x}$ is the input vector. Inversion, in such cases, is typically a surface or contour

$$S = \{\vec{x} : f(\vec{x}) = c\}, \tag{1}$$

where S is the set of surface points, $\vec{x}$ is the input space coordinates, and c is the value of the contour.

When f is simple, the equation describing the contour can be derived relatively easily. However, when f is described by a complex system of non-linear differential equations, a number of optimization methods must be used to locate points on the border. Assuming enough points exist, the border can be reconstructed from these points by linear interpolation.

When the system transfer function is modeled by complex and computationally intensive simulation software, convergence can be slow. As a remedy, NNs can be used to greatly decrease this computational overhead. The inversion of the NN can be performed via a plethora of methods. The concept of iterative NN inversion was first discussed by Williams [9] and again later by Linden and Kindermann [10]. Hwang *et al.* [11] and Hoskins *et al.* [12] discuss applications of the iterative inversion algorithm. Lu *et al.* [15] discuss an interesting application of linear and non-linear programming for neural network inversion.

IV. Constrained Border Identification

The methods discussed in Section III can be applied to finding points anywhere on a border, without any distribution constraints. However, without such constraints, points tend to gather in areas where the border is more easily identified while populating other areas very sparsely. An example of such a two-dimensional surface is shown in Fig. 3, where most of the border points clamp in a portion of the contour. This surface, shown in Fig.3, 4 and 5, is an example plot of $|f(\vec{x}) - c|^2$. The desired border of $f(\vec{x}) = c$ thus coincides with the surface's minimum.

The goal of border tracking is to find points on the border so that areas of interest are covered in a uniform manner with sufficient density to allow accurate border reconstruction via interpolation. The area of interest can be either the entire contour or a portion of the border closest to the current operating state. In order to achieve uniform distribution of the points, constraints must be imposed on their distribution.

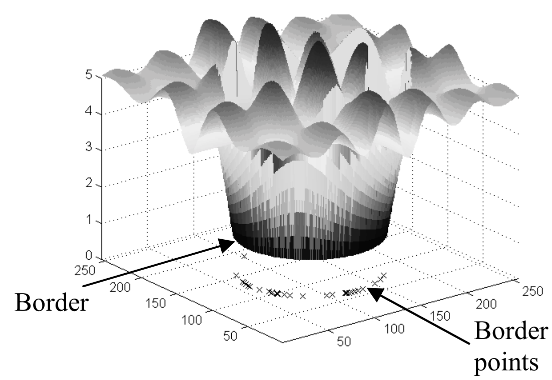

Fig.3. Placing points on the contour without constraints on their distribution. In this case, points tend to gather in some areas, while leaving gaps in others.

Reed and Marks [1] propose an evolutionary multi-agent search where the goal is to distribute points evenly on the entire border. This is performed by penalizing the proximity to any other contour point by adding a penalty factor to the fitness function. Each agent in the algorithm corresponds to a different point on the contour. Therefore, the algorithm finds points on the entire border in a parallel manner.

Jensen *et. al.* [3] propose an algorithm based on the nearest inversion algorithm by Reed and Marks [14] and alternating projections. The algorithm attempts to find the point on the border which is closest to the operating state and satisfies the power flow equations. The procedure begins by projecting a randomly generated search point onto the security boundary and then performs a constrained gradient descent on the security boundary to locate the point closest to the current operating state. Electric power flow constraints are enforced by iterating with an external power flow simulation program.

Kassabalidis et al. [2] propose a border identification method based on PSO. The algorithm locates points either on the entire border or on the section of the border close to the operating state. The points are located by imposing constraints on the fitness function of PSO. More specifically, proximity to neighboring points is penalized, while proximity to the operating state is rewarded. The algorithm is sequential; hence convergence is fast for the first few points. Increased accuracy is achieved once more points are located.

In Fig.4 and Fig.5, an example of implementing these two constraints using the enhanced PSO method is illustrated [2]. In Fig. 4 we see the results of penalizing proximity to any other point. The points are distributed in an almost uniform manner. In Fig.5 results of penalizing for proximity to any other point and rewarding closeness to the operating state are seen. We observe points only on the segment of the contour close to the operating state. These points are distributed in a relatively uniform manner.

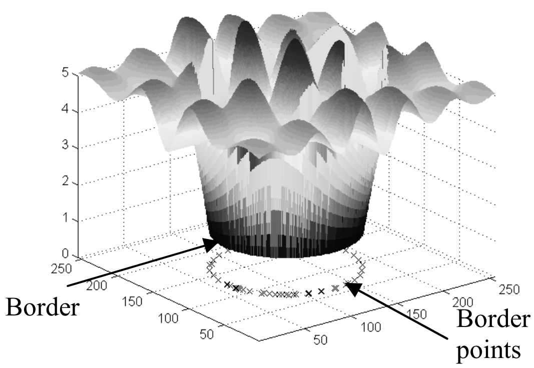

Fig.4. Example of contour identification where points are distributed relatively uniformly throughout the entire contour

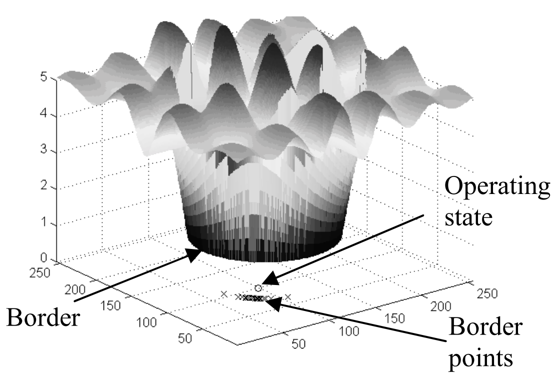

Fig.5. Example of contour identification where points are distributed relatively uniformly on the contour and close to the operating state.

In Table I, we summarize the main features of the above algorithms for finding border points.

TABLE I. FEATURES OF BORDER IDENTIFICATION METHODS

	Evolutionary [1]	Enhanced PSO [2]	Nearest inversion & alternating projections [3]
Placement order	Parallel	Sequential	Sequential
Placement options	Uniformly on entire border	Uniformly on entire border. Uniformly close to operating state	Closest to operating state
Convergence Speed	Attempts to cover the entire border. Suffers from curse of dimensionality.	Can concentrate close to current operating state, thus alleviating curse of dimensionality.	Seeks to find a single point closest to operating state, thus alleviating curse of dimensionality
Other comments	Optimizes for all the points in a parallel manner, but might take longer to converge	Sequential with fast convergence per point. Thus, first few points can be used quickly while accuracy can be enhanced when more points are located	Requires alternating projections. This increases computational times.

V. SUFFICIENT COVERAGE OF THE BORDER

A number of measures can be applied to determine the sufficiency of the generated points on the border. Kassabalidis et al. [2] propose the following measures.

a) Difference between the security index of the points and the desired contour value.

b) Difference between the security index of the midpoints and the desired contour value.

c) Distance of each point to its closest neighbor.

d) Distance of each point to all other points.

The mean of the first metric is used to determine how accurately the points are placed on the contour. However, these points may not suffice to represent the contour with interpolation accuracy.

To determine if enough points are point on the contour, a technique based on the midpoints between neighboring contour points is proposed. The midpoint is defined as the point specified by linear interpolation between a pair of closest neighbors. Each point on the contour has one closest neighbor, thus the total number of midpoints is equal to the number of points. Fig.6 shows the midpoint of two points A and B on the contour, where no other points exist on the contour segment AB. The proximity of the midpoint to the contour is an indicator of whether points A and B are enough to approximate the border segment AB via interpolation. The proximity of the midpoint to the contour can be determined either in the input or the output space. However, as shown in Fig.6, the midpoint-to-contour distance is unknown in the input space since the contour point closest to the midpoint is unknown. Thus, the distance is calculated in the output space.

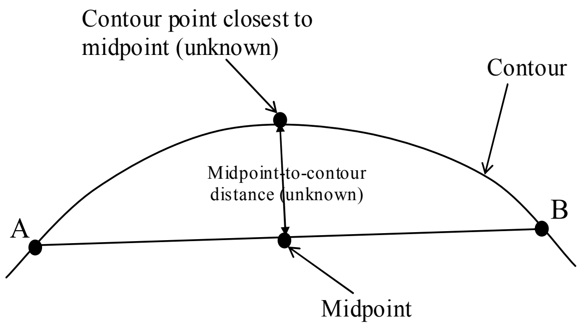

Fig.6. Point B is the closest neighbor of A. Their midpoint is calculated via linear interpolation. In the input space, the distance of the midpoint to the contour is unknown, since the contour segment between A and B is unknown. Thus, we calculate the distance in the output space. The proximity of the midpoint to the contour is an indicator of whether points A and B are sufficiently close to approximate the segment AB via interpolation

Using the mean and variance of the third metric, conclusions can be derived on the distribution of the points. The mean shows us how sparsely the points are distributed. The higher the mean, the larger the size of the covered area. The variance shows us the degree of uniformity of the distribution.

Finally, the fourth metric shows us the size of the covered area. A shown in Fig.7, when the points are clamped in a small region of the contour, the fourth metric will have a smaller mean compared to when the points are spread over a larger area.

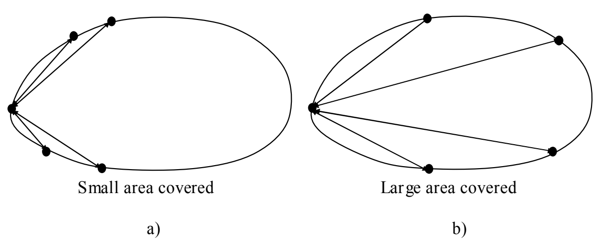

Fig.7. In a) the mean distance of each point to all other points is smaller than in b). This means that the area covered in a) is smaller than in b)

VI. Feature Selection For Preventive control

Feature selection is an approach for dealing with extremely highly-dimensional systems [2, 19, 20]. This technique selects a predetermined number of most important features to reduce the space dimensionality. Once feature selection is performed, the border can be identified on the reduced-dimensionality input space. In power system preventive control, the intention of identifying the border is to calculate the distance, in input space, from the current operating point to the contour. The result is a metric useful for driving the system away from the border. The following steps summarize the procedure proposed.

1) From the coordinates of the current operating state, choose those that correspond to the selected features.

2) Identify the contour close to the operating state in the selected feature space.

3) Use the distances in the feature-space as metrics for preventive control. Since the selected features are the most important of the whole feature set, these distances represent the most important rules to prevent the system from getting close to the border. This procedure is depicted schematically in Fig.8.

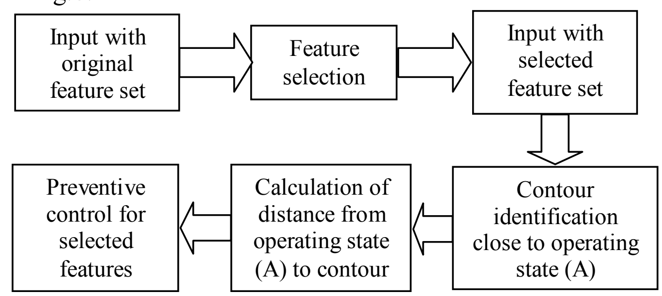

Fig.8. Block diagram of preventive control based on contour identification and feature selection

VII. Conclusions

Border identification is a critical process in power system security assessment. When the border cannot be determined analytically or numerically in real time, a neural network can be trained to generate a mapping essentially equivalent to the analytic or numerical solution. The neural network generates this mapping quickly and can therefore serve as an invaluable tool for the iterative inversion process. For border identification, iterative inversion algorithms must impose constraints on the distribution of the border points, so that coverage is sufficient.

References

[1] Russell D. Reed and Robert J. Marks II, "An Evolutionary Algorithm for Function Inversion and Boundary Marking" Proceedings of the IEEE International Conference on Evolutionary Computation, p. 794-797, November 26-30, 1995.

[2] Ioannis N. Kassabalidis, M. A. El-Sharkawi, R. J. Marks II, Luciano S. Moulin, Alexandre P. Alves da Silva, "Dynamic Security Border Identification Using Enhanced Particle Swarm Optimization", IEEE Transactions on Power Systems, (in press)

[3] C. A. Jensen, R. D. Reed, M. A. El-Sharkawi, and R. J. Marks II, "Location of operating points on the dynamic security border using constrained neural network inversion," in *Proc. Int. Conf. Intelligent Systems Applications to Power Systems, (ISAP'97),*Seoul, Korea, July 1997.

[4] Y. Mansour, A. Y. Chang, J. Tamby, E. Vaahedi, M. A. El-Sharkawi, "Large Scale Dynamic Security Screening and Ranking Using Neural Networks". IEEE Transactions on Power Systems, Vol. 12, No. 2, May 1997.

[5] A. B. R. Kumar, A. Ipakchi, V. Brandwajn, M. El-Sharkawi, G. Cauley. "Neural Networks for Dynamic Security Assessment of Large-Scale Power Systems: Requirements Overview". Proceedings of the First International Forum on Applications of Neural Networks to Power Systems, 1991, pp. 65-71.

[6] R. Marceau, R. Mailhot, F. Galiana, "A Generalized Shell for Dynamic Security Analysis in Operations Planning". IEEE Transactions on Power Systems, Vol. 8, No. 3, Aug. 1993, pp. 1098-1832.

[7] K. Demaree, T. Athay, K. W. Cheung, Y. Mansour, E. Vaahedi, A. Y. Chang, B. R. Corns, B. W. Garrett, "An On-line Dynamic Security Analysis System Implementation". IEEE Transactions on Power Systems, Vol. 9, No. 4, Nov. 1994, pp. 1716-1722.

[8] R. Aresi, B. Delfino, G. B. Denegri, S. Massucco, A. Morini, "A Combined ANN/Simulation Tool for Electric Power System Dynamic Security Assessment". IEEE Power Engineering Society Summer Meeting, 1999, Vol. 2, pp. 1303-1309.

[9] R. J. Williams, "Inverting a connectionist network mapping by back propagation of error," in *Proc. 8th Annu. Conf. Cognitive Science Society.* Hillsdale, NJ: Lawrence Erlbaum, 1986, pp. 859–865.

[10] A. Linden and J. Kindermann, "Inversion of multilayer nets,"in *Proc. Int. Joint Conf. Neural Networks,* vol. II, Washington, DC, 1989, pp. 425–430.

[11] J. N. Hwang, C. H. Chan, and R. J. Marks II, "Frequency selective surface design based on iterative inversion of neural networks," in *Proc. Int. Joint Conf. Neural Networks (IJCNN '90)*, vol. 1, San Diego, CA, 1990, pp. 39–44.

[12] D. A. Hoskins, J. N. Hwang, and J. Vagners, "Iterative inversion of neural networks and its application to adaptive control," *IEEE Trans. Neural Networks,* vol. 3, pp. 292–301, Mar. 1992.

[13] Jensen, C.A.; Reed, R.D.; Marks, R.J., II; El-Sharkawi, M.A.; Jae-Byung Jung; Miyamoto, R.T.; Anderson, G.M.; Eggen, C.J. , "Inversion of feedforward neural networks: Algorithms and applications", *Proceedings of the IEEE* , Volume: 87 Issue: 9 , Sept. 1999 Page(s): 1536 -1549

[14] R. D. Reed, R. J. Marks, II, C. A. Jensen, and M. A. El-Sharkawi, "A neural network inversion procedure," in *Int. Joint Conf. Neural Networks (IJCNN'98),* Anchorage, AK, 1998.

[15] *Bao-Liang Lu; Kita, H.; Nishikawa, Y.* , "Inverting feedforward neural networks using linear and nonlinear programming**",** *Neural Networks, IEEE Transactions on* , Volume: 10 Issue: 6 , Nov. 1999 Page(s): 1271 -1290

[16] R. D. Reed, R. J. Marks, Neural Smithing: Supervised Learning in Feedforward Artificial Neural Networks. MIT Press, 1999.

[17] Particle Swarm optimization, Russell Eberhart, Proceedings IEEE International Conference on Neural Networks (Perth, Australia), IEEE Service Center, Piscataway, NJ, IV: 1942-1948, 1995

[18] R. J. Marks II, "Alternating projections onto convex sets," in *Deconvolution of Images and Spectra,* P. A. Jansson, Ed. San Diego, CA: Academic, 1997, pp. 476–501. *Cognition.* Cambridge, MA: MIT

[19] C. A. Jensen, M. El-Sharkawi and R. J. Marks, Power System Security Assessment Using Neural Networks: Feature Selection Using Fisher Discrimination, IEEE Transaction on Power Systems, pp 757-763, Nov 2001, V0l 16, No 4

[20] Luciano S. Moulin, M. A. El-Sharkawi, R. J. Marks, Alexandre P. Alves da Silva, "Automatic Feature Extraction for Neural Network Based Power Systems Dynamic Security Evaluation". International Conference on Intelligent Systems Applied to Power Systems (ISAP2001),Budapest, Hungary, July 2001.