

# DECISION TREE BASED TRANSIENT STABILITY METHOD A CASE STUDY

L. Wehenkel\*, M. Pavella  
University of Liège

Institut Montefiore; B-4000 Liège, Belgium

E. Euxibie, B. Heilbronn

Electricité de France - Direction des Etudes et Recherches  
1, Av. de Gén. de Gaulle; F-92141 Clamart, France

**Abstract.** The decision tree transient stability method is revisited via a case study carried out on the French EHV power system. In short, the method consists of building off-line decision trees, able to subsequently assess the system transient behavior in terms of **precontingency** parameters (or "attributes") of it, likely to drive the stability phenomena. This case study aims at investigating practical feasibility aspects and features of the trees, at enhancing their reliability to the extent possible, and at generalizing them. Feasibility aspects encompass data base generation, candidate attributes, stability classes; tree features concern in particular complexity in terms of their size and interpretability capabilities, robustness with respect to both their building and use. Reliability is enhanced by defining and exploiting pragmatic quality measures. Generalization concerns multicontingency, instead of single-contingency trees. The results obtained show real promise for the method to meet practical needs of electric power utilities.

**Keywords.** Transient stability assessment; real-time preventive control; artificial intelligence; decision trees.

## 1 INTRODUCTION

From the electric power utilities point of view, transient stability assessment is a need for planning studies as well as operation planning and real time operation. Time domain simulation methods are accurate but time consuming; in addition, their "black-box" type of assessment prevents from identifying and appraising the influence of salient parameters on stability. Many simulations are therefore necessary to determine the stability limiting parameters under various operating conditions, and when a short-term answer is needed, this can only be done in a rather inaccurate way. For short-term operation planning and real time conditions, there is a need for methods providing the operator with reliable indications on the system stability and on the allowed domain for secure operation.

The Decision Tree Transient Stability (DTTS) method was proposed about six years ago with the ultimate goal to meet such types of needs, i.e. to provide straightforward analysis, sensitivity assessment and means to control, under any power system modelling [1]. In its principle, DTTS consists of using Decision Trees (DTs) suitably preconstructed with respect to preassigned contingencies. The building of a tree relies on an inductive inference method which extracts, compresses, and organizes precollected information in a tree fashion, in terms of system parameters ("attributes") automatically selected among a list of candidate ones;

more explicitly: (i) the information is contained in a "learning set", i.e. a set of power system operating states preclassified with respect to the considered contingency; (ii) the candidate attributes are **precontingency** system parameters a priori likely to drive the transient phenomena of concern. The research pursued since the early stage of development contributed to gradually improve and test DTTS on a variety of operating conditions and contingencies, applied on power systems ranging from an academic one-machine-infinite-bus up to a 31-machine system [2, 3]. The method has shown potential to meet practical needs of electric power utilities. These encouraging results have motivated a research project conducted on the EDF system.

The ultimate goal is to assess method's ability to address real life needs, within existing operating strategies, and/or newly created ones, in view of the possibilities opened by this novel approach. This ultimate goal is beyond the scope of this paper. Rather, we aim at reaching the first step towards its achievement. More specifically, we aim at : (i) systematically generating data bases necessary for building and testing the trees; (ii) exploring various types of candidate attributes ranging from very simple and directly controllable up to very complex, less handy ones, and comparing resulting tradeoffs; (iii) similarly, comparing trees for different number of classes and/or different definitions-thresholds of the classes; (iv) enhancing the reliability of the trees for seen and unseen cases to the extent possible; (v) *generalizing them to multicontingency stability assessment instead of the single-contingency trees built so far.* In short, it aims at consolidating, enhancing and extending interesting features that the trees have already shown (such as simplicity, interpretability, reliability, and robustness), in the context of a real world large power system and under real conditions and constraints. The results obtained are quite promising. In particular, it is shown that it is possible to build trees under different conditions to meet different operating needs : for example, simple controllable attributes will yield simple trees, easy to interpret and use for sensitivity analysis and control purposes, while more complex attributes will produce complex but highly reliable trees, particularly suitable for analysis.

The paper is organized as follows. Section 2 gives a brief description of the basic DTTS method. Section 3 focuses on the generation of an "appropriate" data base. This, together with the decision tree method per se, are the key issues upon which the success of the overall DTTS method depends. An "appropriate" data base should be large enough to cover as completely as possible all situations likely to occur, even rarely, yet small enough to be easily tractable. To the best of our knowledge, this is the first time such a systematic in-depth generation is carried out for transient stability assessment of a real power system, via a pattern recognition technique; we therefore deem it necessary to give in an Appendix some relevant working details. Section 4 is the main body of the paper : a sample of constructed trees are reported, their salient features discussed, the "pure" DTTS is extended to multicontingency trees, and techniques are proposed to further improve its performances.

93 WM 235-2 PWRs A paper recommended and approved by the IEEE Power System Engineering Committee of the IEEE Power Engineering Society for presentation at the IEEE/PES 1993 Winter Meeting, Columbus, OH, January 31 - February 5, 1993. Manuscript submitted January 30, 1992; made available for printing December 21, 1992.

\* Research Assistant, F.N.R.S.

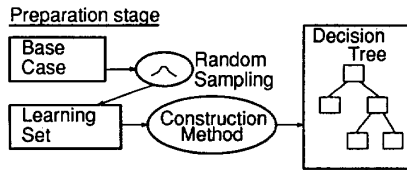


Figure 1. Principle of the DTTS method

## 2 THRUST OF THE METHOD

To make the paper self-contained, we give below a short, intuitive description of the method. For a more comprehensive account, the reader may kindly refer to, e.g. [2, 3].

### 2.1 Principle and basic notation

Decision tree methodology in general is an inductive learning technique able to produce classifiers in form of trees. The building of a tree relies on a learning set ( $LS$ ), i.e. a **sample of preclassified states**, and a **list of candidate attributes**.

In the particular context of DTTS, a **state** is a normal (pre-contingency) **Operating State (OS)**. In a two-class scheme, its classification with respect to a **preassigned contingency**<sup>1</sup> (defined by its type, location and clearance scenario) is obtained by : (i) running a transient stability program, for example a time domain numerical simulation, to assess the Critical Clearing Time (CCT) corresponding to the contingency; (ii) fixing a threshold CCT value (denoted by  $\tau$ , in the sequel), to declare the OS stable or unstable depending upon whether its CCT is larger or smaller than the threshold. This latter is in principle arbitrarily chosen, but may be guided by particular considerations (see in §3.3)<sup>2</sup>.

More generally, a  $c$ -class classification uses  $(c-1)$  threshold CCT values.

In addition to the learning set, the building of a tree requires the preselection of a list of, say  $n$ , **candidate attributes**. These are **precontingency** parameters of the system, presumably carrying sound information about transient stability. Accordingly, each OS will be characterized by the values of its  $n$  candidate attributes. Thus,  $OS_k$  will be characterized by its attribute values  $(v_{1k}, v_{2k}, \dots, v_{nk})$ .

Given a (preclassified)  $LS$  and the list of candidate attributes, the automatic building of a tree conforms to the procedure described below in the simple two-class classification scheme.

• Starting at the **top node** of the tree and with the complete  $LS$ , i.e. the mixture of stable and unstable states, find out a way to split these states into the two most “purified” (stable or unstable) subsets. Their “optimal” splitting is achieved in the two steps :

- (i) for each attribute, say  $a_i$ , find its “optimal” value (optimal in the above sense) by scanning the values it assumes for all learning states; let  $v_{i*}$  be this optimal value;
- (ii) among the candidate attributes, choose the best one,  $a_*$ , along with its optimal value,  $v_{**}$ , to split the node, using the test

$$T : a_* < v_{**} ? \quad (1)$$

In short, step (ii) defines the optimal attribute, step (i) its optimal value.

• The selected test thus applied to the learning set of the node, splits it into two subsets, corresponding to the two successors of the node. These successors are labelled **terminal** or **nonterminal** on the basis of the stop splitting criterion described below. For

<sup>1</sup>In the sequel, the terms “contingency” and “fault” will be used interchangeably.

<sup>2</sup>The above CCT based classification is by no means a restriction of DTTS. It was chosen for the sake of convenience, since CCT is admittedly an objective measure.

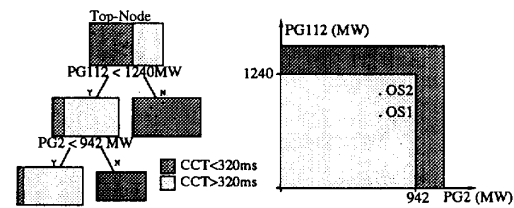


Figure 2. Illustration of a tree and its attribute space [3]

the nonterminal nodes, the node splitting procedure is applied repeatedly to build the corresponding subtrees. Each terminal node is labelled **stable** or **unstable**, depending upon the majority class of its learning states (possibly along with a weighting factor).

A crux of the entire tree construction is the suitable decision upon whether to declare a node terminal, or to continue splitting. It relies on the “stop splitting criterion”, which stops splitting a node as soon as either of the following two conditions is met :

- (i) the local subset of learning states is “sufficiently” class pure; such a terminal node is henceforth referred to as a **leaf**. The degree of required purity is fixed via a parameter which however is assigned a very small value. Consequently, the stop splitting criterion generally relies on the second condition :

- (ii) there is no possibility of enhancing the tree accuracy in a statistically significant way by splitting a node further : this is then considered to be a **deadend** terminal node<sup>3</sup>. This condition can be formulated as a statistical hypothesis test. Quantitatively, it amounts to fixing the risk  $\alpha$  of not detecting situations of statistically non significant class purity improvement : setting  $\alpha = 1$  corresponds to fully growing the tree, setting  $\alpha = 0$  to shrinking it to its top node. The choice of  $\alpha$  has indeed drastic effects on the resulting tree characteristics as will be illustrated in Section 4.

The principle of DTTS is schematically portrayed in Fig. 1.

### 2.2 Possible uses

DTTS provides two possible frameworks of use in practice : the tree itself, and its attribute space representation. More explicitly :

- the tree itself may classify any state in terms of the values of its “test attributes”, i.e. the attributes appearing at the test nodes. For example, in Fig. 2 (drawn in a simplified context), the test attributes are PG112 and PG2. Hence, the two states characterized by

$$OS_1 = [PG112 = 1000MW] \& [PG2 = 650MW]$$

$$OS_2 = [PG112 = 1150MW] \& [PG2 = 650MW]$$

are both stable;

- the tree attribute space is another, in principle equivalent representation, where however the hierarchical classifier (the tree itself) is replaced by a “hyperbox type” classifier [4]. Again,  $OS_1$  and  $OS_2$  are both shown to be stable; but here  $OS_1$  obviously appears to be more stable than  $OS_2$ .

The above description suggests that the two representations have different, complementary possibilities. To the advantage of the tree we mention a synthetic, straightforward description of the phenomena. To the advantage of the attribute space representation, on the other hand, are stability margins, assessment of the sensitivity to the salient system parameters (test attributes), and means to control (“How far” from (in)stability a state is? “How sensitive” it is to the various test attributes? “What if” a state is unstable?). This latter representation relates to “security region” type of applications, for which a large body of literature already exists (e.g. see [5, 6], and the references therein). Admittedly, very interesting outcomes may reasonably be expected from this kind

<sup>3</sup>In fact, this criterion is the  $\chi^2$ -based one developed in [2, 3].

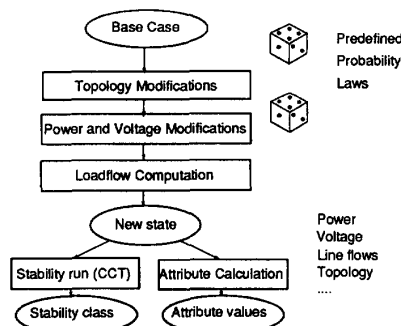


Figure 3. Principle of a data base generation

of approaches. In the sequel, however, we will exclusively focus on the trees themselves, our purpose being to consolidate DTTS to the extent possible before dealing with applications.

### 2.3 Salient tree features

**Complexity.**<sup>4</sup> Parameter  $\alpha$  of the statistical hypothesis test plays a fundamental role on the overall shape and properties of a tree. In general, its choice should be guided by the observation that too large a tree (corresponding to a too large  $\alpha$  value) may yield a higher misclassification rate and hide relevant relationships among more or less random details, whereas too small a tree will not make use of some of the information available in the *LS*.

**Reliability and its evaluation.** The reliability of a tree is its ability to properly classify seen and unseen cases. To assess this or equivalently the misclassification rate, we consider a set of  $M$  test states (independent although similar to the learning states), drop them through the tree, and compute the ratio of misclassified states over  $M$ .<sup>5</sup> This so-called "test sample estimate" is recognized to be a very reliable and unbiased estimate, provided that  $M$  is large enough (e.g.  $M \geq 1000$ ) [7].

In the following we will distinguish three different types of errors of a tree: (i) false alarms, i.e. stable states classified unstable by the tree; (ii) false dismissals, i.e. unstable states erroneously classified stable; (iii) gross errors, i.e. the truly dangerous diagnostics among the preceding ones; these are states classified stable and whose CCT is significantly smaller than the threshold below which a state is considered to be unstable (e.g. 10 percent smaller:  $CCT \leq 0.9\tau$ ).

## 3 DATA BASE GENERATION

The data base provides the *LS* for building trees and the test set, similar but independent of the *LS*, for testing them. Its proper generation is therefore of paramount importance for the method. Figure 3 sketches the principle.

### 3.1 Base case conditions

The considered system is an earlier version of the EDF system formerly used for operation planning studies. It encompasses the complete 400 kV grid of the French system as well as the most important part of the 225 kV network, yielding a 561-bus / 1000-line / 61-machine system. Equivalent representations are used for the surrounding European interconnections (Germany, Switzerland, Italy, Spain and Belgium). The overall generation produced by the 61 (equivalent) machines corresponds to about 60,000 MW of national generation and 50,000 MW of external equivalents.

The case study concerns the stability assessment of an impor-

<sup>4</sup>Intuitively, complexity of a tree refers to the number of its nodes.

<sup>5</sup>In the sequel the terms "test set error rate" or "error probability" are used interchangeably; it is denoted by  $P_e$ .

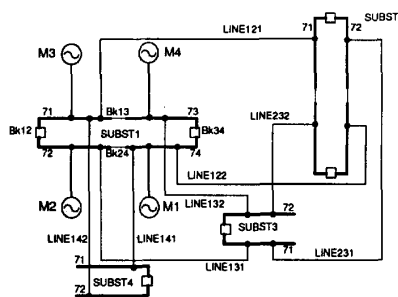


Figure 4. One-line diagram of the study plant substation

tant plant situated in Normandy (North-West part of France). This study plant was selected via a preliminary investigation where 60 different contingencies at the 400 kV level have been screened on 9 different operating states. Figure 4 describes its substation and immediate neighbors at the 400 kV level. The following three severe contingencies have been identified, classified in a decreasing order of criticality.

**"Busbar" fault :** three-phase short-circuit located on the busbar section 71 in substation 1; cleared by opening the lines 121 and 142, tripping machine M3 and opening the breakers 12 and 13 to isolate the faulted section.

**"Double line" fault :** 2 simultaneous three-phase short-circuits nearby sections 71 and 74 of substation 1, on the lines 141 and 142; cleared by opening both lines.

**"Single line" fault :** a three-phase short-circuit on line 131 nearby the busbar section 72; cleared by opening this line.

The CCTs of the above contingencies are computed by a standard step-by-step program where the machines are represented by the classical simplified "constant e.m.f. behind transient reactance" model, and the loads as constant impedances.

The data base was generated from a base case via modifications described below (§3.2). They essentially concern the "study region", but of course all load flow and stability computations are run on the entire system. This study region presumably encompasses all components liable to (significantly) influence the stability of the study plant. It was determined by EDF engineers in charge of stability studies. Interestingly, it was also identified, in an independent way, using the "Combined Electromechanical Distance" approach [8].

The study region is composed of three large power plants (named in the sequel plant 1, 2, or 3 - which is the study plant) along with the surrounding substations, and about 60 lines at the 400 kV and 225 kV levels. The overall installed generation capacity of these plants is about 10,000 MW and the base case load is approximately of 5,600 MW (corresponding to winter peak load) shared among 42 different load buses.

### 3.2 "Controlled" random generation of states

The primary objective was to obtain a sufficiently rich data base, which at the same time contains plausible operating states of the region and covers as much as possible weakened situations. For this purpose, a certain number of independent variables, liable to influence the system stability were defined, concerning both topology and electrical status<sup>6</sup>. For each variable a prior distribution was fixed on the basis of available statistical information about the usual situations, so that all interesting values would be sufficiently well represented in the data base. Moreover, constraints were imposed on values taken by different variables in order to exclude unrealistic situations. 3000 operating states were thus randomly drawn, their stability was assessed and the values of various types

<sup>6</sup>"Electrical status" to distinguish it from the "topological status" or simply the "topology" (see Fig. 3).

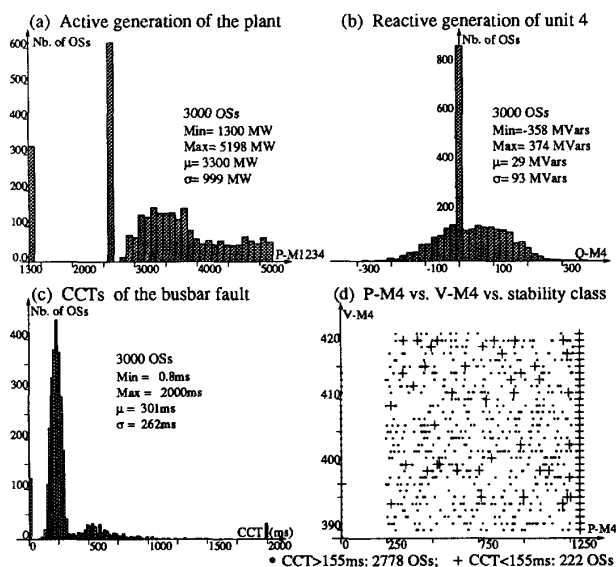


Figure 5. Statistics relative to the study plant

of candidate attributes were computed (see §3.4). The detailed description of the parameters used for the data base generation is given in Appendix A. We merely note here that for each state the following tasks are executed :

1. definition of the load level in the study region, the topology of the 400 kV regional network, local active and reactive generation scheduling;
2. building of the load flow and step-by-step data files;
3. load flow computation and feasibility check;
4. appending of the states attribute values in the corresponding attribute files;
5. computation of the CCTs of the considered disturbances via step-by-step simulations.

The amount of data and computations involved is quite considerable. All in all, they imply more than 25,000 load flow runs and 9,000 CCT computations (3000 for each considered contingency).

Figures 5a, 5b illustrate the diversity of situations generated by drawing the statistical distributions of important variables of the study plant. Figure 5c provides the distribution of the CCTs of the 3000 states for the busbar fault described in §3.1. Figure 5d shows the distribution of these states, classified for the same fault with respect to the threshold  $\tau = 155\text{ms}$  (• if CCT is larger, + if CCT is smaller than 155ms), in terms of voltage and active power of machine M4 of the study plant. Notice the extreme intertwining of cases (•) and (+).

### 3.3 Stability classes

For each contingency, various classifications were considered, for two-, three-, four-class trees and various threshold CCT values.

**Threshold at the “actual clearing time”.** As indicated in Fig. 5c, despite the extreme severity of the contingencies, only a small proportion of the learning states are found to have a CCT lower than the “actual clearing time” (equal to 155ms for a busbar fault and 100ms for line faults). An important question is therefore : how does this imbalance between stable and unstable states influence the DTs?

**Threshold at the median of the CCT distribution.** If the threshold is taken as the median of the CCT distribution (e.g. 235ms for Fig. 5c), the class boundary is situated in a very dense region of the attribute space. Two competing effects are thus ex-

pected : (i) more learning states nearby the class boundary provide richer information on the attribute vs. stability relation; (ii) more test states nearby the class boundary will yield higher error rates. **Multi-class trees.** Four-class trees were built by using three threshold values including the actual clearing time and two larger values. In addition, for the busbar fault two-class trees using a threshold of 350ms, and three-class trees using two threshold values of 155ms and 350ms have also been built.

### 3.4 Candidate attributes

Appendix B describes the 13 lists of candidate attributes of growing complexity (all in all 160 different attributes) which have been proposed to the decision tree method. They characterize the study region by its topology, its electrical status, and/or the combined effect of the two, via more or less complex combinations. These candidate attributes may be classified into one of the three following categories, according to the type of information they convey and the type of applications that the resulting decision trees could handle in practice.

**Controllable** attributes include the regional load level, voltage set-points and active generations of units, as well as the topological variables. The corresponding DTs would yield straightforward analysis and control tools.

**Observable** attributes include in addition dependent variables such as reactive generations, power flows and/or relative phases. Corresponding trees would require auxiliary post-processing tools to allow control applications but their information could still easily be appraised by operators.

**Complex** attributes may take into account any kind of information concerning the fault location and clearance scenario as well as pre-fault operating state information. These may be combined to yield complicate “ad hoc” attributes, which at the expense of a lesser intelligibility may sometimes increase significantly the reliability of the trees.

## 4 APPLICATION RESULTS

The first two paragraphs of this section focus on tree building, according to the standard DTTS outlined in Section 2. The following two paragraphs discuss main features of the obtained trees. Finally, §§4.5, 4.6 propose extensions and improvements of the standard DTTS.

### 4.1 A first broad investigation

Its purpose was to determine appropriate values of parameters of the method, and to provide insight on its capabilities under various conditions. About 40 DTs were thus built for the following range of parameters :

- learning set size :  $N=500$  and  $2000$ ;
- values of  $\alpha$  :  $\alpha = 10^{-1}$  and  $10^{-4}$ ;
- classifications : 11 different classifications were considered, relative to the three faults, and different numbers and values of thresholds defining the classes.

The trees have been tested on the basis of 1000 test states, not considered during the learning stage. Figures 6 and 7 portray two such representative trees consistently using the same notation. Thus, each node of the DT is represented by a box. Above the box appears the node name, labelled T (test), D (deadend) or L (leaf) as appropriate, along with the number of its learning states. The total number of different types of nodes of the tree is indicated at the top-node. The node box itself is subdivided into upper and lower parts. Their relative height is proportional to the relative



Table 1. Distribution of classification errors of a 4-class tree

		True class (thresholds in ms)				
		Nb. of test states (Nb. of all states)				
		<155	155-200	200-250	>250	Total
Class	<155	59 (197)	12 (25)	3 (3)	1 (2)	75 (227)
via	155-200	11 (25)	139 (496)	36 (71)	0 (2)	186 (594)
DT	200-250	0 (0)	24 (53)	286 (867)	41 (88)	351 (1008)
(ms)	>250	0 (0)	1 (1)	30 (70)	357 (1100)	388 (1171)
Total		70 (222)	176 (575)	355 (1011)	399 (1192)	1000 (3000)

**Four-class trees.** The error rate and the complexity are even more important. However, here most of the errors are located in adjacent classes and correspond to a less misleading diagnostic than for two-class trees [3]. This is illustrated in Table 1, where the diagonal entries correspond to correctly classified states, the entries below the diagonal to overly optimistic diagnostics, those above the diagonal to overly pessimistic ones.

## 4.2 Effect of different candidate attributes

The “optimal” parameters determined in the above investigation ( $\alpha = 10^{-4}$ ,  $N = 2000$ ) were used to assess the different types of candidate attributes described in §3.4 and in Appendix B. Trees were thus built for the busbar fault, corresponding to the 13 lists of candidate attributes and to the two 2-class classifications obtained with the threshold values of 155ms and 235ms.

Some interesting characteristics of the resulting trees are summarized in Table 2, for growing attribute complexity : list 1a corresponds to purely controllable attributes, lists 2a-2i to observable ones, and lists 3a-3c to complex ones. The first two columns of the table identify the name of the list and the number of its candidate attributes. The next five columns relate to trees built with the “actual clearing time” threshold (155ms), whereas the following five use the “median” threshold (235ms). For each one of these two blocks, one lists the following columns :

- “DT” : the tree name
- $P_e\%$  : the test set error rate
- $I_Q^{DT}\%$  : the information quantity provided by the tree, evaluated as a percentage of the learning set entropy; it reflects the degree of tree classification capability in a global way [3]
- “Nds” : the tree complexity in terms of its node number
- “Atts” : the number of test attributes selected by the tree.

## 4.3 Discussion

A comprehensive discussion about the rich, multiform information provided by a tree would necessitate much space. We will restrict ourselves to observe again that globally, the trees can indeed provide a clear picture of the intricate transient stability phenomena. At the same time, they assess the stability behavior of an operating state in terms of solely the test attributes relevant to this state. Further, the influence of each test attribute may be appraised by means of its relative position in the tree and by its information quantity or classification capability (not reported in this paper for obvious space limitations).

These and many other pieces of information may be very useful to the system operators; they corroborate and/or complement their own experience obtained via tedious everyday learning of the system behavior, and help them get a refined and confident understanding of the phenomena.

The sheer classification ability of the tree through its hierarchical structure is another fundamental property worth mentioning again; it is nicely highlighted by comparing DT1 of Fig. 7 with the extreme intertwining of states’ stability degree suggested by Fig. 5d, drawn for two quite important numerical test attributes of the tree : P-M4, the active power generated by unit 4 which appears

Table 2. Summary of DTs built for different candidate attributes

		$N = 2000$		$M = 1000$		$\alpha = 10^{-4}$		$H_m = 10^{-2}$	
		$\tau = 155\text{ms}$				$\tau = 235\text{ms}$			
		222 Unst & 2778 St				1493 Unst & 1507 St			
ATTRI-BUTES		DT	Quality	Compl.	DT	Quality	Compl.		
List	Nb.		$P_e\%$   $I_Q^{DT}\%$	Nds   Atts		$P_e\%$   $I_Q^{DT}\%$	Nds   Atts		
1a	38	DT1	1.8   63.7	25   7	DT14	11.3   70.7	49   14		
2a	48	DT2	1.5   74.5	29   9	DT15	11.5   71.4	51   13		
2b	60	DT3	1.7   76.6	23   10	DT16	9.9   71.9	53   15		
2c	72	DT4	cf. DT3	cf. DT3	DT17	10.1   71.4	55   18		
2d	50	DT5	2.3   58.7	21   9	DT18	10.6   70.9	45   13		
2e	63	DT6	2.0   80.2	23   9	DT19	10.0   75.3	55   14		
2f	72	DT7	1.6   76.1	21   10	DT20	cf. DT16	cf. DT16		
2g	92	DT8	cf. DT7	cf. DT7	DT21	11.1   73.5	61   18		
2h	120	DT9	1.7   78.7	23   11	DT22	11.3   75.1	67   20		
2i	123	DT10	1.4   79.9	21   8	DT23	9.7   76.6	55   18		
3a	100	DT11	1.3   78.2	19   9	DT24	8.2   79.2	43   14		
3b	119	DT12	1.9   85.0	9   4	DT25	7.3   88.2	41   15		
3c	103	DT13	1.0   78.6	15   7	DT26	7.1   81.5	51   14		

at three different test nodes and contains about 19% of the tree’s information quantity, and V-M4, which, although used only once, contains 14% of the information quantity of the tree.

Coming to more specific information stemming from the results of this section, we observe the following.

**Stop splitting rule.** The effect of parameter  $\alpha$  on the tree complexity and reliability, observed in previous studies, is fully corroborated : using low  $\alpha$  values allows one to reduce the tree complexity by a factor of 2 to 3, while improving (albeit slightly) reliability.

**DTs built w.r.t. the “actual clearing time”.** Table 2 shows that even the most elementary attributes (list 1a) yield DTs of satisfactory reliability (this is confirmed by a more refined analysis of the classification errors). A very good compromise thus seems to be DT1, if sensitivity analysis and preventive control applications are sought. On the other hand, DT13 seems to be a good choice if only analysis is considered : it is more reliable and the used attributes, although interdependent and fault specific, remain quite easy to appraise in a control center environment. Thus the combined use of both DT1 and DT13 would allow to achieve both reliability and flexibility of use.

**DTs built w.r.t. the median.** For DT14 to DT26, the effect of candidate attributes on tree parameters is more strongly marked : complexity and reliability vary in an important fashion. Likewise DT1 and DT13, DT14 and DT26 appear to be a good choice.

**Linear combinations.** The automatic linear attribute combination allows in general to somewhat improve the tree reliability (lists 1f, 2d and 3c). Their slightly better performances are however obtained at the expense of less straightforward tree interpretability capabilities.

## 4.4 Computing requirements

The main off-line effort for DT building is devoted to the data base generation and amounts to about 4 days CPU time on a 28MIPS SUN/SPARC2 workstation. The total amount of data generated is around 30Mbytes. Note that from the computational burden viewpoint, the heaviest part is by far the CCT computation. On the same workstation, the building of a tree amounts to about 5 minutes (with 2000 learning states and 60 candidate attributes). Its on-line use to classify a state amounts to less than 0.1 ms.

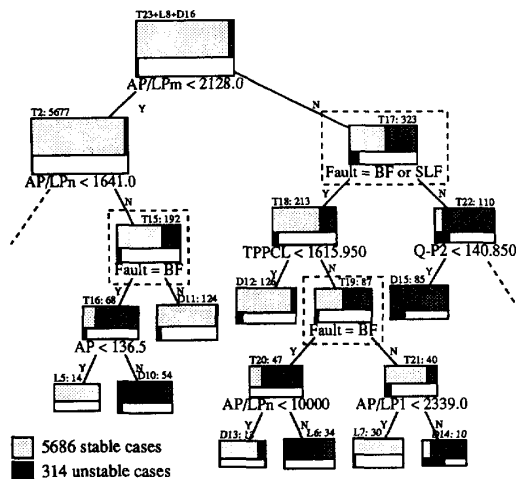


Figure 9. Multicontingency DT : 3 faults  $N = 6000$   $M = 3000$   
 $\alpha = 10^{-4}$   $P_e = 1.6\%$

#### 4.5 Multicontingency assessment

In the “basic” DTTS approach described in Section 2, each tree is designed for a given, single contingency. Accordingly, to assess the stability of an operating state with respect to a set of contingencies, one has to scan the various “single-contingency” trees, each corresponding to one of these contingencies. In particular, the trees reported and discussed so far were all built with respect to the contingency “busbar” defined in §3.1.

In this paragraph we propose to extend the basic DTTS method in order to design “multicontingency trees”, where a single tree handles several (related) contingencies (e.g. faults located in the same substation). Among different explored ways of defining multicontingency trees we succinctly describe the following : each “stability case” is specified by a prefault operating state (characterized by prefault attributes) and a fault, identified by a discrete attribute; it is accordingly classified as (un)stable if its operating state is (un)stable with respect to its fault. The learning and test sets are composed of collections of such “stability cases”; a given operating state yields as many stability cases as there are faults.

Figure 9 sketches a multicontingency tree for the three contingencies defined in §3.1 : (i) the “busbar” fault (denoted “BF”, in the tree), cleared after 155ms; (ii) the “double line” fault (“DLF”); (iii) the “single line” fault (“SLF”), both cleared after 100ms. The three contingencies together with the 3000 operating states yield 9,000 stability cases : a random sample of 6000 are used as the learning set, and the remaining 3000 as the test set. To save space, LH and RH parts of the tree have not been represented in the figure. Note that the nodes where the retained test attribute is “Fault” are encircled by dotted line boxes.

Comparing this tree with the corresponding single-contingency ones, we observe that it has (i) a complexity of 47 nodes vs. 45, the total number of nodes of the three single contingency trees; (ii) an error rate of 1.6% vs. 1.7%, the mean error rate of the single contingency trees; (iii) 14 different test attributes (including the attribute “Fault”) vs. 18, the total number of different test attributes of the single contingency trees.

Thus, without loss of reliability, the multicontingency tree provides a more synthetic view of the stability relationship than several single contingency trees. Moreover, similarities among contingencies are identified and highlighted by the tree (e.g. the operating states corresponding to node D10 are unstable with respect to fault BF; states corresponding to node D11 are stable for the SLF and DLF faults etc.).

Table 3. Quality improvement of DT3 of Table 2

Technique for improving $Q$	$Q_1$		$Q_2$		$Q_3$	
	Nb.	(%)	Nb.	(%)	Nb.	(%)
Basic	9	(0.30)	48	(1.60)	12	(0.40)
(ii)	23	(0.77)	32	(1.06)	9	(0.3)
(ii) + (iii)	47	(1.57)	17	(0.57)	2	(0.07)

Further, inspection of Fig. 9 suggests that, although equivalent to the information provided by a set of single-contingency trees, the information provided by the corresponding multicontingency tree is presented in a more compact and easier to exploit fashion. This can be explained by the fact that similarities of different contingencies are exploited during the tree building so as to simplify the resulting tree. In particular, overlappings of unstable (resp. stable) regions are identified and imbedded in the tree : hence, combinatorial explosion, inherent in multicontingency control on the basis of single contingency trees, is avoided.

Overall, though it is still too early to assess advantages of a multicontingency tree, we observe that it directly provides either of the following types of information :

- for a given fault (among those used to build the tree) is the considered operating state likely to be unstable or not ?
- for a given operating state, are there faults likely to create instability ?
- which conditions characterize the prefault attributes of stable operating states for a given set of possible faults ?

#### 4.6 Reliability improvement

In regard to the different error types introduced in section 2.3, the trees obtained so far via the “pure” DTTS achieve very low error rates, with very few gross errors. Yet, for real life uses, it is desirable to further reduce to the extent possible the dangerous diagnostics, without generating, however, too many “false alarms”. To achieve this goal, the three following “pragmatic quality measures” have been used to account for different types of errors of a tree :  $Q_1$  the number of false alarms;  $Q_2$  the number of false dismissals;  $Q_3$  the number of gross errors, among the preceding, i.e. of dangerous diagnostics; remember, these are states classified stable and whose CCT is significantly smaller than the threshold used to classify the states (see §2.3. E.g.  $CCT \leq 140ms$  for DT1 to DT13).

##### 4.6.1 Reducing the number of dangerous errors

Among several techniques investigated, we mention the most efficient ones : (i) using a CCT threshold slightly (5-10%) larger than the desired one, so as to increase the number of states classified unstable; (ii) biasing the probability of unstable states, by increasing their weight; (iii) using high relative misclassification costs for the unstable states. This amounts to labelling a dead-end as stable, only if a large enough majority of its states are stable. In other words, the “small” deadend nodes, located on the stability boundary, are preferably labelled unstable.

The simulations show that the combined use of either technique (i) or (ii) at the tree building stage and technique (iii) at the tree application stage, yield very satisfactory results. This is illustrated by Table 3, which lists the different types of errors of DT3 of Table 2, and its improved versions. (The percentages are given with respect to the 3000 states used to evaluate the tree qualities.)

##### 4.6.2 Reducing the number of false alarms

As one may see, the previous techniques allow to efficiently reduce the number of dangerous diagnostics of a tree, but at the price of an increased number of false alarms. We therefore propose to use an incremental tree development scheme, in order to compensate for the latter increase. This is illustrated in Fig. 7, where 500 additional learning states have been used to build an incremental subtree for node D1.

The above suggested iterative tree enhancement requires further investigations to become truly effective, e.g. to quantitatively evaluate the amount of additional learning states required and the proportion of tree nodes that need to be expanded. It appears however to be very promising, as shown by preliminary investigations. In particular, a large majority of the false alarms of a tree are generally located at a small number (3 or 4) of its "weak" deadends. Thus only a small part of a tree would need an iterative enhancement, and consequently only a reasonably small number of additional learning states would be required.

## 5 CONCLUSION

The general objective pursued by the case study reported in this paper was to confront the DTTS method with fully realistic, extremely stringent requirements imposed by the large-scale EDF system. Three goals were in particular sought. First, to explore whether method's previously exhibited interesting properties were still valid. This goal has been plenty reached: the obtained decision trees were found to be satisfactorily reliable, simple with respect to the number of tree nodes and selected test attributes, easily interpretable in terms of precontingency system parameters, robust and flexible with respect to various types of candidate attributes and class number and definition. It is interesting to note that the size of this large-scale system has by no means affected these properties.

The second goal was to investigate aspects relating to the method feasibility, and in particular to the proper generation of a data base. Concerning size, it was found that although large, it lies within reasonable, tractable limits. Concerning generation strategies, advice of engineers in charge of the system operation is admittedly extremely helpful. Note, however, that essential expertise is also provided by the trees themselves - and this is another interesting feature of the method. Hence, at the first stage of developments, a cut-and-try procedure appears to be of great help to both power system operation engineers and tree designers. In particular, this procedure contributes to discover and embed peculiar, yet interesting situations and hence to avoid to a large extent dangerous diagnostics.

The third goal was to improve DTTS performances by enhancing its reliability and by broadening its application domain. The pragmatic quality measures and improvement methods have allowed to reach extremely low "dangerous error" rates. Finally, the multicontingency tree approach used for the first time shows to be an interesting generalization.

Admittedly, many fundamental issues are still to be addressed, as, for example, sensitivity and control with respect to the security regions in the attribute space, or adequate strategies for the method's practical application. But, clearly the method opens new avenues to the broad and difficult field of power system security assessment. In this context, an essential factor is the close collaboration between experts on power system operation and tree designers.

**Acknowledgements.** We gratefully acknowledge the help provided by the fruitful discussions with our EDF colleagues M. Trotignon, J.F. Lesigne and A. Duchamp.

## References

- [1] L. Wehenkel, Th. Van Cutsem and M. Ribbens-Pavella, "Artificial intelligence applied to on-line transient stability assessment of electric power systems", *Proc. 25th IEEE Conf. Decision and Control*, Athens, Greece, Dec. 1986, pp. 649-650.
- [2] L. Wehenkel, Th. Van Cutsem and M. Ribbens-Pavella, "An artificial intelligence framework for on-line transient stability

assessment of electric power systems", *IEEE Trans. Power Systems*, Vol. PWRS-4, 1989, pp. 789-800.

- [3] L. Wehenkel and M. Pavella, "Decision trees and transient stability of electric power systems", *Automatica*, Vol.27, No.1, 1991, pp. 115-134.
- [4] L. Wehenkel, Th. Van Cutsem and M. Ribbens-Pavella, "Decision trees applied to on-line transient stability assessment of electric power systems", *Proc. IEEE Int. Symp. on Circuits and Systems*, Vol.2, 1988, Helsinki, Finland. pp. 1887-1890.
- [5] R. Fishl, T.F. Halpin and A. Guvenis, "The application of decision theory to contingency selection", *IEEE Trans. Circ. and Syst.*, Vol. CAS-29, 1982, pp. 712-723.
- [6] F.F. Wu and Y.K. Tsai, "Probabilistic dynamic security assessment of power systems", *IEEE Trans. Circ. and Syst.*, Vol. CAS-30, 1983, pp. 148-159.
- [7] L. Breiman, J.H. Friedman, R.A. Olshen and C.J. Stone, *Classification and Regression Trees*. California: Wadsworth International, 1984.
- [8] R. Belhomme and M. Pavella, "A composite electromechanical distance approach to transient stability", *IEEE Trans. on PS*, Vol. PWRS-6, No. 2, pp. 622-631, May 1991.

## APPENDIX

### A Data base generation parameters

#### A priori distributions of independent variables

**Remaining system.** Since by construction the study region accounts for all major effects on the plant stability, the loads, topology and generation schedule are essentially kept constant outside it. A few very large and noninteracting power plants of the remaining system are however used to compensate for the local active load/generation imbalance, according to a fixed priority list.

**Study region.** The main parameters used to draw randomized predefault operating states are topology, active generation / load, voltages, as outlined below.<sup>9</sup>

**Topology.** It is defined by the 18 regional 400 kV lines and by the number of nodes in the 4 substations represented in Fig. 4.

**Line outages.** 10% of the states are generated with a complete (i.e. base case) topology, 50% with a single line outaged, selected randomly among the 18 lines of the region; the remaining 40% correspond to the simultaneous outage of two "interacting" lines: 40 pairs of interacting lines were defined, consisting of lines either in parallel in a same transmission axis, or emanating from the same bus.

**Configuration of the study plant substations.** Substation 1 is restricted to 1 node (breakers 12 and 34 are closed) if a single generation unit is in operation; otherwise it is 50% of the time configured with 1 node and 50% with 2 nodes, and so are substations 3 and 4. Substation 2 is 90% of the time configured with 2 nodes.

**Load.** The total regional active load level is drawn according to a Gaussian distribution of  $\mu = 3500 MW$  and  $\sigma = 1000 MW$ . It is distributed on the individual load buses homothetically to the base case bus load. The reactive load at each bus is adjusted in order to keep a constant load factor ( $\frac{Q}{P} \approx 0.15$ ).

**Active generation.** The active power generations of the three power plants are defined independently, according to the following procedure.

<sup>9</sup>Although the random generation is very closely tailored to the specific study plant, it might quite easily be transposed to the study of other power plants or regions of the EDF system.

**1. Unit commitment.** Given a plant, the number of its units in service obey a plant specific distribution. Thus, for plant 1, 0 to 4 machines may be in service, according to a priority list, and with uniform probabilities. For plant 2, the 4 following combinations are used : no unit in operation (10%); either unit 1 or unit 2 in operation (30%); both units 1 and 2 simultaneously in operation (60%). For the study plant, 10% of the cases correspond to a single unit in operation, 20% to 2 units, 30% to 3 units and 40% to 4 units; the units being committed are drawn randomly, under the restriction of an as uniform as possible share of the generation on the two nodes of the substation 1, if the latter is configured with 2 nodes.

**2. Active power generation.** Once again, to maximize the interesting cases the rules are plant specific. For plants 1 and 2, a random generation is drawn in the interval of the global feasibility limits of its operating units. For the plant 3 of Fig. 4, the first two units in service are rated at their nominal power of 1300 MW each, the next two are rated according to a random number drawn in the feasibility limits of the units. This enables the generation of a maximum number of highly loaded situations, without losing information about intermediate, albeit less realistic cases.

**Voltage profile.** A simple strategy was used to produce sufficiently diverse voltage profiles, nearby the study plant. The EHV setpoint of its operating units is drawn randomly in the range of 390 kV to 420 kV, independently of the local load level. Furthermore, the voltage setpoint of plant 1 (the next nearest one) is drawn in the same range and independently. This produces a quite diverse pattern of reactive generations and flows in the study region (see below).

#### A posteriori distributions of dependent variables

The above randomized modifications of the base case provide, via load flow computations, the 3000 operating states of the data base. The diversity of situations covered by them is reflected by the statistical distributions portrayed in Figs. 5 corresponding to key variables of the study plant. Figure 5a sketches the total active generation of the plant : the vertical bars at 1300 MW (resp. 2600 MW) represent the number of operating states (OSs) with one (resp. two) units in operation, rated at their nominal power; the bars between 2850 MW and 5200 MW represent the OSs where at least three units are in operation, two of which are rated 100% and the remaining at an intermediate level, ranging from 250 to 1300 MW. Figure 5b shows a typical distribution of the reactive generation of a given unit of the study plant; its Gaussian shape nicely reflects the regional load pattern.

Figure 5c shows the multimodal CCT distribution of the busbar fault. The OSs around 235ms correspond to the great majority of "normal" situations; those near 0ms correspond to topologically exceptionally weak OSs; those above 350ms to "unusually" stable states. Figure 5d illustrates that the sole attributes P-M4 and V-M4 are unable to properly separate the stable (●) and unstable (+) states, despite the important role played by these attributes in the tree DT1 of Fig. 7.

## B Defining families of candidate attributes

Below we describe briefly the 13 lists of candidate attributes used during our application study.

**1a. Controllable attributes.** This minimal list contains the 38 following variables : (i) active generation of each unit of each plant of the region; (ii) their EHV voltage; (iii) global regional load; (iv) logical attributes describing the topology. Ideally, the DTs should rely on this kind of attributes only.

**2. Observable attributes.** The following 9 lists are composed of pre-fault attributes of growing complexity, i.e. of decreasing

contollability. Lists 2a-2e are elementary parameters of the operating state, easily available in a control center. Lists 2f-2i refer to composite attributes, combining information about several power system components, which however are restricted to be fault independent.

**2a = 1a + Reactive generation** of each unit of the region.

**2b = 2a + Power flows** on important lines

**2c (resp. 2d.) = 2b (resp. 1a.) + Phases** of the main substations of the study region, relative to the external load area.

**2e = 2b + Linear combinations** of P, Q and V allow to take into account with a single tree test the combined effect of two different characteristics via a linear combination<sup>10</sup>.

**2f = 2b + Topology combinations**

**2g = 2f + Power combinations**

**2h = 2g + Short-circuit admittances/powers** quantify the combined effect of line outages as well as substation and plant configuration on the strength of the topology.

**2i = 2h + Linear combinations** of P, Q and V

**3. Complex attributes** obtained by including in list 2g attributes of arbitrary complexity possibly taking into account the during and / or postfault configuration. They generally yield simpler and more reliable trees, but require more complex computations and a certain expertise to use them.

**3a = 2b + "Ad hoc" combinations** suggested by prior experience and physical interpretations. They take into account the effect of topology and electrical status on the accelerating power during the fault on period and of the number of available lines to exchange the stored energy during the postfault swings.

**3b = 3a + Postfault information** provided in the form of equivalent postfault equilibrium parameters (Thévenin e.m.f., power angle, maximal electric power ...) of an empirical "one machine infinite bus" representation used by planning engineers, as a rule of thumb for first shot stability assessment.

**3c = 3a + Linear combinations** of P, Q and V<sup>11</sup>

## Biographies

**Louis Wehenkel** was born in Nurnberg, Germany, in 1961. He received the Electrical (Electronics) engineering degree in 1986 and the Ph.D. degree in 1990, both from the University of Liège, Belgium, where he is presently a research assistant of F.N.R.S. (Fonds National de la Recherche Scientifique). His research interests lie mainly in the application of artificial intelligence methods to power system security.

**Mania Pavella** received the Electrical (Electronics) engineering and the Ph.D. degrees from the University of Liège, Belgium, where she is presently a Professor. Her research interests lie in the field of electric power system analysis and control.

**Edwige Euxibie** was born in Amiens, France, in 1962. She received the Electrical engineering degree in 1984, and the Electrical DEA degree in 1990 (Diplôme d'Etudes Approfondies), both from the Ecole Nationale Supérieure d'Ingénieurs Electriciens de Grenoble, France. She is presently a research engineer in the "Control and Operation of Power Systems" group, R&D Dpt of EDF. Her research interests lie mainly in the area of power system operation.

**Bertrand Heilbronn** was born in Paris, France, in 1950. He received his degree of engineer from the Ecole Polytechnique in 1972, and from the Ecole Nationale Supérieure de Télécommunications in 1974. He has been working as a research engineer at EDF since 1978, and since 1985 in the field of power system control. He is presently manager of the the research group "Control and Operation of Power Systems", in charge of power system monitoring and control studies.

<sup>10</sup>The inductive inference method was generalized to treat linearly combined numerical attributes. Thus, for two attributes (e.g. P and Q), the method searches for a test of the type  $P + \beta_1 Q < \beta_2$ , by tuning  $\beta_1$  and  $\beta_2$  in an optimal fashion.

<sup>11</sup>Notice in Table 2 the good performances of DT13 and DT26 obtained by using attributes of this type.

### Discussion

**J. L. Carpentier**, Electricité de France, Clamart, France: The authors are to be commended for their excellent work in a particularly difficult field. Some comments would be appreciated on the two following points:

1. The standard decision tree, as proposed in this paper, approximates the stability boundary in a staircase fashion, i.e. by combining hyperplanes which are orthogonal to the attributes, with elementary attributes equal to state or control variables of the system. Perhaps a better or more robust approximation could be obtained either by using arbitrary hyperplanes, i.e. by splittings involving linear combinations of elementary attributes, or by using functions of several state or control variables as attributes. The authors' comments on the practical feasibility of such an approach would be highly appreciated.
2. The authors have developed multi-contingency decision trees where an operating state may be classified stable with respect to some of the constraints and unstable with respect to some others. Another approach could be to assess the simultaneous stability of an operating state with respect to a complete set of contingencies. Could the authors expand about this kind of approach and discuss its potential?

Manuscript received February 18, 1993.

**Robert Fischl** (Drexel University, Philadelphia, PA): The authors are to be congratulated for their work on the assessment of the Decision Tree Transient Stability (DTTS) method in terms of: (i) its ability to identify the transient stability limiting contingencies of large system such as the EDF system; (ii) the amount of data needed to do this; and (iii) the evaluation of the DTTS method's contingency classification performance.

Of particular interest to me are the results shown in Figs. 5, 6 and 8. The results shown in Fig. 5 are for the base case. It would be interesting to see: (a) how the results of Figs. 5b, 5c and 5d change as a function of each contingency; and (b) the effect of the threshold,  $\tau$ , on the results of Fig. 5d. Is the extreme intertwining of cases (\*) and (+) in Fig. 5d mainly due to the uncertainty in the network topology? What type of statistics were used for the line outages and configuration of the substation?

In reference to the histograms shown in Figs. 6 and 8, it would be more informative to give separate histograms for the  $\Pr\{Q_1 = \text{false alarm}\}$ ,  $\Pr\{Q_2 = \text{miss}\}$  and  $\Pr\{Q_3 = \text{gross errors}\}$  as a function of the CCT. In addition, a useful tool (Refs. [D1] and D2) we have found for evaluating the performance of decision aids for security is the performance curve of  $\Pr\{\text{Correct detection of instability}\} = [1 - \Pr\{Q_2 = \text{miss}\}]$  vs  $\Pr\{Q_1 = \text{false alarm}\}$  as a function of the various values of the threshold,  $\tau$ . This performance curve can be used to determine how effective your technique no. (i) is in improving  $Q$ , and probably more important, it will enable you to systematically identify the most effective attributes and their optimal values which will reduce the  $\Pr\{Q_2 = \text{miss}\}$  to an acceptable value.

[D1] J-C Chow, Q. Zhu, R. Fischl, M. Kam, "Design of Decision Fusion Rule for Power Security Assessment," IEEE 1992 PES Summer meeting, paper No. 92 SM 600-7 PWRs, Seattle, WA., July 1992.

[D2] J-C Chow, R. Fischl, "Probabilistic Evaluation of Security Criteria in Power System Security Assessment," submitted to the IEEE 1993 PES Summer meeting.

Manuscript received March 3, 1993.

**L. Wehenkel, M. Pavella, E. Euxibie and B. Heilbronn.** We thank the discussers for their interest in our work. Their thoughtful questions give us the opportunity to clarify and comment on particular aspects of the DT approach.

Before answering specific questions, we re-emphasize that DTs applied to power system security (and in particular to transient stability) pursue a much broader objective than mere classification, which is the current aim of any pattern recognition approach. Indeed, in addition to this latter, DTs are able to decompose-describe-interpret the mechanism of the physical problem. This latter aspect is an essential asset of the DT approach, and makes it much closer to the human thinking than many other approaches, of the "black-box" type; not only does it help power system operators understand the phenomena and take decisions, but also it helps DT designers take full advantage of human expertise. Now, the above two objectives (classification ability and interpretability) make the DT design (e.g. in terms of types of candidate attributes) depend upon the priorities. In particular, one may realize a compromise or fully put the accent on one of the objectives; in the former case a sole type of DTs may be designed, in the latter it might be more appropriate to design two distinct families of trees.

**Dr. J. Carpentier.** 1. In its basic form, DTTS uses a single attribute at each test node. This has as consequences computational efficiency (allowing to handle very large problems with hundreds of variables and thousands of learning states), and more importantly, interpretability of the DTs. But we agree that, on the other hand, this simplicity might lead to suboptimality in terms of robustness. To circumvent this difficulty, an interesting possibility is to use functions of several attributes at each test node. Several approaches have been suggested in the literature for linear or non-linear functions of real valued (e.g. electrical) attributes, and logical combinations of discrete (e.g. topological) attributes. Such an approach is briefly mentioned in the paper, in footnote 10 of Appendix B. This approach is however restricted to a small number of such combinations, due to computational burden. A more systematic and efficient approach is the following extended DT growing procedure: the list of candidate attributes may be replaced by a list of candidate sets  $S_i$  of attributes; for each candidate set  $S_i = \{a_{i1}, a_{i2}, \dots\}$ , the optimal splitting procedure would then search for an "optimal" hyperplane  $\sum_{a_{ij} \in S_i} \beta_j * a_{ij} \leq \beta_0$ . This is basically the perceptron search problem solved by the back-propagation algorithm, provided that a smooth optimality criterion is used [C1]. Yet another approach to the construction of appropriate complex combinations of attributes is described below, in an answer to Prof. Fischl.

Whatever the solution, one should keep in mind that more complex attributes may result in somewhat (although not necessarily significantly) better reliability, but to the detriment of interpretability.

2. The paper has almost exclusively focused on the fundamentals of the DT method, by considering single-contingency trees, which assess stability with respect to a given contingency. Only a brief mention was made to multicontingency DTs of the "contingency-dependent" type; here the stability of a state is assessed with respect to a given set of known, explicitly identified contingencies. (See § 4.5 of the paper). The results of a general study on multicontingency DTs are reported in Ref. [C2], where two distinct approaches are proposed. One goes along the above "contingency-dependent" multicontingency DTs - although in a somewhat different way. The other considers so called *global* DTs (GDTs) aiming at the identification of the stability limits of a system, in a "contingency-independent" fashion. Such GDTs would therefore classify an operating state as stable if it is stable *simultaneously* with respect to each contingency. GDTs have interesting properties and applications. In particular, for the real-time operation of a power system where their synthetic information about stability limits of a subsystem would be useful to identify secure regions of operation. Thus, GDTs seem to be appropriate for stability assessment involving an operator,

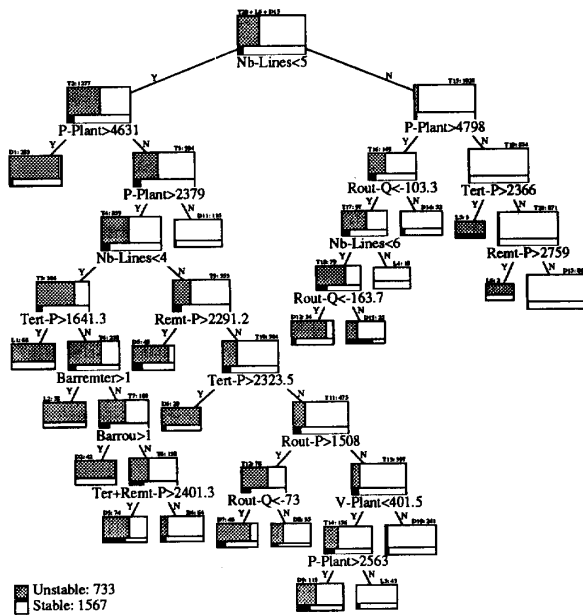


Figure C1. A "global" decision tree

e.g. as a tool for sensitivity analysis and for preventive control. Figure C1 portrays such a global tree built for 14 contingencies, defined for the study plant substation of Fig. 4 of the paper : 6 "single-line" faults, 6 "double-line" faults, 2 "busbar" faults. (See their definition in § 3.1 of the paper).

**Prof. R. Fischl.** 1. Concerning the intertwining of stable and unstable cases of Fig. 5, it is indeed the effect of variable topology on the stability limits, in particular the number of nodes in substations 1 and 3 and/or the prefault line outages, which may lead to various limits. This is clearly shown by the DTs of Figs. 6 and 7 of the paper : the test attributes selected by the trees

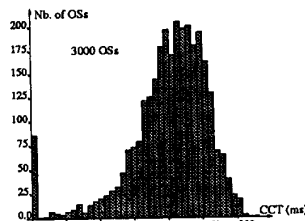


Figure C2. CCT distribution of double-line fault #2

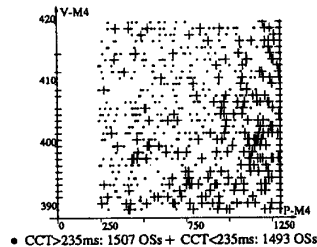


Figure C3. Plotting  $V-M_4$  vs.  $P-M_4$  vs.  $\tau = 235$  ms

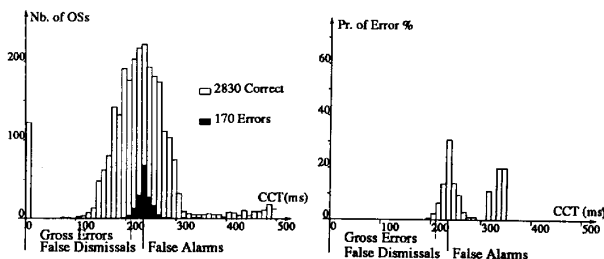


Figure C4. CCT distribution vs. types of errors

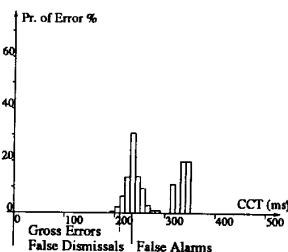


Figure C5. Error probability vs. CCTs

carry a good deal of information about topology. Figure C2 on the other hand shows the distribution of CCTs for the double line fault. The main difference is that there are no "very" stable states, due to the fact that there is always at least one unit at full rating remaining in operation in the postfault, in the study plant. As concerns the outlook of Fig. 5d, if  $\tau$  is increased or decreased slightly (e.g. by 10%), the number of unstable states will decrease or increase slightly, but qualitatively the diagram will not change. If however  $\tau$  gets close to the median of the CCT distribution, the intertwining becomes significantly worse, as is shown by Fig. C3 where  $\tau = 235$ ms was used to classify the states. The statistics used for line outages and busbar configuration are given in Appendix A of the paper.

2. The probability curves proposed by Prof. Fischl would indeed be informative. So far, for the analysis of errors we found it also interesting to use histograms as the one given in Fig. C4, corresponding to the diagram of Fig. 8 for DT26 of the paper. In addition to the number of states falling in a certain CCT range, the diagram shows the proportion of correct classifications and of errors. Depending on the value of the CCT these are either gross errors ( $CCT < 0.9\tau$ ), or false dismissals ( $CCT \leq \tau$ ) or false alarms ( $CCT > \tau$ ). Thus, the proportion of the black area to the overall height of the bars provides the error probability as a function of the CCT. The corresponding curve is provided in Fig. C5.

The performance curves are indeed a convenient tool to assess the effectiveness of a technique to reduce different types of errors. This kind of approach was actually used in order to determine appropriate non-detection costs for the stable and unstable cases. These costs are used by the technique (iii) of the paper in § 4.6.1 in order to assign the appropriate class to a terminal node. In complement to the different techniques suggested in the paper, we mention a recently developed hybrid DT-ANN approach [C3]. It consists of first building a DT on the basis of a discrete stable/unstable classification, then deriving therefrom a multilayer perceptron type of ANN, the weights of which are adapted so as to allow the "smooth" approximation of the CCT in terms of the DT test attributes. In short, this hybrid approach attempts to exploit the advantages of each of the DT and ANN methods (interpretability properties of the trees and very high reliability for the ANNs) while avoiding their drawbacks (loss of part of the information contained in the learning set for the trees; inability for the ANNs to properly select the essential attributes only, resulting in heaviness and "black-box" type of information). The first results reported in [C3] show that the use of CCT information allows indeed to improve effectively the precision of the provided stability information, in terms of an approximate CCT. We believe that the performance curves mentioned by Prof. Fischl would also be an interesting tool in this context, for example in order to determine appropriate CCT thresholds to define security regions.

[C1] P.E. Utgoff. Perceptron trees : a case study in hybrid concept representation. In *AAAI-88. Proc. of the 7th Nat. Conf. on Artificial Intelligence*, Morgan Kaufman, 1988, pp. 601-606.

[C2] V.B. Akella, L. Wehenkel, M. Pavella, M. Trotignon, A. Duchamp, and B. Heilbronn. Multicontingency decision trees for transient stability assessment. *Proc. of the 11th Power Systems Computation Conference*, Aug-Sept 1993.

[C3] L. Wehenkel, V.B. Akella. A hybrid decision tree - neural network approach for power system dynamic security assessment. *Proc. of the 4th Int. Symp. on Expert System Application to Power Systems*, Melbourne, Australia, Jan. 4-8, 1993.

Manuscript received April 2, 1993.