

Visualizing Dynamic Power System Scenarios for Data Mining

Pierre GEURTS and Louis WEHENKEL[†]

Dept. of Electrical Engineering - University of Liège - Sart Tilman B 28, B 4000 Liège - Belgium

[†]Research Associate, F.N.R.S.

Abstract — This paper presents a power system dynamic scenario visualization tool which is used to support user interaction in the context of data mining. Data mining may be used in order to extract important synthetic information from data bases containing power system dynamic trajectories under various conditions, for example generated by simulations or obtained from historical records. Data mining relies on automatic learning algorithms, such as decision trees or neural networks, to extract information and uses various graphical tools to communicate this information to the user. The paper describes the type of information stored in dynamic security assessment data bases, the purpose of data mining and the tools developed. A postscript version of this paper, showing the graphics in color is available from the world wide web at URL <http://www.montefiore.ulg.ac.be/~lwh/>.

Keywords — Data mining; Automatic learning; Power system dynamics; Graphical visualization

1 INTRODUCTION

Like many other application areas, the power system field is presently facing an explosive growth of data. In power systems, irrespectively of the particular application, there are three main sources of data: (i) field data, collected by various devices distributed throughout the system, such as digital recorders; (ii) centralized data archives, such as those maintained by control center SCADA systems; (iii) data from simulations, carried out in planning or operation environments. Trends in technology (e.g. in digital protection and recording, control center architectures, simulation software and hardware...) result in more and more data generation and dramatic cost decrease of mass storage devices make it possible to store them efficiently.

However, it will not be possible to exploit effectively such huge amounts of data (maybe TERAbytes), unless appropriate software tools are developed to help experts to extract meaningful information from it. Note that, due to the stochastic nature of power systems, the available data will essentially be of statistical nature. Hence, exploiting it properly will call for a tool box of statistical knowledge extraction techniques (i.e. automatic learning) in order to fit the variety of power system problem types (linear/non-linear, local/global, smooth/discrete, static/dynamic...). At the same time, data will be corrupted by noise, outliers, or missing information and appropriate data cleansing methods will also be necessary. In addition, such

data mining tools will require appropriate graphical front-ends to enable information exchange with power system engineers.

The development of such software environments and of the underlying methods, is the purpose of the recently emerged fields of Knowledge Discovery from Data Bases (KDD) and Data Mining (DM) [1]. Note that within the KDD community, it is well recognized that DM and KDD tools need to be tailored to application specifics in order to be effective. Some important power system specific aspects of DM applications are: the large scale character of power systems (thousands of state variables); the temporal nature of data and the coexistence of very fast phenomena (ms) with slower trends (minutes, hours, weeks, years) all being important for some problems; mixture of discrete (e.g. events such as topology changes or protection arming) and continuous information (i.e. analog state variables); necessity of relating information to geography and topology of the power system, i.e. necessity of using one-line diagrams and power system maps to communicate with the experts [2].

The purpose of this paper is to present useful data visualization techniques (some generic and some power system specific) which are enhancing a Data Mining software developed at the University of Liège. This software incorporates various automatic learning methods (statistical, artificial neural networks, and machine learning) tailored to power system data mining problems, together with appropriate graphical user interfaces. It was progressively developed during the last 10 years, in the context of applying automatic learning techniques to power system dynamic security assessment and is able to handle large scale applications.

The paper is organized as follows. Section 2 first describes the type of information contained in power system security assessment data bases, and then focuses on both generic and special purpose visualization techniques. Section 3 describes the developed tools from the software implementation point of view. Section 4 provides examples from a real large scale data base created for dynamic security assessment of the EHV system of Electricité de France [3]. In the conclusions we briefly discuss the remaining work to be done and review a certain number of applications where the developed tools could be useful.

2 DATA MINING AND VISUALIZATION

2.1 Dynamic security information data bases

Although much of our discussion is also relevant to other power system data mining problems (some will be mentioned in the conclusions) we will concentrate on data bases containing information about the dynamic behavior of a power system under various conditions. Such data bases are presently mainly obtained from simulation studies, but in the future it may be envisioned that similar information could be provided by real-time wide area measurement systems.

Although we will not focus our discussion on the way such

	Attributes (characteristics)								
	Temporal				Non temporal				
	Curves		Events		Numeric		Discrete		
Objects (scenarios)	ant1	...	adt1	...	an1	...	ad1	...	sc1
									sc2
									...
									scN

Figure 1. Data base structure (3rd dimension : time)

data bases are obtained in practice, let us mention that one possible methodology consists of using Monte-Carlo sampling together with parallel computations, in order to generate automatically such data bases in off-line study environments (planning or operation planning) [4]. Another possibility would be to store systematically in a data base the simulations carried out manually by the engineers in charge of design and security assessment, or those that are run by an on-line dynamic security assessment tool coupled with the energy management system of a control center [5].

2.1.1 Data base structure

Before discussing which kind of tools may be useful in order to extract useful information from such data bases, let us briefly describe their general structure.

A data base, as we view it, is composed of a certain number of *objects* each one of which is described by a certain number of *attributes* (see Fig. 1). In a dynamic security information data base, each object essentially corresponds to a dynamic trajectory of the studied system under some particular conditions; we use the term *scenario* to denote such objects. The attributes correspond to characteristics of various devices in the system, and more generally to variables deemed interesting. We may distinguish among the following types of attributes.

Temporal. These attributes are used in order to describe what happens along the system trajectory; there are mainly two types of such temporal attributes :

Numerical curves. Examples are the variation with time of the values of rotor angles, voltage magnitudes, currents. . .

Discrete events. Examples are time tagged sequences of events indicating various relay armings, breaker trippings, fault occurrences. . .

Non temporal. These attributes provide static characteristics about a particular scenario, such as lists of available plants, lines, transformers, load demand. . . Again we may distinguish among numerical and discrete attribute values. In addition, it is worth noticing that non temporal attributes may be derived as functionals of temporal ones (e.g. initial or final value of a state variable, total number of elements of a particular type tripped during a certain time period, stability/instability classification. . .)

As is illustrated in Fig. 1, a dynamic security information data base may be viewed as a three-dimensional structure, which is normally filled in a sparse fashion. Indeed, not all attribute

values are meaningful for all scenarios (e.g. if a line is out of operation, the corresponding current need not to be represented). Furthermore, along the time axis the values of the temporal attributes are only represented at the relevant instants (e.g. events when they occur; curves with variable time steps adapted to their dynamics in a particular case). In addition, some attributes may be implicitly represented as functions of others, and are computed on the fly upon request (e.g. total generation in the system as a sum of individual generators active power).

In large scale dynamic security assessment data bases, the number of scenarios may be very large (typically, a few thousand) as well as the number of attributes (typically, a few hundred), and time-constants may range from milliseconds to several minutes. Hence it is quite important to take advantage of the sparse structure of the data base, in particular along the time axis.¹

2.1.2 Physical world

While in a data base attributes are abstract entities, they generally correspond to devices or sets of devices which are well defined in the physical world (e.g. voltages are related to buses, currents to lines). Moreover, in the physical world various relationships exist between entities (e.g. a line is related to the buses it connects).

An engineer who is working with such a data base will think about the power system according to the physical world relationships, and it is hence useful to allow the data mining tool to be aware of these latter. In the sequel we suppose that the physical world relationships are given in the form of an external data base, the so-called *physical data base*, and links are established between this latter and the dynamic security information data base. More precisely, the data mining tool is supposed to know for each attribute (as well as for discrete attribute values) to which entity they are related in the physical world.

2.2 Data mining

“Data mining” is a buzzword which denotes a set of activities aiming to extract interesting information (“data nuggets”) from data bases. In power system dynamic security information data bases, it can be used for example to build rules characterizing secure and insecure scenarios, or to assess the influences among different parameters, or to evaluate how accurately one can predict future states or events from past information, or to determine criteria to rank contingencies, or to evaluate consequences of unstable behaviour. . .

In order to help extracting such a large diversity of types of informations, data mining tools offer a wide range of functionalities, and in particular the following ones:

Subview selection. This allows the user to select part of the scenarios and/or part of the attributes in order to focus his analysis. Subsets of scenarios may, for example, be selected on the basis of some property defined on their attribute values, by name, or as a random subset of the data base. Attributes are generally chosen manually by the expert, according to the task he wants to carry out.

Sorting and grouping. Attributes may be sorted or grouped by correlations w.r.t. a reference attribute; objects are sorted or grouped by attribute values or by distance to a prototype.

¹2000 scenarios, 500 attributes, fixed time-step of 10ms and 10 minutes of scenario duration, would lead to 200GB of memory requirements.

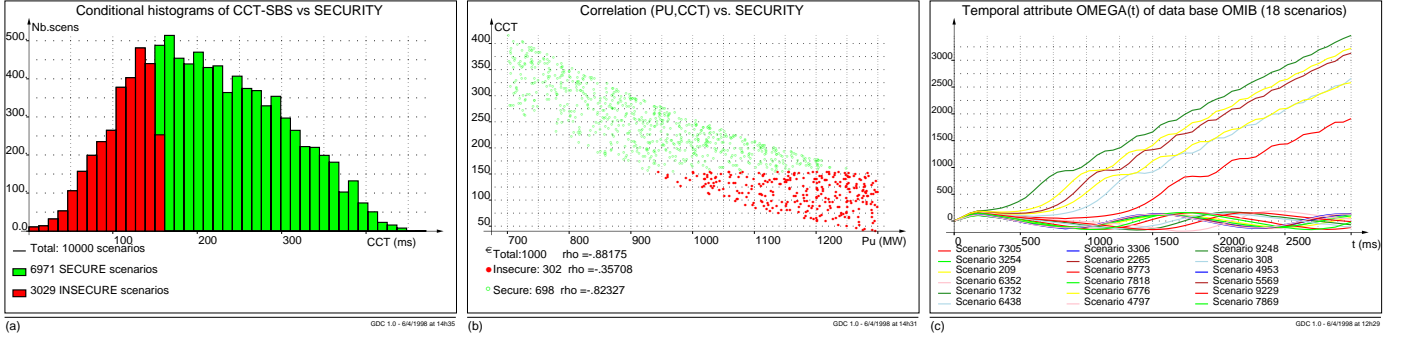


Figure 2. Examples of generic visualization techniques : (a) histograms; (b) scatter plots; (c) temporal curves

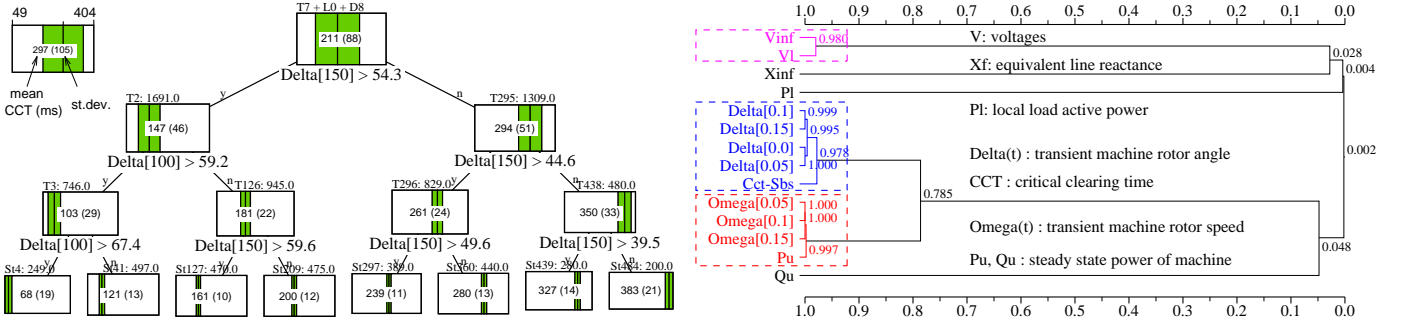


Figure 3. Data mining results visualization: (a) regression tree (CCT function of $\delta(t)$); (b) correlation dendrogram

Temporal transformations. They consist of shifting the time origin (e.g. to synchronize scenarios according to some event), increasing or decreasing time granularity or truncating it, in order to focus on a particular aspect of the problem.

Attribute definition language. This very important facility enables the user to define new attributes by combining values of existing ones. For example, he could easily combine elementary characteristics into more synthetic ones, and hence implement various possible criteria.

Automatic learning. Algorithms are provided in order to build automatically function attributes (in various forms) so as to approximate a given attribute as a function of some others (e.g. decision trees are used to approximate discrete attributes; linear or non-linear regression may be used to approximate numerical attributes. . .)

Similarity analysis. Algorithms are provided in order to evaluate similarities among scenarios, find the most similar scenario to a given one, build a set of representative prototypes, analyse correlations among attributes. . .

Visualization. Graphical tools are used in order to view information such as attribute values for sets of objects, and results of automatic learning.

In the sequel, we focus on visualization techniques. We refer the interested reader to [2] for a detailed description of relevant automatic learning and similarity analysis methods.

2.3 Generic visualization techniques

Generic visualization techniques are those which are useful for any type of data mining activity, power systems related or not. Figure 2 illustrates three basic types of graphics which are very useful. Note that they are drawn from an academic

example data base related to transient stability assessment of a one-machine-infinite-bus (OMIB) system.

The first two are related to non temporal attributes : *conditional histograms* (Fig. 2a) provide information about the distribution of a given numerical attribute (horizontal axis) in relation to a discrete attribute (colors or grey levels); *scatter plots* (Fig. 2b) provide information on the relations between two numerical attributes (horizontal and vertical) and a discrete one (colors and markers). Figure 2a represents the distribution of critical clearing times (CCTs) of the OMIB system, among the 10,000 states of the data base, classified into secure and insecure scenarios (w.r.t. a threshold of 155ms on the CCT). Figure 2b shows the relationship between the active power generated by the machine and the CCT values, indicating a quadratic shape and also for each active power level the range of CCTs possible due to the effect of other parameters. Figure 2c provides a plot of a temporal attribute for some scenarios : the variation of the mechanical speed of the machine, when the fault is cleared at $t_{cl} = 155ms$.

Figure 3 illustrates two generic visualization techniques for data mining results.

Figure 3a (left part) shows a (partial view of a) regression tree which approximates the CCT of the OMIB system as a piecewise constant function of the machine rotor angle in the during fault period. Note that the shaded area and the number in each box provide information about the mean and standard deviations of CCTs of the scenarios corresponding to each node. The top-node corresponds to all possible scenarios, whereas the terminal nodes correspond to a subset of scenarios falling within a certain range of $\delta(t)$ values. For example, the lower left node corresponds to scenarios such that $\delta(150) > 54.2^\circ$ and $\delta(100) > 67.4^\circ$; for this kind of scenarios the mean CCT value is 68ms and its standard deviation is equal to 19ms. Similarly, the lower right node corresponds to scenarios such that $\delta(150) \leq$

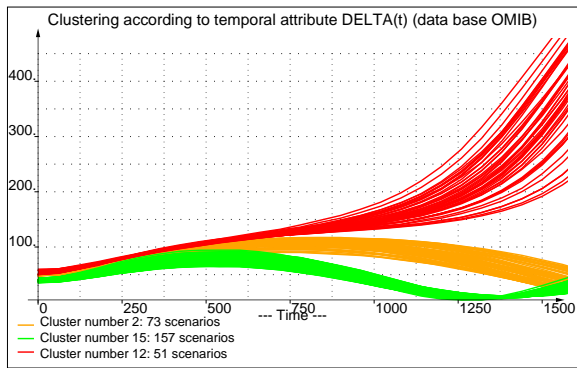


Figure 4. Example of visualization of scenario clustering

39.5°; they have a mean CCT of 383ms and a standard deviation of 21ms. Thus, the regression tree allows one to approximate the CCT from rotor angles in the during fault period; in our example this approximation is actually quite accurate, the mean absolute error being about 2ms.

Figure 3b (right part) shows a dendrogram summarizing a correlation analysis among various attributes, and three groups of attributes which may be considered as highly correlated (the numbers refer to correlation coefficients among variables computed on a representative subset of the data base).

Figure 4 illustrates how curve coloring can be useful to analyse the results of a clustering algorithm. A random sample of the data base was automatically partitioned by a clustering algorithm into 15 classes, according to the values taken by the temporal attribute $\delta(t)$, i.e. rotor angle temporal behaviour. The $\delta(t)$ curves of the scenarios belonging to three of these clusters are shown in Fig. 4.

2.4 Requirements of specific visualizations

Power system specific tools for visualization are mainly using two-dimensional maps, where different icons are used to represent different types of elements and colors in order to highlight groups of elements which share a common property (same voltage level, same island, same status on/off...). In addition, some fields of these maps are showing numerical information (voltage magnitudes, power flows...). Given the complexity of most power systems, it is normally impossible to visualize all elements in full detail without overloading the picture. Thus, tools must be able to display information at different levels of detail, and allow the user to easily switch from one level to another. In addition, graphical maps should of course provide easy to use zooming, focusing and navigation mechanisms.

These facilities are well known in the control center community; some interesting research has also been carried out in order to design special features useful in the context of displaying results from security assessment [6, 7, 8]. We will not elaborate on the general features of such displays, which are clearly also required in the context of data mining applications. Rather, we will discuss special features which need to be added to these basic tools in order to provide a tool to engineers in charge of data mining security information data bases. In passing we notice that, while the graphic tools provided in the real-time environments of control centers are indeed highly sophisticated, those at the disposal of engineers in charge of studies in the off-line environment are often very minimal.

The following provides a (probably non exhaustive) list of desirable features of such visualization tools:

2.4.1 General features needed in the study environment

Focusing and searching objects on the map. Since a rather large number of attributes and physical objects may be represented on a map, it is useful to provide some focusing mechanism which automatically centers the display on the relevant part of the map. Another useful feature is to allow the user to input physical object names for focusing. Notice that these features are specially useful since the user is not necessarily fully acquainted with the layout.

Flexibility. Since in the off-line study environment a very large variety of problems are considered, the tool must be able to display any kind of information.

Easy and incremental editing. In particular, it should be easy for an engineer to edit graphic layouts, add or remove information on them, since it is not possible to design in advance graphical layouts for any kind of application.

Paper version of the graphics. In order to write reports it should be possible to generate automatically paper versions of maps, for a specified set of scenarios, and with a minimum amount of user intervention.

2.4.2 Data mining related features

Interaction with the data mining tool. The visualization tool should provide facilities to send and acquire information to and from the data mining tool. In particular, it may be very useful in order to display groups of attributes in different colors and select or unselect some attributes by clicking on the corresponding graphical representation. It is also useful to allow the user to call some generic visualizations directly from the tool.

Multiple scenarios. It should be possible to switch quickly from one scenario to another, or to visualize simultaneously several scenarios, in order to compare their temporal behaviour in the data mining process.

Temporal motion. In order to visualize temporal attributes it should be possible to move forwards and backwards through time easily, or to skip to a particular event occurrence.

Most of these features have been incorporated in the tool described in the next section and will be illustrated below. Notice that they are quite different from the graphics requirements in the real-time environment.

3 SOFTWARE

Below we briefly describe the data mining software (called GTDIDT) and then focus on the implementation of the scenario visualization tool VISIONET. Notice that both have been developed in the research environment, and thus should be considered as prototypes rather than full-fledged software tools.

3.1 Data mining tool : GTDIDT

The data mining tool was developed gradually in the context of various research projects [9]. The features recently added concern the representation and manipulation of the (large amounts of) temporal data encountered in dynamic security information data bases. The overall architecture of GTDIDT is depicted in Fig. 5. It is composed of two main components.

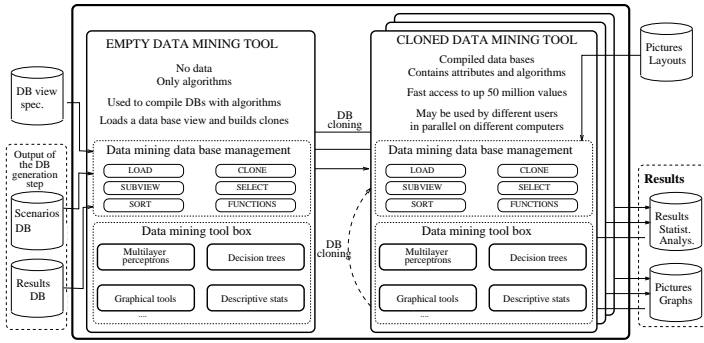


Figure 5. Data mining tool GTDIDT

Data mining data base management module. It loads data from flat files, while allowing one to select a subset of the scenarios and a subset of attributes, yielding a particular **view** on the data bases. These data are then compiled into internal representation and an executable image is cloned, containing the viewed data and the data mining algorithms. On a workstation with 1GB of main memory it is thus possible to access simultaneously up to 200 million attribute values. The data base management module allows the user to define additional attributes as functions of the basic ones extracted from the simulations and compiles them into efficient code. It provides also facilities for various subset and variable sorting/selection/grouping operations and time axis transformations.

Data mining tool box. This module contains both low level and high level algorithms in order to allow the user to extract knowledge from a data base. The former are basic statistical summarizations in text and/or graphical forms (means, standard deviations, correlations, bar diagrams, curves, scatter plots. . .). The latter are automatic learning algorithms, deemed useful for security assessment applications, e.g. unsupervised learning (clustering of similar types of scenarios or attributes), top down induction of decision and regression trees, and smooth linear and non-linear regression techniques. Notice that the illustrations in Figs. 2, 3 and 4 were all generated by GTDIDT.

GTDIDT is written in Lisp and runs under Unix/X11 : its user interface is a customization of the GNU-Emacs editor while the core of the tool is written in GNU CommonLisp (GCL). It generates generic graphics in the form of postscript files which are visualized using any available postscript previewer. The automatic learning algorithms generate their output mainly in the form of graphics and function attributes (lisp code) which are automatically compiled and added to the data base.

3.2 Security scenario visualization : VISIONET

VISIONET and GTDIDT have been developed independently; they run in parallel (different processes) and communicate using Unix sockets, as is illustrated in Fig.6.

VISIONET is written in Tcl/Tk: Tcl (Tool command language) is an interpreted command language; Tk is a graphical tool-kit for X11 integrated with Tcl. Rather than computational efficiency, the main features of Tcl/Tk are portability, open design (its integration with other languages was a main objective when it was designed), and user-friendliness (thanks to the provision of powerful graphical libraries). It was hence perfectly suitable for the development of a general power system visualization tool, where flexibility is more important than

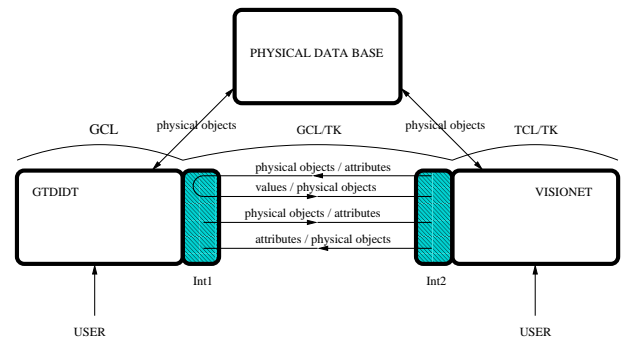


Figure 6. Interfacing GTDIDT and VISIONET

efficiency (in comparison to CAD/CAM, or image processing, for example).

The software may run either in stand-alone mode or coupled with GTDIDT. In stand-alone mode, starting with the physical power system model, it allows the visualization and editing of one-line diagrams, with features like zooming, searching of physical elements, hard-copy, easy to use graphics editor. The power system may be decomposed into several logical levels (typically corresponding to different voltage levels) which may be displayed together or separately, and also into geographical regions. Notice that, while the tool was designed for power systems, it could be used to represent the topology of any other kind of interconnected system (computer network, ground transportation. . .).

The coupling between this tool and GTDIDT was carried out using the 'gcl-tk' package, which provides the Tk facilities to GCL. GTDIDT may send directives to VISIONET (who then behaves as a graphics server), and it is possible to let GTDIDT take control over any existing Tcl/Tk program. The interaction works also in the other direction : Tcl/Tk may send expressions to GTDIDT (who then behaves both as a data and computation server) and get results back. The two applications are working in parallel, and it is thus possible to continue interacting with the graphic tool while GTDIDT is busy doing some computations. They take advantage of the Unix/X11 environment, which makes it possible to run them on different hosts, or to use several screens to display information.

As is shown in Fig. 6, both tools make use of the physical data base as a common vocabulary and information source : GTDIDT links its attributes with physical objects, Visionet links graphical objects (positions, icon geometries, colors. . .) with physical objects. To realize the coupling, GTDIDT was extended by two interfaces : int1 and int2 (see Fig. 6). Upon user request to GTDIDT, the latter creates a new Tcl/Tk window 'int2' which is attached to VISIONET; this window contains buttons and fields corresponding to the additional functionalities related to the data mining activities (visualization of attribute values, selection of candidate attributes, switching among scenarios. . .). During this initialization stage, VISIONET receives also information about the data mining environment in the form of GTDIDT names (scenarios, attributes).

In normal use, most of the interactions correspond to two-way communications. In order to serve a graphic request from the user, VISIONET will eventually call GTDIDT to get information (e.g. a particular attribute value, corresponding to a given scenario and time instant). Then, GTDIDT sends information back in the form of values together with a list of physical object

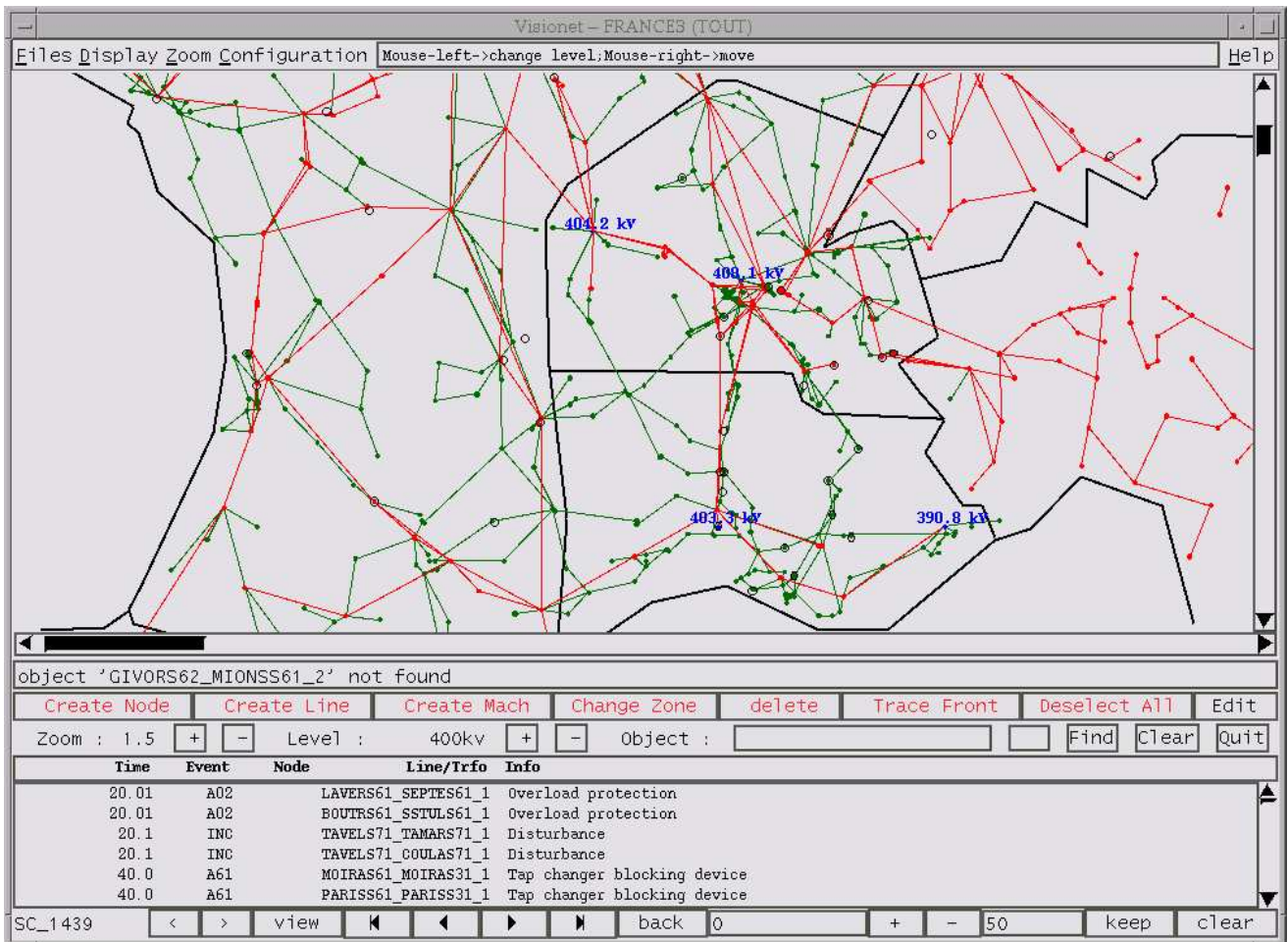


Figure 7. VISIONET graphical user interface outlook : lower part specific to "int2"; upper part is generic

identicators to which these values relate. In turn, from the list of physical objects VISIONET can derive its graphical information required in order to serve the user request.

Other types of interactions may be purely one-way. For example, VISIONET may be directly controlled by GTDIDT in order to reflect data mining results automatically in the Graphics. Similarly, VISIONET may act directly on the GTDIDT environment, e.g. in order to call some of its facilities (curve plotting, attribute selection...). In this latter case, physical object names are sent to GTDIDT who achieves the translation.

Figure 7 shows the graphical outlook of VISIONET's user interface. The lower part corresponds to the 'int2' window, and the upper part to the standard graphics display where the one-line diagrams are shown. The 'int2' window contains in its lower part a task bar with tape recorder buttons allowing to move through time, or through a list of scenarios, and above this a scrollable window allows the selection of groups of attributes and/or events to display on the map. On the top of this is located a task bar with navigation functionalities (zoom, level focusing, searching and showing object names). The task bar immediately above this latter contains editing functions and the 'Edit' button which allows to switch between editing and visualization modes. The main window (showing part of the 400kV and 225kV network of Electricité de France) contains objects representing the power system and attributes values. Rather than indicating the device names on the map, they are automatically displayed when the mouse moves over an object. Next section (and Fig. 8) illustrates some other features.

4 ILLUSTRATION

4.1 Case study description

We draw our illustration from a large scale study which was carried out on the EHV system of Electricité de France (see [3], for further details). The data base is composed of about 1500 scenarios, among which a certain proportion is highly disturbed. The aim of the study was to identify main system failure modes in the South-Eastern part of the system, by studying both long term and fast dynamics, and taking into account all kind of protective devices which may act when the system is highly disturbed. The simulation model comprizes 11,000 state variables and each scenario is simulated during a period of about 60 minutes of real-time, using a variable time step numerical integration method (EUROSTAG [10]). Below we will focus on a particular scenario which shows very complex and interesting phenomena, appropriate to illustrate our point.

4.2 Examples of visualizations on scenario 1439

Let us briefly describe the chronology of events happening within this scenario (see also Fig.8) :

Initialization (t=0s to t=505s). At t=20s a first (bus bar) fault appears in a 400kV² substation (see Fig. 8a), leading to immediate tripping of 225kV lines (overload protections working improperly). The fault is cleared at t=20.1s, by permanent tripping of breakers in the substation, leading to the loss of the two 400kV lines connected to the bus bar. At t=40s 12 tap-changer

²In the color version the 400kV part is in red, and the 225kV in green.

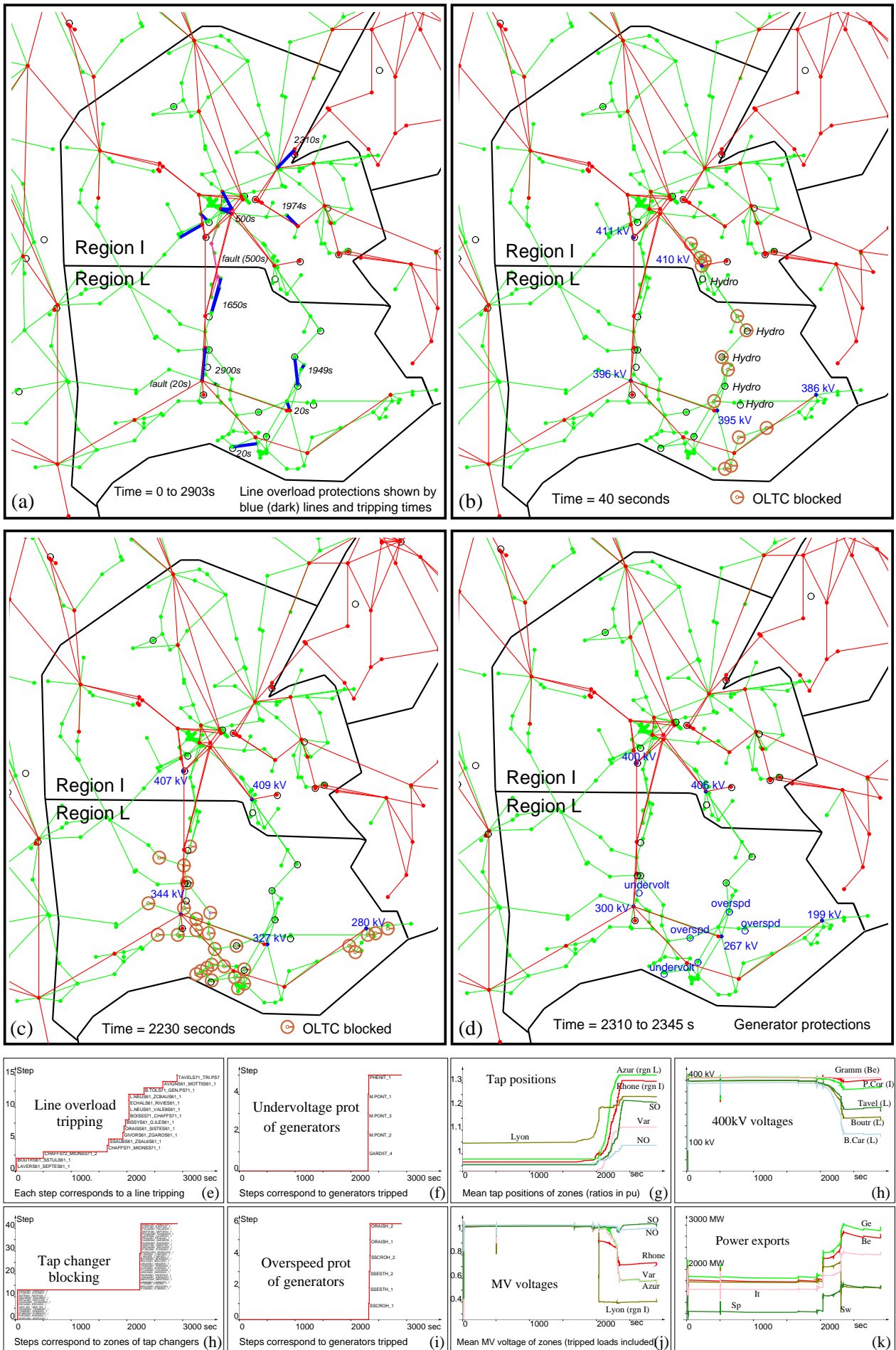


Figure 8. Visionet maps and curves showing various types of events for scenario 1439

blocking devices act within the 225kV subsystem close to the hydro plants (see Fig. 8b,h). Nothing else happens until the occurrence of the second fault, at $t=500s$: it is a three-phase short-circuit on a critical 400kV line in the Rhône valley (see Fig. 8a), leading to the loss of two major 400kV lines (a parallel line, due to overload protection misoperation, and the faulted line due to normal operation of protections).

Intermediate stage ($t=505s$ to $t=2000s$). Given the number of circuits lost, 9 (400kV and 225kV) lines are overloaded in the study region, leading to their successive tripping between $t=1650s$ and $t=2000s$ (see Fig. 8a,e). During the same period, tap changers start reacting, first in region I, then in region L (see Fig. 8g).

Voltage collapse in region L ($t=2000s$ to $t=2904s$). Upon loss of the last three lines the critical point is reached in the Eastern part of region L: tap changers continue raising their taps but load decreases (compare Fig. 8j and 8g). At about the same time, low voltage protections start disconnecting 56 further (mainly 225kV) lines around Lyon leading to the loss of 3100MW of load in region I (see Fig. 8j). We can observe on Fig. 8k how this load decrease is compensated by an increase in the interconnections flows out of France. At $t=2230s$ a second wave of 30 tap changer blocking devices are activated throughout region L, unfortunately too late (see Fig. 8c,g,j). At $t=2310s$ a further 400kV line in the North-Eastern part of region I trips on overload protection (this is actually an interconnection between France and Switzerland, as can be deduced from the sudden decrease in the corresponding power flow in Fig. 8k) and two seconds later some generators in the Southern part trip on undervoltage protections (see Fig. 8d,j). About ten seconds later, some other generators are lost due to overspeed protections (see Fig. 8d,i); two further lines are tripped on overload at $t=2650s$ and $t=2900s$, but since the simulation is stopped shortly afterwards we can't observe further consequences. At $t=2904s$ the system stabilizes with very low voltages throughout region L, normal ones in region I (see Fig. 8h).

To summarize, this scenario corresponds to the loss of 9 400kV and 65 225kV lines, 5 thermal units, 6 hydro units, 3100MW of load in region I and 4700MW in region L; increase in total exportation of 3600MW; very low voltages in region L (153kV to 266kV) and normal voltages in region I (408kV). While the dynamic behavior of power systems may be extremely complex, our illustration shows how visualization tools can make the engineer's life easier, if not easy.

5 CONCLUSION

The aim of this paper was to present tools and techniques which may help engineers to study (and hence improve) the dynamic performance of their power system in a systematic way. More and more possibilities become available to generate and collect data about power system dynamics, be it from simulations or from historical recordings. The data mining and visualization tools discussed in this paper provide a multitude of possibilities to the engineers in charge of power system planning, maintenance and operation. Some of these tools are generic in nature and could be provided by any commercially available data mining tool; however, much more possibilities are offered by designing special purpose tools, making use of physical information about the power system structure and presenting information in a way compatible with the approach experts use

to analyse system behavior.

Our illustrations have been drawn from the power system dynamic security assessment area, which is certainly one of the fields where needs tend to become more and more stringent. However, there are many other problems in power systems where data mining may be very useful. In particular, we mention the broad field of modeling (i.e. making use of measured data in order to develop better models), state-estimation and forecasting (building up and maintaining statistical models about load patterns, measurement and signaling errors. . .), and the very broad area of Monte-Carlo based probabilistic tools used in many long term and short-term planning problems (to find out which parameters influence most significantly loss-of-load-probabilities, costs. . .).

Further work will aim at improving the flexibility and graphical features of the proposed tool so as to be used for other data mining problems, and building interfaces in order to couple it also with dynamic simulation packages.

ACKNOWLEDGMENTS

This research was partly carried out in the context of a collaboration with Electricité de France. We thank the engineers of Electricité de France for the valuable suggestions and feedback.

REFERENCES

- [1] U.M. Fayyad, G. Piatesky-Shapiro, P. Smyth, and R. Uthurusamy. *Advances in Knowledge Discovery and Data Mining*. AAAI Press/MIT Press, 1996.
- [2] L. Wehenkel. *Automatic learning techniques in power systems*. Kluwer Academic, Boston, 1998.
- [3] L. Wehenkel, C. Lebrevelec, M. Trotignon, and J. Batut. A probabilistic approach to the design of power systems protection schemes against blackouts. In *Proc. IFAC-CIGRE Symp. on Cont. of Power Plants and Power Syst.*, pp. 506–511, Beijing, 1997.
- [4] Y. Jacquemart, L. Wehenkel, and P. Pruvot. Practical contribution of a statistical methodology to voltage security criteria determination. In *Proc. of the 12th PSCC*, pp. 903–910, Aug. 1996.
- [5] T. E. Dy-Liacco. Enhancing power system security control. *IEEE Computer Applications in Power* **10**, no. 3, pp. 38–41, July 1997.
- [6] P. M. Mahadev and R. D. Christie. Envisioning power system data: concepts and a prototype system state representation. *IEEE Trans. on Power Syst.*, 1993.
- [7] F. L. Alvarado, Y. Hu, C. Rinzing, and R. Adapa. Visualization of spatially differentiated security margins. In *Proc. of the 11th PSCC*, pp. 519–525, Aug-Sept 1993.
- [8] P. M. Mahadev and R. D. Christie. Envisioning power system data: vulnerability and severity representations for static security assessment. *IEEE Trans. on Pwr Syst.*, 1995.
- [9] L. Wehenkel, M. Pavella, E. Euxibie, and B. Heilbronn. Decision tree based transient stability method - a case study. *IEEE Trans. on Power Syst.* **PWRS-9**, no. 1, pp. 459–469, 1994.
- [10] B. Meyer and M. Stubbe. EUROSTAG, a single tool for power system simulation. *Transmission and Distribution International* **3**, no. 1, pp. 47–52, 1992.

BIOGRAPHIES

Pierre Geurts is currently a student at the electrical engineering department of the University of Liège, nearing completion of his undergraduate studies. He started working on temporal data mining problems about one year ago.

Louis Wehenkel 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 associate of the F.N.R.S. His research interests lie in automatic learning and its application to power system security analysis and control.