FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO
Departamento de Engenharia Electrotécnica e de Computadores

APLICAÇÃO DE TÉCNICAS HÍBRIDAS DE APRENDIZAGEM AUTOMÁTICA PARA AVALIAÇÃO RÁPIDA DE SEGURANÇA DINÂMICA DE REDES ISOLADAS COM PRODUÇÃO EÓLICA

"APPLICATION OF HYBRID AUTOMATIC LEARNING TECHNIQUES FOR FAST DYNAMIC SECURITY ASSESSMENT OF ISOLATED POWER SYSTEMS WITH WIND POWER PRODUCTION"

Maria Helena Osório Pestana de Vasconcelos

Licenciada em Engenharia Electrotécnica e de Computadores – Ramo de Sistemas Eléctricos de Energia – pela Faculdade de Engenharia da Universidade do Porto

Dissertação submetida para satisfação parcial dos requisitos
do programa do curso de
Mestrado em Engenharia Electrotécnica e de Computadores
(área de especialização de Sistemas de Energia)

Porto, Setembro 1999

Dissertação realizada sob a supervisão de

Doutor João Abel Peças Lopes
Professor Auxiliar com Agregação

Departamento de Engenharia Electrotécnico e de Computadores da Faculdade da Universidade do Porto

Dissertação realizada por

Maria Helena Osório Pestana de Vasconcelos

Unidade de Energia (*Power Systems Unit*)
INESC Porto
Telefone: 351-2-2094224 Fax: 2-351-2084172
E-mail: hvasconcelos@inescn.pt
Web page: http://power.inescn.pt/

RESUMO

Nesta dissertação é apresentada uma metodologia onde se aplica uma nova técnica de aprendizagem automática – *as árvores de regressão híbridas* – que explora o conhecimento funcional sobre o comportamento de sistemas. Esta metodologia foi especialmente concebida para exploração do comportamento dinâmico de redes eléctricas isoladas com grande produção eólica. As árvores de regressão híbridas, para além de classificarem de forma rápida a segurança de exploração do sistema, permitem ainda quantificar em tempo real o grau de robustez do sistema através da emulação de índices de segurança contínuos que traduzem o seu comportamento dinâmico em face de algumas perturbações.

É descrita a aplicação da metodologia desenvolvida à rede eléctrica da ilha da Terceira. O objectivo desta aplicação consistiu na obtenção de estruturas de segurança, que realizem avaliação rápida do comportamento dinâmico do sistema atendendo a problemas de instabilidade da frequência. Esta descrição incluí a explicação do procedimento efectuado para a geração do conjunto de dados.

É também descrito como a metodologia foi aplicada ao sistema eléctrico da ilha de Creta, no sentido de desenvolver ferramentas avançadas que permitam apoiar os operadores da rede na gestão da potência eólica instalada.

Neste documento apresenta-se ainda a avaliação de desempenho das estruturas de segurança treinadas, incluindo uma análise comparativa com árvores de decisão e redes neuronais.

O trabalho de investigação que conduziu à elaboração desta dissertação foi realizado no âmbito da fase final do projecto CARE do programa Europeu JOULE/THERMIE. Os trabalhos decorreram na FEUP (Faculdade de Engenharia da Universidade do Porto) e no INESC Porto (Instituto de Engenharia de Sistemas e Computadores do Porto).

Palavras Chave

Redes Eléctricas Isoladas com Produção Eólica
Avaliação Rápida de Segurança Dinâmica
Monitorização de Segurança
Problemas de Estabilidade da Frequência
Técnicas Híbridas de Aprendizagem Automática
Técnicas de Aprendizagem Máquina
Árvores de Regressão
Modelos de Regressão Kernel

ABSTRACT

This thesis presents a methodology that applies a new automatic learning technique – *the hybrid regression trees* –, which exploits the functional knowledge about systems behavior. This methodology was specially conceived to exploit the dynamic behavior of isolated power systems with large wind power production. The hybrid regression trees, besides producing fast security classification of the system, can still quantify in real-time the security degree of the system by emulating continuos security indices that define the power system dynamic behavior.

The application of the developed methodology to the power system of Terceira island is described. The main goal of this procedure was to obtain security structures, which produce fast dynamic security assessment of the system regarding frequency instability problems. A description of the data set generation procedure is included.

This document also describes the application of the methodology to the power system of Crete. The main goal of this procedure was to develop advanced tools that can help operators to perform the management and operation of the installed wind power.

The performance evaluation of the trained security structures, including comparative assessment with decision trees and neural networks, is also presented.

The research leading to this thesis was developed within the framework of the last stage of the EU CARE project of the JOULE/THERMIE program. The work was carried out at FEUP (Faculdade de Engenharia da Universidade do Porto) and at INESC Porto (Instituto de Engenharia de Sistemas e Computadores do Porto).

Keywords

Isolated Power Systems with Wind Power Production
Fast Dynamic Security Assessment
Security Monitoring
Frequency Stability Problems
Hybrid Automatic Learning Techniques
Machine Learning Techniques
Regression Trees
Kernel Regression Models

RESUME

Dans cette thèse on présente une méthodologie où on applique une nouvelle technique d'apprentissage automatique – *des arbres de régression hybrides* – qui exploite connaissance fonctionnelle sûr le comportement des systèmes électriques. Cette méthodologie a été spécialement conçu pour aider à l'exploitation du comportement dynamique des réseaux isolés qui ont une grande production éolienne. Les arbres de régression hybrides permettent de classifier de forme rapide la sécurité d'exploitation du système, permettant aussi quantifier en temps réel le dégrée de robustesse du système à travers de l'émulation des indices de sécurité d'une façon continue en traduisant le comportement dynamique du réseau en face de quelques perturbations.

On décrit l'application de cette méthodologie au réseau électrique de l'île Terceira aux Açores. Comme résultat de cette application on a obtenu des structures de sécurité, qui permettent de évaluer de forme très rapide le comportement dynamique par rapport à des problèmes d'instabilité de fréquence. Cette application inclus le développement de la procédure effectué pour faire la génération de l'ensembles de données.

On décrit aussi la façon comme cette méthodologie a été appliqué au réseau électrique de l'île de Crète, en tenant en considération le fais que les structures développés se destinait à être intégrées dans un système de aide à la gestion du réseau, nommément dans ce qui concerne la gestion de la production éolienne.

Dans ce document on présente aussi l'évaluation de la performance de ces structures de sécurité, de une façon absolue et en les comparant aussi avec la performance obtenu avec des arbres de décision et des réseaux neuroneaux.

Ce travaille de recherche a été développé dans le cadre d'un programme de recherche Européen JOULE/THERMIE. Les travaux on été développes à la Faculté des Ingénieurs de l'Université de Porto (FEUP) et à l'INESC Porto.

Mots Clés

Réseaux Electriques Isolés avec Production Eolienne
Evaluation Rapide de Sécurité Dynamique
Monitorization de Sécurité
Problèmes de Stabilité de Fréquence
Techniques Hybrides d'Apprentissage Automatique
Des Arbres de Régression
Modèles de Régression Kernel

AGRADECIMENTOS

Em primeiro lugar, queria agradecer ao meu orientador, Professor Doutor João Abel Peças Lopes, pelo acompanhamento, colaborações prestadas e confiança que depositou em mim durante a realização do presente trabalho. Desejo também agradecer todo o seu apoio, o que certamente contribuiu para o meu entusiasmo pela execução dos trabalhos.

Gostaria ainda de expressar a minha gratidão a todos as pessoas que directa ou indirectamente contribuíram para a realização deste trabalho. Agradeço aos meus colegas da Unidade de Energia do INESC Porto, nomeadamente pelos esclarecimentos de programação prestados bem como pela troca de ideias que tivemos sobre o presente trabalho. Gostaria ainda de mencionar a contribuição dos elementos do consórcio internacional do projecto Europeu CARE, nomeadamente atendendo a aspectos relacionados com a envolvente conceptual e de implementação do projecto.

Por último queria agradecer a toda a minha família, e em especial ao meu marido pelo apoio e ajudas prestadas em questões de programação que foram muito apreciadas, bem como pela paciência e empenho com que leu uma versão desta dissertação.

Esta dissertação só foi possível ser realizada graças à FCT – Fundação para a Ciência e Tecnologia –, que me concedeu uma Bolsa de Mestrado (PRAXIS/BM/17742/98) no âmbito do Programa PRAXIS XXI, e ao INESC Porto – Instituto de Engenharia de Sistemas e Computadores do Porto – devido aos meios e suporte financeiro prestados.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

Below is provided a list of the abbreviations most used in this document.

WG:	Wind Generator
SCADA:	Supervisory Control and Data Acquisition
SA:	Security Assessment
DSA:	Dynamic Security Assessment
SM:	Security Monitoring
SR:	Spinning Reserve
WP:	Wind Penetration
WM:	Wind Margin
ED:	Economic Dispatch
UC:	Unit Commitment
OP:	Operating Point
AL:	Automatic Learning
PR:	Pattern Recognition (a class of AL techniques)
ANN:	Artificial Neural Network (a class of AL techniques)
ML:	Machine Learning (a class of AL techniques concerned with automatic design of interpretable symbolic rules)
DT:	Decision Tree (a ML method)
RT:	Regression Tree (a ML method)
KNN:	K-Nearest Neighbors rule (a PR method)
KRT:	Kernel Regression Tree (an hybrid approach of AL techniques)
HRT:	Hybrid Regression Tree
DS:	Data Set
LS:	Learning Set
TS:	Testing Set

1 Introduction

This thesis reports the research work related to the application of a new hybrid Automatic Learning (AL) technique - the Hybrid Regression Trees (HRTs)- in the field of Dynamic Security Assessment (DSA) and Security Monitoring (SM) of isolated power systems with high penetration of wind power. The research was developed within the framework of an EU project of the JOULE/THERMIE program.

This new hybrid approach was implemented to make, for the first time, fast dynamic security assessment (DSA) of power system in the field of frequency stability problems. The results of the application of this algorithm to Crete can be found in [1] and [2].

The HRT method was presented by Luís Torgo [3] in 1997, integrating Regression Trees (RTs) ([4] – Breiman et al., CART, 1984) with kernel regression models ([5] – Watson, 1964; [6] – Nadaraya, 1964). The first application of the RT approach in DSA, used in the field of voltage stability problems, is due to Wehenkel [42], in 1995. Recently, an application of a HRT approach in the same security assessment problem was presented in [43] by Peças Lopes et al..

Another, not less important, goal of the research work reported in this document was to evaluate the performance of the applied HRT technique, by comparing results with other existent automatic learning techniques, namely Decision Trees (DTs) and Artificial Neural Networks (ANNs). These approaches were tested on the electrical power systems of the Terceira island (in Azores - Portugal) and of the Crete island (in Greece).

For the Terceira case study, a scenario of the year 1999 was considered. This electrical network consists on an isolated Diesel-hydro power system, where a peak load of approximately 20 MW was considered. Two wind parks, with a total installed power of 4.8 MW, are foreseen to be operating in Terceira by the year 1999. This power system is properly explained in Chapter 4 (Section 4.2.1).

For the Crete case study, it was considered a scenario of the year 2000. This electrical network consists on an isolated power system, which production is based on the exploration of several types of conventional units, namely steam and Diesel units, gas turbines and 1 combined cycle plant. A peak load of approximately 360 MW was considered for the year 2000. A total of 10 wind parks, consisting on 162 wind turbines with an installed capacity of more than 80 MW, are or will be installed (have been approved) in Crete by the year 2000. A more detailed description of this system is presented in Chapter 4 (Section 4.3.1).

As it can be seen from the two previous power systems, the framework of this Master thesis concerns with medium size and large isolated power systems with high penetration of wind power production. In isolated power systems, the electric power is usually produced by conventional fuel units, which results in high costs of electricity due to fuel cost itself and costs of transportation. Therefore, in these systems, the production of electric energy from wind presents particular interest, especially when important wind energy potential exists, which is usual in many islands. In fact, significant replacement of conventional fuels can be obtained by a high wind power penetration. In this case however, it is important to ensure that the power system operation will not be adversely affected by an increased penetration of this volatile form of energy.

Fast wind power changes and very high wind speeds resulting in sudden loss of wind power production, can cause large frequency excursions and dynamically unstable situations. Moreover, frequency and voltage oscillations might easily trigger the protection relays of the wind parks, causing unbalance in the system generation/load. The resulting frequency oscillations might lead to load shedding, or even to the system collapse.

In order to guard the power systems against foreseen disturbances, the system operators tend to adopt a conservative policy for unit scheduling operation and generation dispatch, leading to under exploitation of the installed wind power and to uneconomic operating points. To maximize wind power penetration without compromising the system security, on-line dynamic security assessment functions can prove to be very valuable. The integration of these DSA functions in a proper advanced control system can help the operators taking decisions, by suggesting optimal unit operating schedules and dispatch, both from economic and security point of view, as well as by providing security monitoring. As such functions require on-line performances, the evaluation of the system dynamic security obtained by means of conventional analytical tools of dynamic simulation would create unfeasible high computational times. This fact leads to the application of AL techniques to deal in a proper way with this problem.

Within the framework of an European R&D project of the JOULE II program (contract JOU2-CT92-0053) [7], such functions have been developed and are integrated within an advanced control system tailored to the needs of small isolated power systems with increased wind power penetration. A pilot control system has been installed on the Greek island of Lemnos – an isolated Diesel-wind system with a peak load of approximately 10 MW. In this system, dynamic security assessment is taken care by two modules based on Artificial Neural Networks and Decision Trees [40][41].

Within the framework of an European R&D project of the JOULE/THERMIE program (contract JOR3-CT96-0119), this control system is currently being extended into the CARE system, to cover the needs of large isolated systems with high wind power penetration. The CARE system is an advanced control system that aims to achieve optimal utilization of renewable energy sources, in a wide variety of medium and large size isolated systems with diverse structures and

operating conditions [8]. During the present year, a pilot installation of this system is being installed on the energy management center of Crete island. In this system, dynamic security assessment is performed by three modules based on Artificial Neural Networks, Decision Trees and Hybrid Regression Trees, being this later approach implemented within the research work of this Master thesis.

This document is organized as follows. Chapter 2 addresses some problems related to the operation of medium size or large isolated power systems with a high penetration of wind power production. Having in mind these problems, the main functions of the CARE advanced control system are synthetically described. Within the description of this technical solution, some of the concepts related to the application of automatic learning techniques to perform dynamic security assessment in power systems are explained at an intuitive level.

Chapter 3 gives an overview of the concepts related to the application of AL techniques to make DSA in power systems. First, a short review to the definition of DSA is provided. Then, a historical perspective of the application of AL techniques is presented. In this Chapter, the problem formulation that is behind the application of AL techniques to perform DSA in power systems is also presented. The functions provided by the AL security structures are also addressed, highlighting their advantages regarding the ones provided by the classical analytical models. In this Chapter, an overview of the security structure provided by two existent AL methods – Regression Tree (RT) and K-Nearest Neighbors (KNN) rule – is provided. After the construction of a security structure it is mandatory to evaluate its generalization capabilities by estimating its accuracy. Therefore, the most applied procedures to estimate accuracy of AL security structures – the testing error estimation and learning error estimation – are also described in this Chapter. The overfitting problem is also addressed, where an illustration of this problem is demonstrated in security structures extracted by applying the implemented Hybrid Regression Tree method. Finally, a particular important issue described is the methodology developed in this work to apply AL techniques in the field of DSA.

In Chapter 4, a technical description of the procedure developed to generate a data set for Terceira electrical network, is provided. This task was mandatory to be performed in order to extract AL security structures for this power system. This procedure involved the development of a software tool, which provides a general methodology to generate data sets for Diesel-wind isolated power systems. This software tool was developed in partnership with M. A. Mitchell (researcher of the Power Systems Unit of INESC Porto), within the framework of his Master thesis requirements. For the Crete case study, researchers of the NTUA (National Technical University of Athens – Greece) provided the data set required to apply AL techniques. A synthetic description of the procedure developed to generate this data set is also presented in Chapter 4.

The implemented Hybrid Regression Tree algorithm is described in Chapter 5. Regarding the application of AL techniques in the Terceira and Crete power systems, the obtained results with

the implemented HRT algorithm are presented in Chapter 6. In this Chapter, the available ANN and DT results are also presented for comparison purposes. Researchers of the NTUA provided the DT results, whereas other researcher of the Power Systems Unit of INESC Porto provided the ANN results. The security structures of the Crete case study are presently being integrated into the CARE control system, which will be in operation in Crete during this year. Chapter 6 ends with the conclusions achieved from analyzing the obtained results, which includes a comparative assessment between the performance of the three applied methods. The general conclusions obtained from the work reported in this document are presented in Chapter 7.

The work reported in this Master thesis was developed within the framework of the European R&D JOULE/THERMIE (JOR3-CT96-0119) project "CARE – Advanced Control Advice for Power Systems with Large-Scale Integration of Renewable Energy Sources". The project began in February 1997 and concluded in July 1999. It involved several R&D institutions: NTUA [Gr] – prime contractor, INESC Porto [Pt], RAL [UK], ARMINES [Fr], UMIST [UK], AUTH [Gr], TUC [Gr]; industrial partners: EFACEC [Pt]; and utilities: PPC [Gr], EDA [Pt].

2 Management and Operation of Isolated Systems with Large Wind Power Production

2.1 Operation of Isolated Systems with Large Wind Power Production

In isolated systems, conventional production is mainly provided by Diesel units, and therefore, with high cost of electricity production [9]. As an example, the islands of Cape Verde Republic can be referred, where the electricity production costs impose restrictions to the country development. In fact, this country imports Diesel fuel for electricity production and for production of fresh water, which is done through desalination of sea water [10].

By increasing renewable power penetration in these systems, significant replacement of conventional fuels consumption can be obtained, and therefore, the power production cost can be considerably reduced. Moreover, in islands, usually there are particularly favorable meteorological conditions for wind power sources exploitation. For these reasons, in the last years a large amount of wind power sources have been installed in islands. Examples of recent (or on going) installation of large penetrations of wind power in islands can be found in Greece (Lemnos island [7], Crete island [11]), in Portugal (Madeira island [12][13]) and in Cape Verde Republic (islands of S. Vincente [15], Santiago [16] and Sal [17]). From Figure 2.1 to Figure 2.4, the foreseen installed power scenarios of the different types of electrical power sources that are presently under exploitation (or are expected to be explored) on those systems are presented.

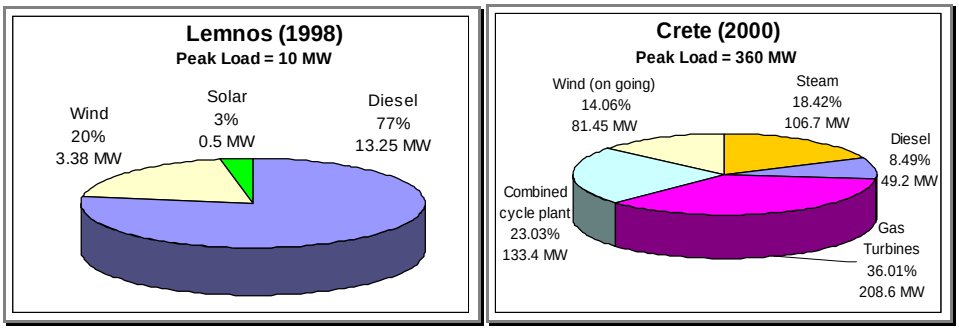


Figure 2.1 – Foreseen installed power scenarios for the Lemnos and Crete islands - Greece

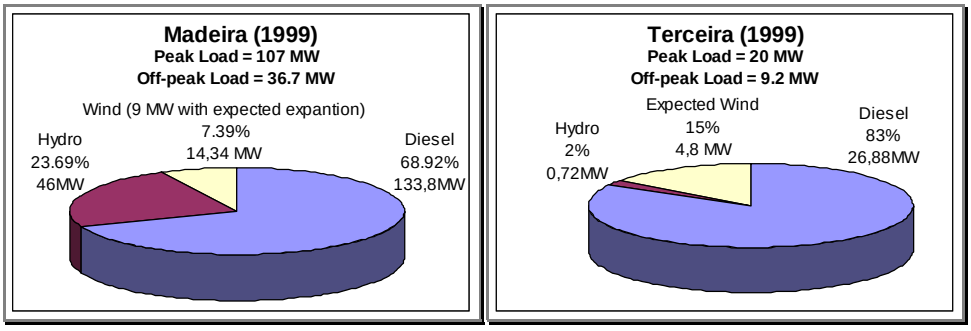


Figure 2.2 – Foreseen installed power scenarios for the Madeira and Terceira islands - Portugal

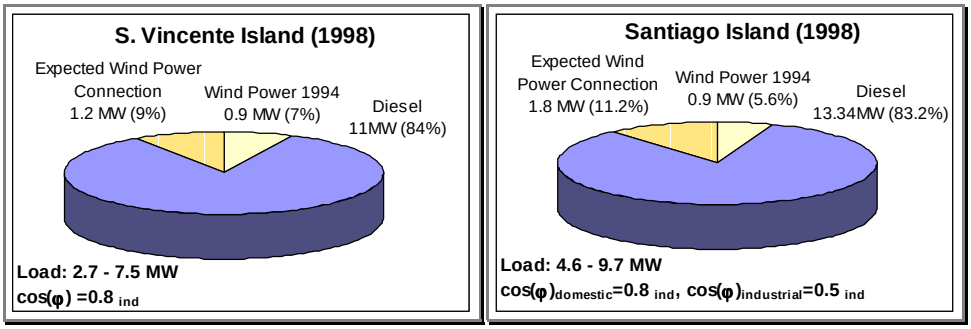


Figure 2.3 – Foreseen installed power scenarios for the S. Vincente and Santiago islands – Cape Verde Republic

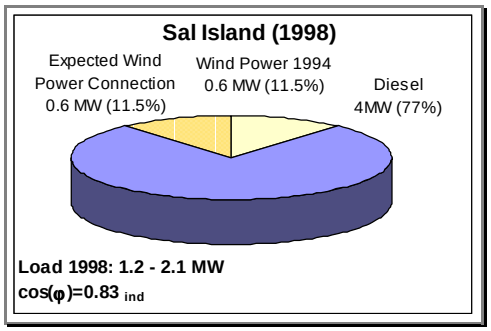


Figure 2.4 – Foreseen installed power scenario for the Sal island – Cape Verde Republic

In these systems it is important to ensure that the electric power system operation will not be adversely affected by an increased integration of this volatile form of energy. In fact, due to the uncertainty and uncontrolled characteristic[1] of wind power production, and due to the weakness of isolated power systems, the integration of large penetration of this kind of production in isolated systems might introduce dynamic behavior problems.

When compared to interconnected power systems, isolated systems, although having the same general operating rules, should be operated under some additional constraints. Since there is no available help from neighboring systems, the grid is weaker and therefore greater concerns exist related to system security, control of frequency and management of system generation reserve.

A common aspect to all these problems is the requirement to ensure that sufficient reserve capacity exists within the system to compensate sudden loss of generation. In fact, mismatches in generation and load and/or unstable system frequency control might lead to system failures. This type of instability is called frequency instability and depends on the ability of the system to restore balance between generation and load following a severe system disturbance with minimum loss of load [18]. Generally, frequency instability problems are associated with inadequacies in equipment responses, poor coordination of control and protection equipment or insufficient generation reserve [1]. Thus, in isolated systems there is also a great concern in

[1] Wind power production level is uncertain and not controlled by operators because it depends on wind climatic conditions, which presents random nature. If wind parks belong to public owners, as a control action to maintain system security, the operators can disconnect wind generators. On the other hand, if wind parks belong to private owners, these power sources are non-dispatchable, and therefore, the operation policy of disconnecting generators may be more difficult to apply.

operating them with robust conventional thermal generators (i.e., with proper inertia constant values and proper response of automatic voltage regulators (AVRs) as well as speed regulators).

Having in mind these problems and the additional difficulties caused by the introduction of a high penetration from wind energy, in order to guarantee service quality and continuity of operation in the sequence of the integration of the wind power, new operating rules have to be adopted. In the common practice a large number of operating scenarios are screened off-line, by running analytical tools of load-flow and dynamic simulation, from where operating guidelines including corrective measures can be prepared. Whenever there is a significant change in the system, these studies are repeated and new operating guidelines should be prepared [42]. These new operating guidelines usually increase the complexity of the system operation and management and, namely, might create generation dispatching and scheduling difficulties [9].

In very small isolated power systems, where the maximum demand does not exceed 1 MW, the existing Diesel generators are small units, with short starting times. On the other hand, for medium size (maximum demand up to 30 MW), the starting time of Diesel units is in the scale of 10 up to 30 min, or even more. Thus, in this systems optimal dispatching depends largely on the judgements made by operators [7]. For this reason, in medium-size or large power systems with high penetration of wind power, if operators do not have a proper control system to help them taking decisions, they will tend to operate it in a conservative way, leading to uneconomic operating points. This issue is particularly important in isolated systems, because it will increase the already existent high electricity costs.

In the next two sections, the dynamic behavior problems and generation dispatching and scheduling difficulties, introduced by high levels of wind power production in isolated systems, are addressed in a more detailed way.

2.1.1 Dynamic Behavior Problems Introduced by Wind Generators

Dynamic behavior problems can occur from the introduction of a high penetration from wind energy in isolated systems. Fast wind power changes and very high wind speeds that result in sudden loss of wind generator (WG) production, can cause large frequency and voltage variations, or even high rate of frequency changes. These oscillations, besides resulting in the loss of service quality, might lead to partial service interruption, or even to system instability and to the system collapse. In fact, they might trigger the operation of system protection devices, performing the disconnection of generation units or load shedding, and thus increasing the adversity of the disturbances.

The disturbances introduced by the WGs, which might originate these dynamic behavior problems, can have origin in the random nature of wind power source and/or from the technical characteristics of the energy conversion system. Usually, the main types of disturbances introduced by WGs are the following:

- Electrical power outputs variations of WG, due to wind speed variation (turbulence in wind speed or/and sudden and large wind speed changes).
- Disconnection of WGs from the network, due to variations in wind speed that leads to violation of cut-off or cut-in speeds. One example of these operating limits can be seen in Figure 2.5 where the power curve of a WG NORDTANK 300/31 [300 kW, 400 V] with stall power regulation is presented.
- Disconnection of asynchronous WG after the occurrence of a short-circuit in the grid, due to the generator not being able to re-excite after fault elimination;
- Disconnection of synchronous/asynchronous WG after the occurrence of a short-circuit in the grid, due to the activation of protection devices.

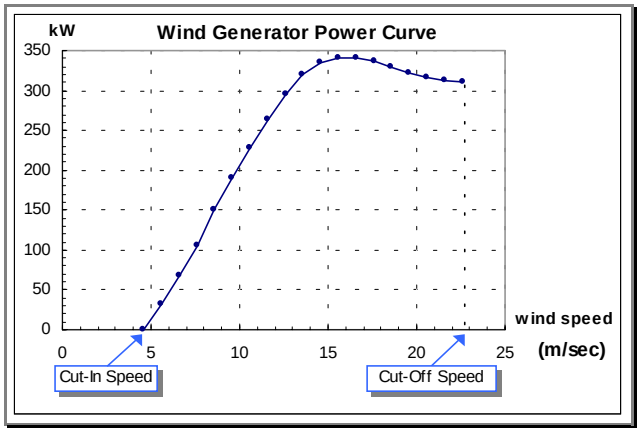


Figure 2.5 – Power curve of WG NORDTANK 300/31 [300 kW, 400 V]

Usually, these disturbances are particularly critical during light load hours, because in those periods wind power penetration can be considerably important. Although during light load hours the wind disturbances typically cause the higher frequency and voltage deviations, in some systems, less damped oscillations are more likely to occur during peak hours.

Sometimes, when the system suffers a sudden loss of wind power, to avoid its collapse, operators might have to shed non-priority loads. However, if operators would have the proper tools to help them, they could avoid unnecessary load shedding actions, or even avoid shedding any load.

2.1.2 Generation Dispatching and Scheduling Difficulties due to the Integration of Wind Generators

In order to ensure the system security, conventional units must fast and efficiently compensate the mentioned wind power changes. To help operators exploiting the system in a secure way, usually more efficient rules have to be found for the generation dispatching and scheduling policy. These rules can include the connection of a particular more robust machine, an increased minimum requirements of static conventional generation reserve (to accommodate load if wind power becomes unavailable), and an increased minimum requirements of spinning reserve (to compensate wind power variations).

Usually, the local utilities adopt a very conservative policy of generation dispatch and scheduling. This policy leads to unnecessary large spinning reserve (SR) requirements, to under exploitation of wind power production, and therefore, to uneconomic operating points. One example of a minimum SR requirement criterion, which is usually applied in Diesel-wind isolated powers, is the following:

$$SR \geq \alpha \times P_D + \lambda \times P_W - P_{Sh} \qquad (2.1)$$

where:

P_D – Load demand;
α – Load margin;
P_W – Wind power production;
λ – Wind margin;
P_{Sh} – Amount of load that can be shed when frequency decreases dramatically.

The α and λ factors are perceptual values that express the SR that the system must have, in order to accommodate a load increase of $\alpha \times P_D$ and a wind generation decrease of $\lambda \times P_W$. Obviously, if the uncertainty of wind production profile and load demand profile increases, then α and λ factors must have higher values in order to guarantee a high level of security in the system.

When α and λ factors are too high, in order to guarantee the SR requirements, a large number of Diesel machines have to be in operation. In this situation, Diesel machines might have to operate at low load levels and therefore with low efficiency (to understand the Diesel generator load level/efficiency relationship see Figure 2.6, where a typical curve of the fuel consumption *versus* load level for this machines is presented). Moreover, when Diesel generators operate under minimum technical limits, they need to consume Gas Fuel Oil (GFO), which leads to higher costs of operation.

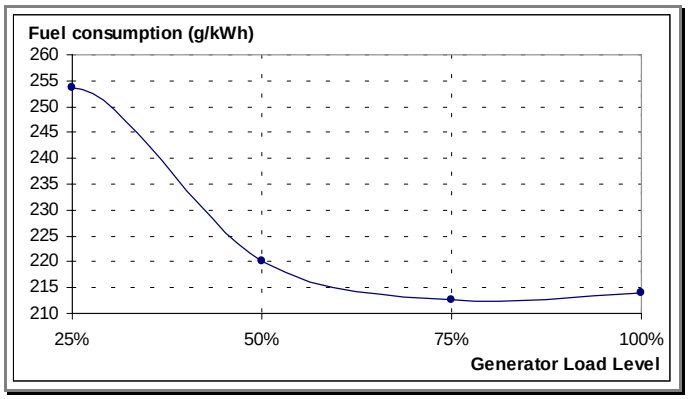


Figure 2.6 – Fuel consumption curve for Diesel generator Deutz [2.3 MW, 6.3 kV]

One example of an extremely conservative policy of dispatch and scheduling, is when operators try to guarantee a SR enough to accommodate the disconnection of a full wind park, and therefore have to consider a very high value for the λ factor.

As the uncertainty of load and wind changes imposes large SR levels, if proper prediction of load consumption and wind power would be available, then the load margin and wind margin could be decreased, providing that the SR requirements could be minimized and the wind penetration maximized.

When wind penetration reaches too high values, a measure that is sometimes used by utilities in order to follow the SR requirements consists on the disconnection of WGs, which might not be the best solution.

2.2 Technical Solutions Recommended: an Advanced Control System

Global control systems are the most suitable tools to use, in order to optimize the management and operation of isolated systems where a large amount of wind power production needs to be accommodated [9].

These control systems can help the power system operators taking decisions by performing the following functions:

Function 1:
Suggesting on-line optimal scenarios, both from economic and security point of view, by providing:

- the start and stop schedule of conventional generating units and WGs for the next planning horizon (usually for the upcoming hours with a time step of minutes);
- the load level for the conventional generating units selected to be in operation for the upcoming minutes.

Function 2:
Providing security monitoring.

The suggested scenarios are the ones that minimize production costs and maximize wind power penetration, without compromising the security of the system. To accomplish these requirements, the control systems must include functions of:

- wind power forecast;
- load forecast;
- economic operation, including unit commitment and dispatching functions;
- and dynamic security assessment.

These functions have to be integrated with a SCADA system, a data-base, and a man-machine interface.

Due to special characteristics of the operation of medium-sized or large isolated systems with large amount of wind power production, coordination between unit commitment (UC) and economic dispatching (ED) must be different than in the case of interconnected systems.

In interconnected power systems, UC is usually performed off-line, typically with a horizon of a week, with hourly time-steps. This gives the basis for performing the ED every 10 or 15 minutes, most of the time including also reactive power dispatch and perhaps security constraints related to major contingencies. In small isolated power system, on the other hand, a simple unit scheduling is usually necessary, due to the simplicity of the system, even when renewable power sources are present [19]. However, in medium-sized or large isolated power systems with high penetration of wind power sources, a different approach is necessary. In fact, the wind power production, whose generated power must be forecasted and has some degree of uncertainty, has a strong influence in the dynamic security and economy of dispatch and generation schedule. Thus, besides load forecast, the suggested units scheduling and generation dispatch must consider wind power forecast, and therefore UC can no longer be performed off-line. Economic operation must be divided in a unit commitment module and a dispatch module that are performed in sequence, with an optional intermediate decision step that allows the operator to take into account information automatically produced by a security assessment module.

Since January of 1995, a prototype of such control system is running in the Greek island of Lemnos (an isolated Diesel-wind system with a peak load of approximately 10 MW – see Figure 2.1), within the framework of an European R&D project of the JOULE II program [7]. Currently, within the framework of a European R&D project of the JOULE/THERMIE program [8], this control system is currently being extended into the CARE system to cover the needs of large isolated systems with high wind power penetration. By this year, the CARE system will be in operation in the control center of Crete, the largest Greek island (an isolated conventional-wind power system with a peak load of approximately 360 MW – see Figure 2.1).

The general architecture of such a control system is presented in Figure 2.7. In the remainder of this Section, a synthetic description of the modules presented in Figure 2.7 is performed, where emphasis is made to the dynamic security assessment functions. A complete description of the Lemnos and Crete control centers and its modules can be found in [7] and [8].

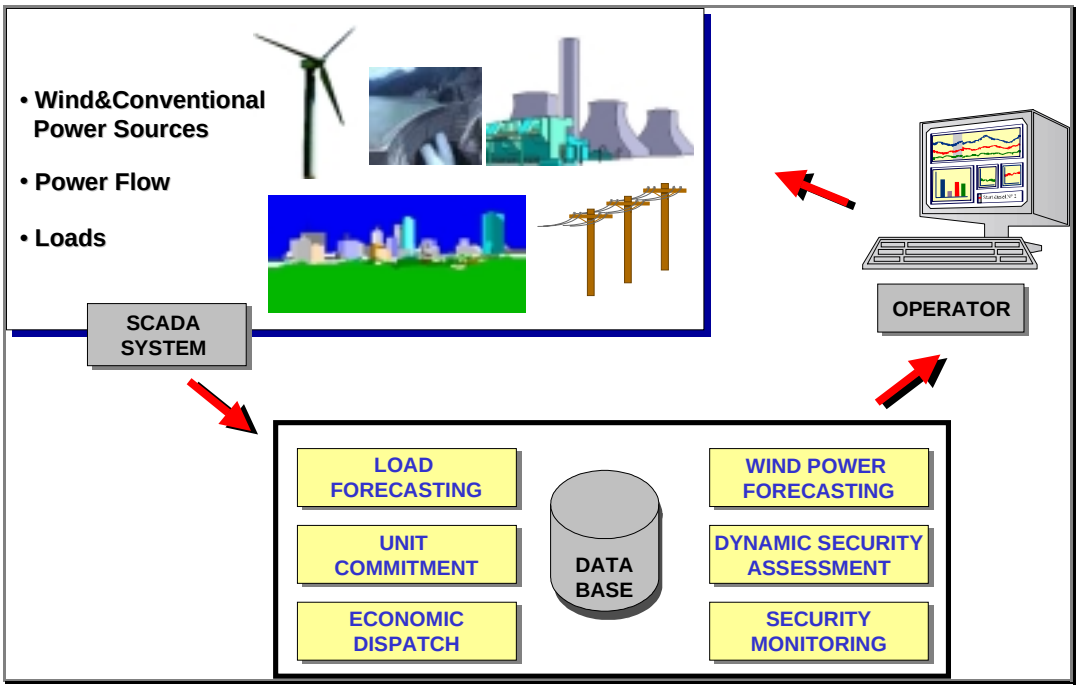


Figure 2.7 – Advanced control system architecture

The *Load Forecasting* and *Wind Power Forecasting modules* perform the prediction of load and wind power sources profiles for the upcoming planning horizon, with a time step at least equal to the one used by the unit commitment module. To make prediction, these modules need measures from the network, provided by the SCADA system, related to load consumption and wind power production.

The start and stop schedule of all the generating units for the next planning horizon are determined by means of an on-line *Unit Commitment (UC) module*. This module aims at to optimize the operating costs (fuel costs, start-up and shut-down costs) taking into consideration the operating constraints of the power system, namely:

– technical constraints of the Diesel units (generation limits, start-up and shut-down times, etc);
– and the prediction of load and wind power profiles for the next planning horizon.

The load level of the conventional generating units, for the upcoming minutes, is determined by means of an *Economic Dispatch (ED) module*. Like the UC module, this module aims at to optimize the operating costs under the operating constraints of the power system.

The control system of Crete island was projected to provide, every 20 minutes, a UC solution for the next 8 hours (with a time step of 20 minutes), and a ED solution for the upcoming 20 minutes. In the prototype installed in the control center of Lemnos, every 10 minutes, presents a UC solution for the next 2 hours (with a time step of 10 minutes), and a ED solution for the upcoming 10 minutes.

2.2.1 Dynamic Security Assessment Functions

Due to the difficulties mentioned before, in order to guarantee the dynamic security of the system a *Dynamic Security Assessment (DSA) module* must be also available in the control center. This module has the task to guarantee that the predispatch solutions, produced by the UC module, and the dispatch solutions, produced by the ED module, lead to dynamically "secure" operating points (OPs), for all of a set of pre-defined disturbances. To accomplish this, if the DSA module identifies that the dispatch policy leads to an "insecure" OP, then the solution is rejected and a request for a new solution is made.

Another interesting security function to be available is to have a real-time evaluation of the system dynamic security for the current OP. In the developed control systems this task is performed by a *Security Monitoring (SM) module*. The security monitoring task might be important during the situation where the predictions of load and wind power, used to produce the UC and ED solutions, are less accurate. In fact, in spite of the UC and ED solutions being evaluated by the DSA module, there is always a risk for the system to reduce security levels. In order that the SM module be able to provide security evaluations, the parameters that characterize the current OP must be available in the control center database.

In the designed control centers, security evaluation functions can be activated/deactivated "on-call" by the operator, namely security monitoring. The operator might also want to guarantee security, just for some of the disturbances. To accomplish this, the man-machine interface has a window, like the one presented in Figure 2.8, where the operator can select the disturbances he wants to consider for the DSA and SM modules. For instance, in the example presented on Figure 2.8, the operator wants to guard security for the disturbances "machine loss" and "wind variation".

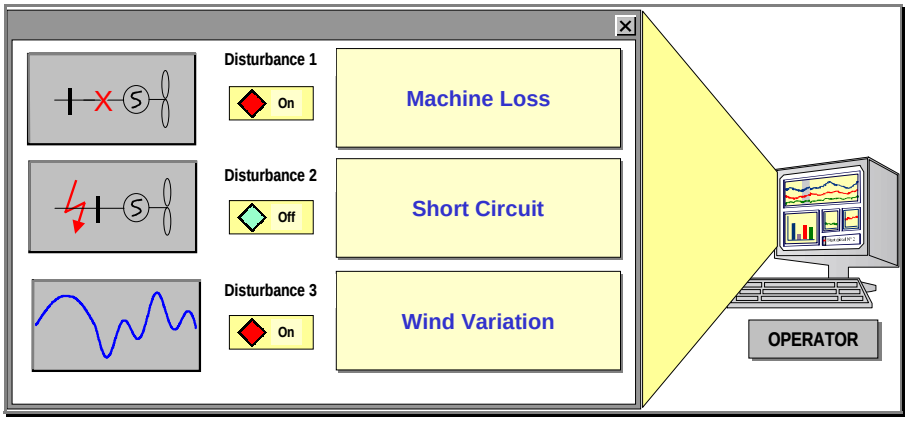


Figure 2.8 – Window for activation of disturbances for the DSA and SM modules

2.2.1.1 Security Classification

In both DSA and SM modules, one OP can be classified as "secure/insecure" according to a pre-defined *security criterion*. As already referred in Section 2.1.1, in isolated systems with wind power production, the dynamic problems that are usual to occur are transient and frequency

stability problems. Thus, a usual practice is to classify an OP as "insecure" if it is expected that a disturbance provokes a large frequency excursions (Δf), or a large frequency rate (df/dt), or even the loss of synchronism.

To perform this classification, it is necessary to define the *security boundaries*, i.e., the numerical values of Δf or df/dt from which the OP is considered to be "insecure". These values generally are the ones that trigger the operation of the system protection devices.

2.2.1.2 Evaluation of Security Degree

Besides classifying the OPs as "secure/insecure", it is also interesting to obtain the *degree of security* (also called *security robustness* or *security margin*), or in other words, obtain knowledge of the "distance" to the security boundaries. This task can help the operators in defining more efficient preventive control actions.

Security degree is provided by means of numerical continuous *security indices* that measure the expected system dynamic behavior for each one of the pre-defined disturbances. Examples of possible security indices are: maximum and minimum values reached by transient frequency (f_{min} and f_{max}) or maximum value reached by the rate of frequency change (df/dt_{max}).

For instance, special windows might be available to present to the operator the expected evolution of the security indices for the proposed UC schedule, for the upcoming planning horizon. An example of such a window is presented in Figure 2.9, where the security index used is the minimum value reached by the power system frequency (f_{min}). This window displays, for two pre-defined disturbances, the expected minimum frequency for each time-step of the UC schedule. To provide visualization of the system security margin, there exists a light gray zone that highlights the acceptable values for frequency (which is limited by the security boundaries). The load profile forecast is also visualized. Putting together all this curves, the operator can get, in a fast way, an overview of the periods of the day when security issues are more critical [8].

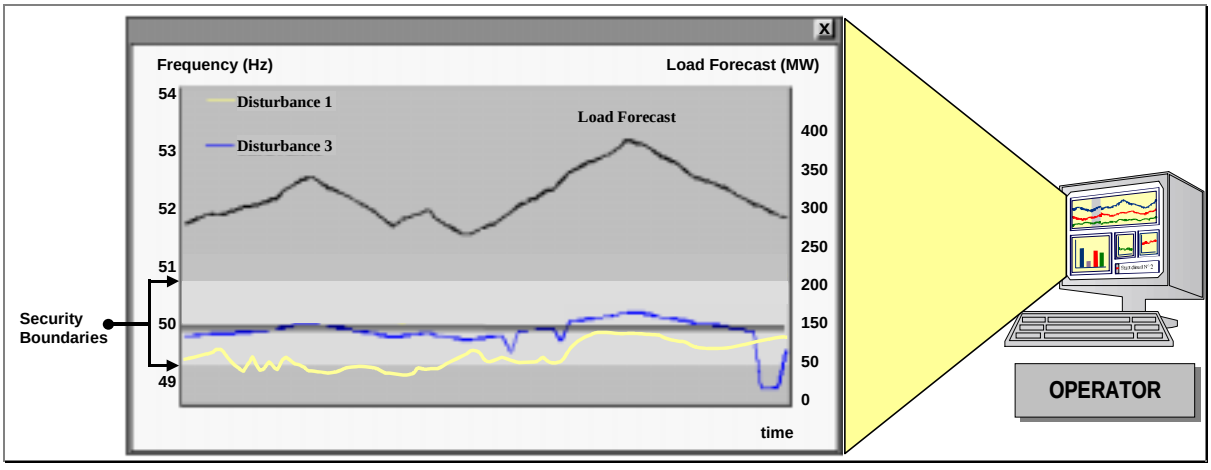


Figure 2.9 – Window with DSA profiles for the proposed UC schedule

2.2.2 Application of AL Techniques to Perform On-Line DSA of Power Systems

The *on-line* requirements for the unit-commitment and economic dispatch functions, also as the r*eal-time* requirements for the security monitoring function, impose tied execution time constraints that must be performed, namely by the DSA and SM modules.

For this reason, the evaluation of the system dynamic security obtained by means of analytical tools of dynamic simulation would create unfeasible high computational times. In fact, to make this evaluation, it would be necessary to perform numerical solution for the differential equations that model the dynamic behavior of the power system components (asynchronous and synchronous generators with governors and AVRs devices), having a complete analytical model of the power system. Thus, the dynamic simulation time obtained by means of an analytical tool (although being dependent of the power system dimension, range of simulation time and computer processor characteristics) typically reaches several minutes.

Therefore, in order to obtain fast and accurate predictions of the system dynamic behavior, it is mandatory to apply automatic learning (AL) techniques in the DSA and SM modules. Namely, according to Wehenkel in [42], Regression Trees and Artificial Neural Networks typically take less than a millisecond per security index estimation.

Besides computational time, there are still other advantages of applying AL techniques, which are referred in Section 3.4.

2.2.2.1 Automatic Learning Techniques Applied in the Implemented Control Systems

In the control system developed for Lemnos, dynamic security assessment functions are taken care by two modules based on Decision Trees (DTs) and Artificial Neural Networks (ANNs). DTs are used to check security for the operating schedules proposed by the economic dispatch module, with respect to characteristic wind power fluctuations. ANNs are used to give a real-time quantitative security evaluation of the current operating state system, by emulating the expected maximum frequency deviation to the pre-defined wind disturbance. To a more detailed description of these applied AL approaches see [40], [41], and [7].

In the control system projected for Crete, both DT and Hybrid Regression Tree (HRT) approaches are used to evaluate the security of the solutions provided by the UC and ED. The security assessment is performed with respect to the outage of a major gas turbine and/or to a three-phase short-circuit at a critical bus near the wind parks. The operator has the possibility to select one of the implemented methods to classify the dynamic security of the unit scheduling and dispatch solutions. The ANN method is used to provide security monitoring, by emulating the expected minimum value reached by the system frequency and maximum value reached by the rate of frequency change. Details of the unit-commitment/dispatch and dynamic security modules can be found in [19] and [1].

In the Crete control system, the HRT technique is being applied for the first time to provide dynamic security assessment of power systems.

2.2.2.2 Security Functions Provided by HRT, ANN and DT

Here it is necessary to highlight that, while DT can only provide security classification, the HRT and ANN methods, besides security classification, can also provide evaluation of security degree.

The DT and HRT techniques are within the Machine Learning (ML) field, and therefore can provide symbolic security rules similar to those used by human experts (i.e., *"if-then-else"* rules), which can be easily understood, discussed, and eventually adopted by the operators.

On the other hand, ANN provides quite opaque models of the system behavior, being usually compared to a black box composed by inputs (which are the parameters that characterized the system operating condition) and outputs (which are the indices that provide security classification and evaluation of security degree).

3 Application of AL Techniques to Make DSA of Power Systems

3.1 Dynamic Security Assessment of Power Systems

As already explained, the work described in this document concerns with the application of a new hybrid automatic learning approach to make fast dynamic security assessment of power systems. In this context, a short overview of the definition of security assessment (SA) and dynamic security assessment (DSA) is provided in this Section.

Security Assessment (SA) consists on:

Evaluating the security of the power system, or by other words, evaluate its capacity to face foreseen disturbances without leading to violation of the system operating constraints (i.e., to the violation of the equality and inequality algebraic equations of the power system, defined by Dy Liacco).

By performing SA, the most appropriated control actions to take are proposed whenever the system is considered to be no longer secure (i.e., when the system is no longer in the normal state defined by Dy Liacco).

In fact, disturbances occur frequently in a power system, and therefore, the impact that they introduce must be controlled. These disturbances can result from external or internal events (ex: switching actions initiated by operators or internal failure *vs* short-circuits due to lighting), and can be slow or fast (ex: smooth profile load *vs* line or generator tripping or sudden load increase).

In Figure 3.1, a short description of the initial security conceptualization issues due to *Dy Liacco* is presented [20]. This diagram presents the different operating states and transitions that can occur in a power system.

In *Normal State* the system is considered to be “secure”. All the operating constraints of the power system are satisfied (equality and inequality algebraic equations), and the system is operated with maximum economy and with guarantee of security to the occurrence of foreseen disturbances. Here is necessary to highlight that, since predicting future disturbances is difficult, there is always a risk of the system to be classified as “secure” and lose integrity if some severe and unforeseen disturbance occur.

When insecurity is detected, the system is said to be in *Alert State*. In this state if a disturbance occurs the system might fall into *Emergency Sate*. Therefore *preventive control actions* are necessary to be performed, in order to move the system again into *Normal State*.

In *Emergency Sate*, system inequality constraints are violated (ex: overloads, undervoltages, underfrequency, instabilities) due to the occurrence of an actual disturbance, and therefore the system is in the process of losing integrity. In this sate, *emergency control actions* are necessary to be performed in a fast way, in order to avoid partial or complete service interruption (i.e., system collapsing). Therefore, in this state, the time response is critical and economic considerations are left to a second level of concern.

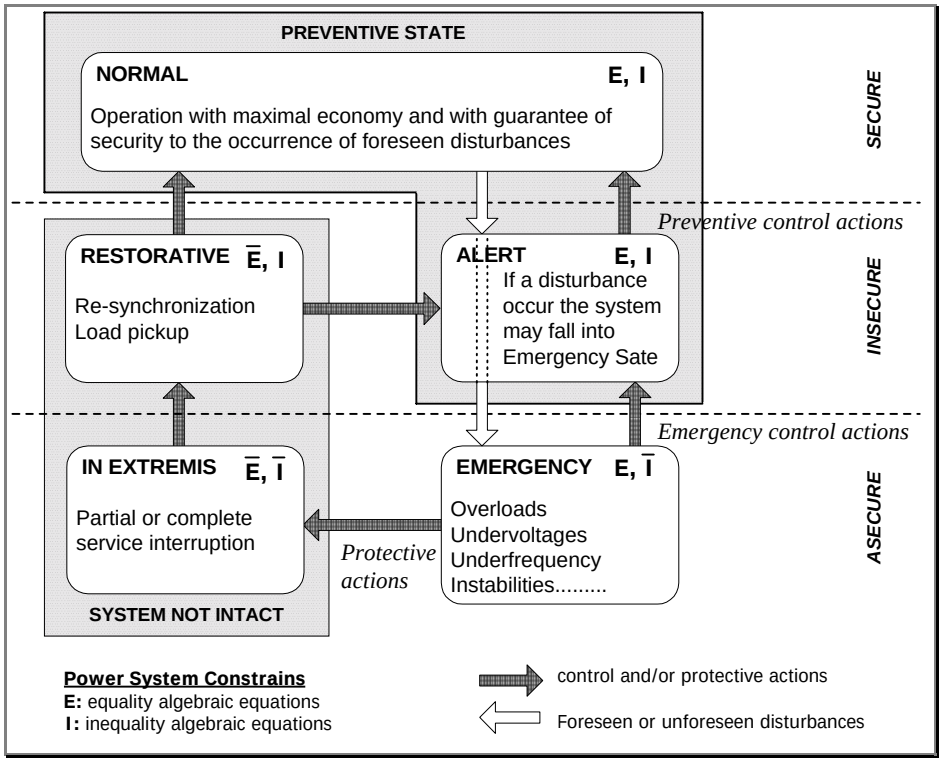


Figure 3.1 – Short description of the Dy Liacco state diagram

If the control actions failed in bringing the system parameters back to values that satisfy inequality constraints, then, in order to avoid damages in system components, the system protection devices will act. This leads to further disturbances, which might result in system splitting and partial or complete service interruption (i.e., violation of the system equality constraints). If this last situation happens, the system will fall into the *In Extremis State*.

Automatically, operators have to minimize the amount and time of undelivered energy, trying to bring system back into *Normal Sate*, by re-synchronizing generators and picking up the disconnected loads.

According to the type of problem and system behavior, SA can be divided into the following two main classes:

- *steady-state security assessment*;
- *dynamic security assessment* (DSA).

Steady-state SA concerns with the capability of the power system to face a realistic set of lines and/or generators outages without leading steady-state parameters to values that violate inequality constraints (ex: components overflows or large bus voltage deviations from nominal value).

Steady-state SA is performed with a power-flow simulation tool. Thus, to have a complete answer about the system security, DSA must be also performed.

DSA main concern is to evaluate the capacity of the power system to face eminent disturbances without system collapsing.

The dynamic problems, which can lead to system collapsing, can be classified into different types of stability problems. Figure 3.2 provides an overview of the existing classes of stability problem according to Kundur [18]. Beneath each class, is presented the *type of phenomena* characteristic of the problem and the *type of physical cause* that leads to the problem.

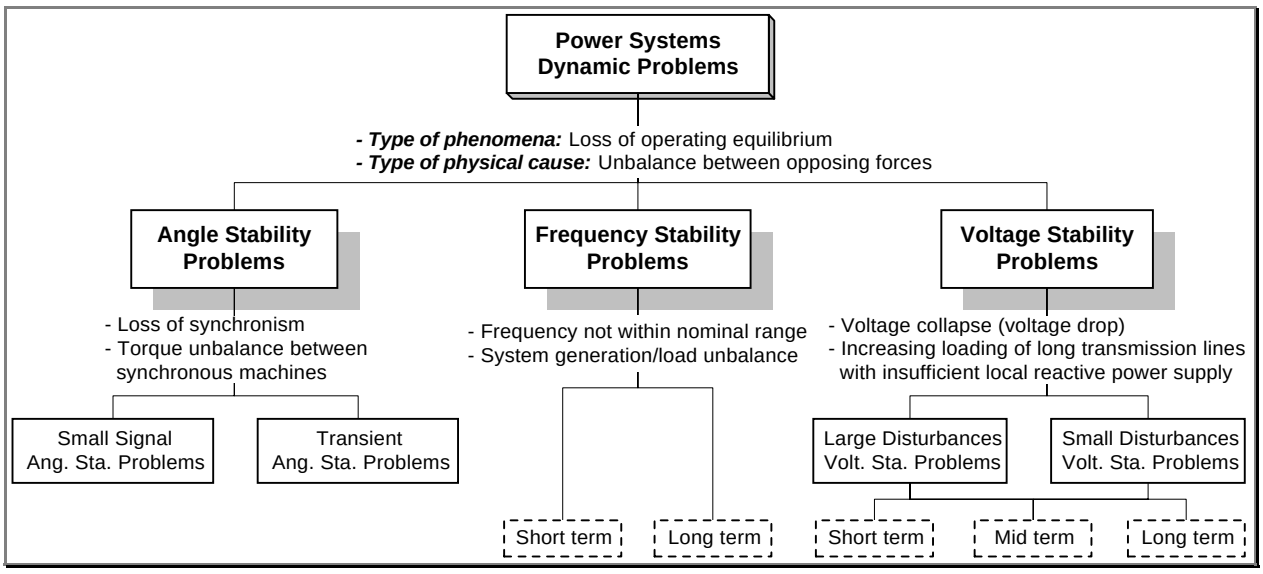


Figure 3.2 – Types of power system dynamic problems

Traditionally, to perform an accurate security analysis of a power system, there are available specific analytical tools that should be adopted according to the type of dynamic problem to be

addressed. These available tools are named analytical tools because they all extract security information by applying numerical methods to analytical models of the power system. They all need a large computational time and, therefore, are unfeasible to be applied for on-line purposes. As already said, the only way to perform DSA, outdating this time consuming problem and without losing accuracy, is by using AL technique application tools.

3.2 Historical Perspective of Automatic Learning

The term *Automatic Learning* (AL) is used to denote a research field concerning the extraction of high level synthetic[2] information (knowledge) from data set containing large amount of low level information.

When the knowledge to extract is a continuous variable y, this last problem is denoted by *regression problem*. This problem consists on obtaining a functional model $y = f(x_1, x_2, ..., x_n)$ that relates the value of y (denoted by *goal variable*) with the values of the variables $x_1, x_2, ..., x_n$ (denoted by *predictors* or *candidate attributes*). The functional model is obtained by extraction from a large set of samples of the unknown regression function, being of the form $(x_1, x_2, ..., x_n, y)$. These samples describe different mappings between the predictors and the goal variable.

Many AL methodologies were developed and applied, including *statistical data analysis and modeling*, *artificial neural networks* (ANNs) and *machine learning* (ML) methods. The early attempts to apply these techniques can be found in [21] (Laplace, 1810) and [22] (Gauss, 1826) in the field of statistic, in [23] (McCulloch et al., 1943) in the field of ANNs, and in [24] (Hunt et al., 1966) in the field of ML [3].

The traditional statistical approaches to regression problem assume a particular parametric function and use all the given samples to estimate the values of the function parameters, where the selected values are the ones that optimize some fitting criterion. An example is the *linear regression* ([25] – Draper & Smith, 1981), where it is assumed a linear model to the unknown regression function, being the values of the model parameters estimated as the ones that minimize the mean squared error.

Modern statistical approaches to regression problems include *k-nearest neighbors* (KNN) ([26] – Fix & Hodges, 1951), *kernel regression* ([5] – Watson, 1964; [6] – Nadaraya, 1964), *local polynomial regression* ([27] – Stones, 1977; [28] – Cleveland, 1979), *artificial neural networks* and others.

[2] In this document, to make a distinction between information obtained by means of analytical tools and by means of AL techniques, the latter is named as *synthetic information* and the former as *analytical information*.
[3] This historic information was adapted from Wehenkel work [45].

The application of AL techniques must look into account three main issues:

- predictive accuracy;
- computational efficiency;
- comprehensibility.

Machine learning (ML) is a class of AL techniques, which concerns with the automatic design of interpretable symbolic rules, similar to those used by human experts (see example of extracted *"if-then-else"* rule in Section 3.5). Therefore, comprehensibility has always been considered a key advantage of ML approaches. Examples of ML techniques are the *Decision Tree* (DT) and *Regression Tree* (RT) methods ([4] – Breiman et al., CART: Classification And Regression Trees, 1984).

In the last two decades, AL techniques have been the focus of study of many researches and applied in several areas. Experimental comparisons of different learning methods on various real world problems have shown the impossibility to select a method that performs better in all domains ([29] – Michie et al., 1994). This is sometimes called the *selective superiority problem* ([30] – Broadley - 1995).

In the context of power systems SA, the first research works with AL techniques started, in the late sixties and seventies, with the statistical Pattern Recognition (PR) methods ([31] – Dy Liacco, 1968; [32] – Pang et al., 1974; [33] – Gupta et al., 1975). In the early attempts, the methodology was essentially limited by the small size of the data set that could be managed and by the parametric nature of the existing PR methods, being unable to handle properly the highly non-linear characteristic of the power systems security problems. Since mid eighties, researches in AL have produced new regression techniques able to handle the complexity and highly non-linearity of power system security problems. In particular, much work has been done in the field of ANNs and ML methods. This can be seen by the large number of existing publications about ANN and ML application to make security assessment of several dynamic power systems problems (just to quote a few examples see from [34] to [43])[4].

One of the main reasons responsible for the latest increasing interest for AL techniques in the field of power systems SA, was the tremendous development achieved in the computation field. This made possible to generate rich data sets to real large scale problems with acceptable response times, and also feasible to apply the compute intensive AL algorithms into this large data sets.

Another, not less important, reason that is contributing for the increasing interest for AL techniques applied to power systems SA, is that further and further power systems have to operate closer to their operating limits and with more erratically behavior. This operating policy is leading to increasing the complexity of power systems security assessment. The common practice of engineers is to use, in an off-line procedure, analytical tools of power system

[4] Some of this historic information was adapted from Wehenkel work [45].

behavior simulation, together with their expertise, to run some scenarios. From those simulations, they extract the relevant security information in order to define planning and operating strategies. However, with the growing complexity of the security assessment problem, only with AL technique application tools the power systems engineers can control the system security in a more efficient way.

3.3 Problem Formulation to Apply AL Techniques for DSA of Power Systems

3.3.1 General AL Problem Formulation

The generic problem of AL techniques application, can be formulated as follows:

> Given a large set of *samples* of the system behavior - a *learning set* containing large amount of *inputs/output* pairs of the system -, extract the best approximation to the unknown function of the existing relationship between the *inputs* and *output*, which might be used to predict the *output* value for any new unseen vector of *inputs*, and/or explain the observed pairs.

When the output is a numerical continuous variable, the AL problem is usually called *regression problem*. When it is a categorical variable[5], the AL problem can be named *classification problem*.

3.3.2 AL Problem Formulation for DSA

In the context of AL techniques application for the DSA of power systems, the parameters of the previously described problem formulation are the following:

- *The inputs*: form a system *operating point* (OP), being characterized by a *measurement vector* of *candidate attributes* with a standard structure, i.e.:

  $$OP = [a_1, a_2, ..., a_{Na}] \quad (3.1)$$

  Each candidate attribute a_i is a value that characterizes one specific operating parameter of the system. Candidate attributes can consist on values measured from the system (like power generations, voltage magnitudes, and consumption) or in values derived from those measures (like spinning reserve, wind margin, and wind penetration). The *measurement hyperspace A* is defined as the hyperspace that contains the measurement vectors.

- *The output:* usually named as *goal variable*, can be a continuous or categorical value B that quantifies/classifies the power system dynamic behavior to a pre-defined disturbance.

[5] The values of a categorical variable belong to a finite set not having any natural ordering.

In this document, the continuous goal variables are called *security indices*, and the categorical ones are called *security classifiers*. Examples of possible security indices are the maximum and minimum values reached by transient frequency and the maximum value reached by the rate of frequency changes. An example of one of the most used security classifier, is a two classes one that classifies the system as "secure/insecure". In this document, only this type of security classifier is going to be considered.

Security indices provide knowledge of the system *security degree* (also called *security robustness* or *security margin*), by presenting the "distance" to the *security boundary* B_0. If the security index B of an OP violates its security boundary B_0, that OP is considered as "insecure", otherwise it is considered as "secure".

- *The learning set:* summarizes the knowledge of the power system dynamic behavior, obtained from a large number of dynamic simulations provided by analytical tools, which are screened off-line via massive random sampling. The learning set (LS) is composed by a set of pre-analyzed scenarios, being defined by:

$$LS = \{(OP_1, B_1), \ldots, (OP_{N(LS)}, B_{N(LS)})\} \quad (3.2)$$

- *One sample:* is a pre-analyzed scenario, being defined by an inputs/output pair (OP,B).

Using the nomenclature presented above, the problem formulation of the application of AL techniques for DSA of power systems can be summarized as follows:

Given a LS, containing a large amount of pre-analyzed scenarios (OP,B) of the power system dynamic behavior to a pre-defined disturbance, extract the best approximation to the unknown function $B=f(OP)$, which might be used to predict the B value for any new unseen OP, and/or explain the observed pairs of (OP,B).

In this document, the synthetic information extracted from the LS, as being the best approximation to the function $B=f(OP)$ is denoted by *security structure*. After the extraction procedure, the security structures can be used in the existing control center, to perform dynamic security assessment functions.

3.4 Functions Provided by AL Security Structures

The functions of security assessment provided by the AL synthetic security structures, can complement the ones provided by the classical analytical power system models, in two main ways:

- computational efficiency;
- interpretability.

In the sense of computational efficiency, by using AL security structures, instead of analytical power system models, much higher speed might be reached when predicting the response B of the system for a new unseen OP. This makes feasible to transfer most of the manual and off-line tasks, of extracting security information, to be automatically performed and in an on-line environment. Namely, by integrating the AL security structures in a proper advanced control system (like the one presented in Section 2.2), the power systems engineers can have tools to control the system security in a more efficient way.

Furthermore, to run AL applications, the data requirements are much lighter. In fact, analytical tools require a full description of the power system operating scenario, while AL security structures might be designed in order to evaluate security by exploiting only the most relevant operating parameters.

In the sense of interpretability, AL techniques can be much more powerful than analytical tools. In fact, there are some approaches of AL techniques, like DT and RT, that might be used to explain the observed pairs of (OP,B) by providing symbolic security rules similar to those used by human experts (see example of Section 3.5). These rules identify **how** the operating parameters influence the dynamic system behavior. Therefore, they can be easily understood, discussed, and eventually adopted by the operators, giving supplementary help to define the operating guidelines that, in present practice, are manually extracted off-line by running analytical tools of the system dynamic behavior.

According to the predicting value, security structures can be exploited in two main ways:

- To make *security classification*, by outputting *security classifiers*;
- To make *evaluation of security degree*, by outputting *security indices*.

Examples of automatic learning techniques that provide security classification are the Decision Trees in the field of Machine Learning ([4], [44], [45]), and many of the Pattern Recognition methods, like Fisher Discriminant Transformation ([44]) and K-Nearest Neighbors rule ([44], [45]).

Examples of automatic learning techniques that, besides security classification, can also provide evaluation of security degree are the Regression Trees in the field of Machine Learning ([4]), and Artificial Neural Networks ([46], [47]).

3.5 Regression Tree Security Structure

Regression Tree (RT) is a non-parametric statistical methodology that deals with continuous goal variables (i.e., consists on a method to solve regression problems). So, the output of a RT security structure is a security index B that measures the security degree of a hypothetical OP to a pre-defined disturbance. The RT consists on a machine learning (ML) method. Thus it provides security structures that can be translated into interpretable security rules.

Figure 3.3 shows an example of a security structure extracted by the RT method, from a hypothetical power system LS with 1844 samples. In this LS, each OP is characterized by the measurement vector $[a_1, a_2]$ = [*Spinning Reserve, Wind Penetration*]. The emulated security index is $B = |\Delta f|_{max}$ = *maximum absolute value reached by frequency deviation to a wind park disconnection*.

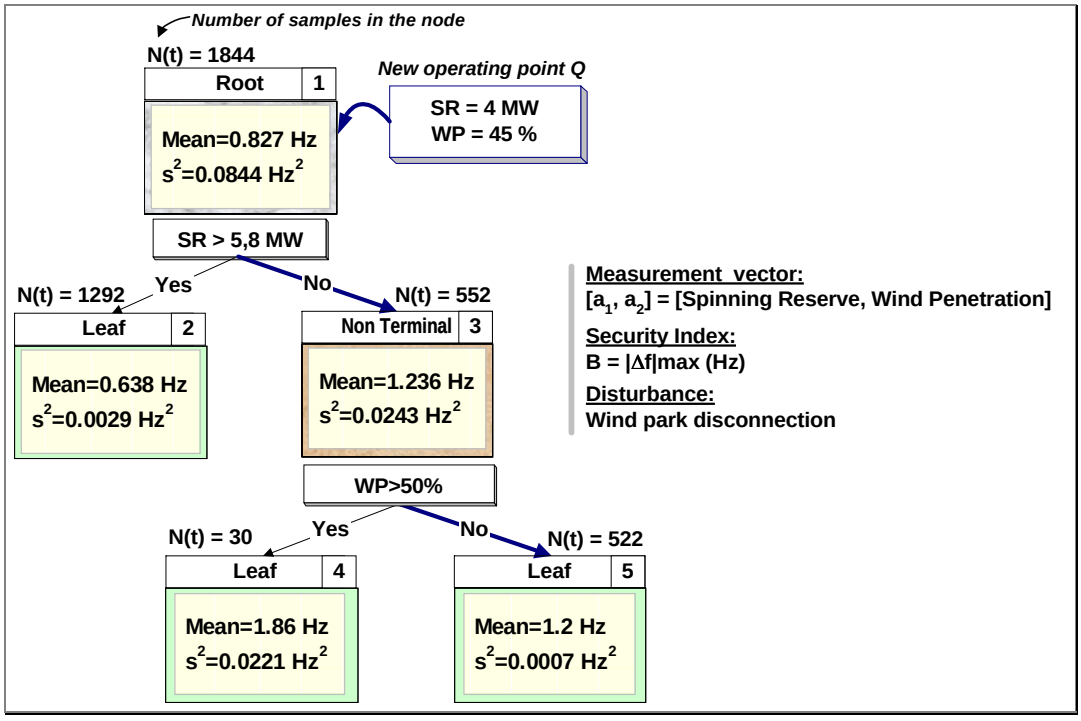


Figure 3.3 – Hypothetical binary regression tree

The security structure consists on a binary Regression Tree (RT), which is composed by nodes and arcs. Each node contains a set of stored OPs from the LS. The first node of the RT, named as *root node*, contains all the LS samples stored. To each *non-terminal node*, it is associated a splitting test applied in the measurement hyperspace A. For instance, the splitting test associated to the root node is {*Load > 5,8 MW*}*?*. On each node t, the splitting test defines the way how the set of stored samples is divided into two disjoint subsets, creating the two successor nodes t_L and t_R.

The terminal nodes of the RT, which are named *leafs*, define a partition of the LS by disjoint regions in such a way that in each region the security index B is as constant as possible. According to this goal, the design of a RT consists on explain as much as possible the mean squared error of the security index B observed in the LS.

Let $N(t)$ be the number of learning samples stored on each node t, and $f_t(OP)$ to be the predicting function to use in the node, then:

$$\text{Mean squared error of B in leaf } t = MSE(t) = \frac{1}{N(t)} \sum_{OP_i \in t} (B_i - f_t(OP_i))^2 \qquad (3.3)$$

In the standard version ([4] - CART, 1984), RTs assign a constant value to the prediction of B on each node. In those cases the predicting function used is the mean value of B, $\overline{B}$. This choice is based on the following elementary lemma:

The constant "a" that minimizes the mean value of $(B-a)^2$

is the mean value of B.

In this case, the design of a RT consists on explain as much as possible the variance of the security index B observed in the LS, where:

$$\text{Variance of B in leaf } t = s^2(t) = \frac{1}{N(t)} \sum_{OP_i \in t} \left(B_i - \overline{B}_t\right)^2 \qquad (3.4)$$

Then, the variance in the Regression Tree RT, of the security index B observed in the LS, is given by:

$$s^2(RT)^{LS} = \sum_{t \in \{\text{Leafs of } RT\}} \frac{N(t)}{N(LS)} s^2(t) \qquad (3.5)$$

In the above equation, $N(LS)$ is the total number of learning samples. For instance, with the hypothetical RT of Figure 3.3, the variance of the LS, which is 0.0844, is reduced to a mean value of $\frac{1}{1844}(1292 \times 0.0029 + 30 \times 0.0221 + 522 \times 0.0007) = 0.00259$. Thus, this RT explains $\left(1 - \frac{0.00259}{0.0844}\right) \times 100 = 96.9\%$ of the variance of B observed in the LS.

Given a new hypothetical unseen operating point, Q = [*4MW,45%*], to evaluate its dynamic security relatively to the considered wind park disconnection, the following procedure needs to be performed:

- Find the leaf that verifies the Q operating conditions. The bold arcs in Figure 3.3, show how the OP of our example, starting at the root node, crosses the tree in a top-down fashion to reach a leaf. During this procedure, being Q in a node t, the successor node where Q must fall is defined by the splitting test of t.

- Predict the $|\Delta f|_{max}$ of Q to the wind park disconnection by applying a function to the samples stored in the leaf. The existing RT approaches differ in the predicting function used in the leafs. For instance, Breiman et al. in CART [4] uses a mean value of B, whereas Karalic in [48] and Quinlan in [49] use a linear regression function. Considering the mean value as the function to apply in the tree leafs, then, for our hypothetical OP, the prediction is that $|\Delta f|_{max}(Q) = 1{,}2\ Hz$.

- Predict the security classification of Q relatively to the wind park disconnection by applying a security criteria of the form:

  If $|\Delta f|_{max}(Q) <$ *security boundary* $\Rightarrow$ ***Q is classified as "secure"****;*
  Else If $|\Delta f|_{max}(Q) \geq$ *security boundary* $\Rightarrow$ ***Q is classified as "insecure"****.*

  For instance, by considering a security boundary of 1 Hz, the OP of our example would be classified as "insecure".

Also by considering the mean value of B as the predicting function to use in the leafs, the RT of Figure 3.3 can be translated into the following set of "*if-then-else*" regression rules:

> ***If*** $(SR > 5{,}8\ MW)$ ***Then*** $(|\Delta f|_{max} = 0{,}638\ Hz)$ ***Else***
> ***If*** $(SR \leq 5{,}8\ MW)$ *and* $(WP > 50\ \%)$ ***Then*** $(|\Delta f|_{max} = 1{,}86\ Hz)$ ***Else***
> ***If*** $(SR \leq 5{,}8\ MW)$ *and* $(WP \leq 50\ \%)$ ***Then*** $(|\Delta f|_{max} = 1{,}2\ Hz)$

By considering a security boundary of 1 Hz, these rules can also be translated into the following classification rule:

> ***If*** $(SR > 5{,}8\ MW)$ ***Then*** *system "secure"* ***Else*** *system "insecure"*

Starting with the LS, the design of a Regression Tree involves the definition of three interrelated issues:

- A way to select a split at every intermediate node;
- A method to determine when a node is terminal;
- A predicting function $f_t(OP)$ to use in the tree leafs, to assign a value B for any new OP that falls into the measurement hyperspace of a leaf t.

At the end results a RT security structure. The method used to design this kind of structures is properly explained in Chapter 5.

From now on, in this document the term T will be used to denote the binary tree structure, whereas the term RT will be used to denote a Regression Tree structure. The RT consists on a binary tree T with a predicting function $f_t(OP)$ in the leafs. Although the existing approaches differ in the used $f_t(OP)$ function, in the work reported in this document, the CART approach [4] was adopted. Therefore, the mean value of B was considered as being the prediction function $f_t(OP)$ to use in the RT leafs.

3.5.1 Terminology and Properties of Binary Trees

The following terminology will be used for binary trees:

T : Binary tree structure
$\tilde{T}$: Set of all the leafs in T
$T - \tilde{T}$: Set of all the non-terminal nodes in T
$|T|$: Number of nodes in T
$|\tilde{T}|$: Number of leafs in T
$|\overline{\tilde{T}}|$: Number of non-terminal nodes or splitting nodes in T
t : Node of T
$t_L = left(t)$: Successor node of t resulting from the verification of its splitting test
$t_R = right(t)$: Successor node of t resulting from the violation of its splitting test
T_t : Subtree of T, which results from having the node t as the root node
$root(T)$: Root of T
$N(LS)$: Number of samples in the LS
$N(t)$: Number of learning samples in node t

For instance, for the example of Figure 3.3:

$\tilde{T} = \{t_2, t_4, t_5\}$
$T - \tilde{T} = \{t_1, t_3\}$
$|T| = 5$; $|\tilde{T}| = 3$; $|\overline{\tilde{T}}| = 2$
$t_{1L} = left(t_1) = t_2$; $t_{1R} = right(t_1) = t_3$
T_{t3} = Binary tree composed by the root node t_3, and by the t_4 and t_5 nodes, which are the tree leafs

Some properties of binary trees are presented bellow:

1. Nodes t_L and t_R are disjoint sets whose union is t
2. The leafs of T form a partition of the LS by disjoint regions
3. $|T| = 2 \times |\tilde{T}| - 1$
4. $|T| = 2 \times |\overline{\tilde{T}}| + 1$
5. $|\tilde{T}| = |\overline{\tilde{T}}| + 1$
6. A subtree T_1 of T is referred as a *pruned tree* of T if $root(T_1) = root(T)$. This can be denoted by $T \succ T_1$

Binary trees have many other properties that, however, are not referred in this document for not being relevant to understand the subjects here described. A detailed description of binary tree properties can be found in [4].

3.6 K-Nearest Neighbors Security Structure

As it was previously referred, K-Nearest Neighbors (KNN) rule belongs to the field of Pattern Recognition (PR) methods, providing security classification.

KNN presents a very simple decision rule. Given a new OP, it will be classified with the majority class among its K nearest neighbors in the LS, being K an odd number. Neighborhood is measured by means of a distance function defined in the measurement hyperspace *A*. Usually, the function used to measure the distance between OPs in the hyperspace *A* is the Euclidean distance.

Figure 3.4 illustrates the application of the KNN method to a simple power system LS, defined in a two-dimensional measurement space *A*.

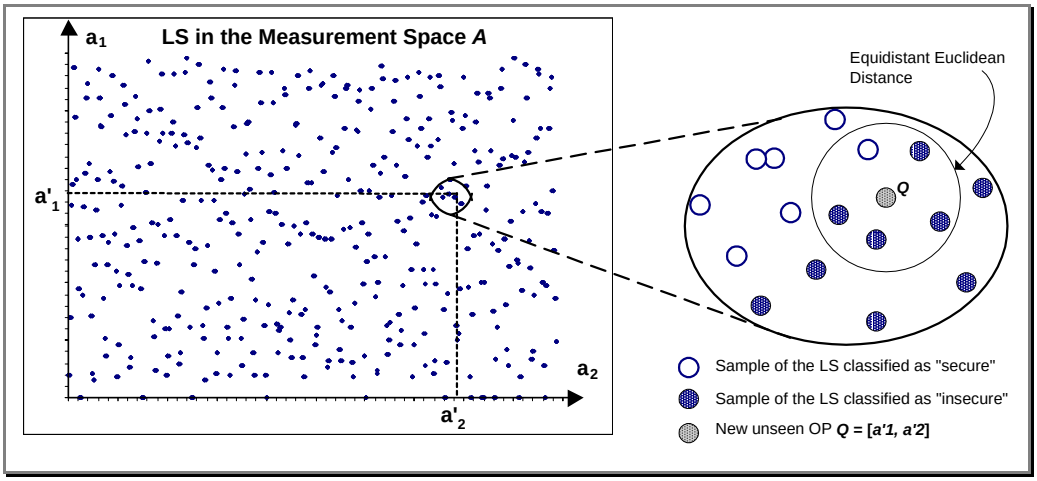


Figure 3.4 – Hypothetical LS and the 5 nearest neighbors for a new unseen OP

The measurement vector [a_1, a_2] characterizes each OP of the LS. For simplicity, the two candidate attributes are considered to have the same units and scale (otherwise normalization would have to be performed before calculating distances between OPs). The emulated goal variable is B = "secure/insecure" to a pre-defined disturbance.

In the right hand part of Figure 3.4, a zoom of the nearest neighbors for a new unseen OP Q = [a'_1, a'_2] is presented. For instance, according to a 5NN rule, and considering the Euclidean distance, this OP would be classified as "insecure", because, among the 5NN, four belong to the "insecure" class and only one belongs to the "secure" class, i.e.:

$$because\ Ki > Ks$$

where:

$$K = Ki + Ks = 5$$
$$Ki : Number\ of\ k\text{ - }nearest\ neighbors\ belonging\ to\ "\ insecure"\ class = 4$$
$$Ks : Number\ of\ k\text{ - }nearest\ neighbors\ belonging\ to\ "\ secure"\ class = 1$$

In the simplest version, given a LS, the design of a K-Nearest Neighbors rule involves the definition of two issues:

- The choice of the k odd value;
- The type of distance function used to measure neighborhood.

3.7 Estimating Accuracy for AL Security Structures

After the construction of a security structure, it is mandatory to know its generalization capabilities, i.e., how accurate it is when predicting B for unseen OPs. Performing this evaluation allows comparing predicting performances between different AL methods, and between security structures extracted by a same AL method. Therefore, the most applied procedures to estimate accuracy of AL security structures – the testing error estimation and learning error estimation – are described below.

3.7.1 Testing Error Estimation

The accuracy of a structure is measured by means of its predicting errors. These errors can be estimated by using a large set of unseen samples of the system, named as testing set (TS). The testing samples must result from the same distribution of the learning samples, but must be independent from them [4]. The most common procedure used to generate a LS and a TS with the properties earlier described, consists on generating a large data set (DS) of samples and then randomly dividing it into a TS and a LS. Being the LS defined as it is presented in equation 3.2, the TS is defined by:

$$TS = \left\{\left(OP_1', B_1'\right), \ldots, \left(OP_{N(TS)}', B_{N(TS)}'\right)\right\} \quad (3.6)$$

Here is necessary to highlight that the accuracy of a security structure strongly depends on the quality of the used LS and TS. Thus, in order to make comparative predicting performance between different AL methods, or between security structures extracted by a same AL method, it is mandatory to use the same LS and TS.

3.7.1.1 Regression Errors

For security structures S that provide a security index B, the predicting error depends on the difference between the true pre-computed values B_i' and the predicted values $f_S\left(OP_i'\right)$ assigned by the S security structure. In these cases, prediction error is usually estimated by the following numerical indices [4]:

$$\text{Mean Absolute Error}(S)^{TS} = MAE(S)^{TS} = \frac{1}{N(TS)} \sum_{OP_i' \in TS} \left| B_i' - f_S(OP_i') \right| \quad (3.7)$$

$$\text{Mean Squared Error}(S)^{TS} = MSE(S)^{TS} = \frac{1}{N(TS)} \sum_{OP_i' \in TS} \left(B_i' - f_S(OP_i') \right)^2 \quad (3.8)$$

Note that, while $MAE(S)^{TS}$ estimates prediction error by making a simple mean value of the mismatches $\left|B_i' - f_S(OP_i')\right|$, $MSE(S)^{TS}$ can highlight some particular high or low value of $\left|B_i' - f_S(OP_i')\right|$. Thus, by choosing a structure S that minimizes $MSE(S)^{TS}$, instead of the one that minimizes $MAE(S)^{TS}$, besides trying to minimize the mismatches $\left|B_i' - f_S(OP_i')\right|$, it is also tried to have closer mismatch values.

The value of these last two predicting errors depends on the scale in which the goal variable B is measured. For this reason, a normalized measure of accuracy, which removes this scale dependence, is often used. This measure is the following:

$$\text{Relative Mean Square Error}(S)^{TS} = RE(S)^{TS} = \frac{MSE(S)^{TS}}{s^2(TS)} \quad (3.9)$$

where $s^2(TS)$ is the variance of B in the TS.

$RE(S)^{TS}$ is always non-negative. Most of the predicting structures S are more accurate than $\bar{B}$, and therefore, in such cases, $RE(S)^{TS}$ <1. Eventually, some construction procedure can result in poor predicting structures, and therefore $RE(S)^{TS} \geq 1$.

3.7.1.2 Classification Errors

For security structures S that provide a two classes security classifier of the form "secure/insecure", the predicting error is not metric. Instead, accuracy depends on the number of classification errors. In these cases, prediction error is usually estimated by the following misclassification rates [7]:

$$\text{Global Classification Error}(S)^{TS} = \frac{\#\{\text{TS samples incorrectly classified by } S\}}{\#\{\text{TS samples}\}} \times 100\,\% \qquad (3.10)$$

$$\text{False Alarm Error}(S)^{TS} = \frac{\#\{\text{"secure" TS samples classified by } S \text{ as "insecure"}\}}{\#\{\text{"secure" TS samples}\}} \times 100\,\% \qquad (3.11)$$

$$\text{Missed Alarm Error}(S)^{TS} = \frac{\#\{\text{"insecure" TS samples classified by } S \text{ as "secure"}\}}{\#\{\text{"insecure" TS samples}\}} \times 100\,\% \qquad (3.12)$$

Obviously, missed alarm error is a misclassification rate with higher importance, since missed alarms correspond to actually "insecure" OPs for which the structure failed to warn. False alarm error is less significant, since false alarms do not cause system security loss. For these reasons, the evaluation of the classification capability of structures is normally mainly based on their global classification error and missed alarm error.

3.7.2 Learning Error Estimation

An also used procedure to estimate accuracy for AL structures is the learning error estimation. It consists on using the LS, i.e., the set of samples used to construct the structure, also to estimate its accuracy. Among others, with this procedure the following predicting error will result:

$$\text{Mean Squared Error}(S)^{LS} = MSE(S)^{LS} = \frac{1}{N(LS)} \sum_{OP_i \in LS} (B_i - f_S(OP_i))^2 \qquad (3.13)$$

The problem of using learning error estimation errors to estimate accuracy is that, as a consequence of not using an independent set of samples, they generally give an overly optimistic picture of the structure accuracy.

Note that equation 3.13 is identical to equation 3.5, by considering the mean value of B to be the predicting function used in the leafs. In fact, the mean squared error of B in the Regression Tree RT consists on its learning error estimation $MSE(RT)^{LS}$.

3.8 Overfitting

In the problem formulations for AL techniques application presented in Section 3.3.2, the sentence "*extract the best approximation to the unknown function B=f(OP)*" in part denotes that the extracted AL structure must be as much as possible proximal to the function *B=f(OP)* presented in the LS. Besides that, it also denotes that, although wanting a structure that models the LS with good accuracy, in order to avoid *overfitting* a limit must be establish to the *complexity* of the extracted structure [45].

A function that *overfitts* the LS will exploit irrelevant information (i.e., noise), and therefore will be sub-optimal in terms of generalization capabilities, i.e., will have lack of accuracy when predicting B for new unseen OPs.

Figure 3.5 illustrates the overfitting phenomenon to a simple one-dimensional problem, i.e., for a security structure with the form $B=f(a_1)$. As it can be seen in this figure, if a complex function $B=f'(a_1)$ is used to make prediction of B, it will result in being very accurate when making prediction for all the learning samples. However, for a new unseen OP Q of the same system, the prediction of its response will be very poor. Using a function with less complexity $B=f''(a_1)$, although being less proximal to the LS, it will allow to improve accuracy when predicting B for the new unseen OP Q.

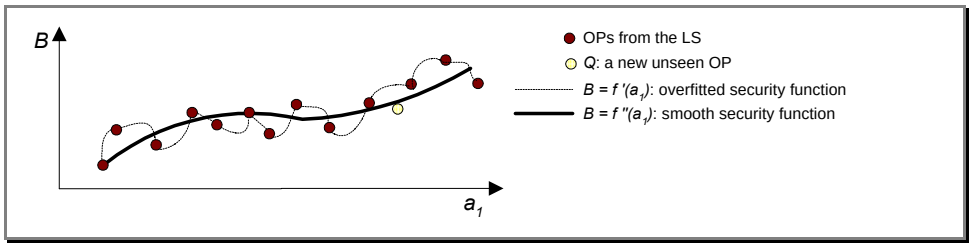


Figure 3.5 – Overfitting illustration

For AL techniques where the extracted security structure consists on a binary tree T, the *complexity* of the structure is proportional to the number of nodes $|T|$. The complexity of a KNN rule is proportional to the number of attributes used in the distance calculation [45].

To control overfitting, it is necessary to reach a trade-off between *bias* and *variance*. When the AL structure is too simple, it presents a large *bias*: i.e., the adaptation is poor and therefore the extracted functional model is too rough. On the other hand, if the AL structure is too complex, it overfitts the LS by exploiting irrelevant information, and therefore, although the *bias* can be small, the structure presents a large *variance* by strongly depending on the random nature of the LS. Both *bias* and *variance* lead to generalization errors, being thus necessary to reach a trade-off between these two errors [45].

3.8.1 Overfitting in Binary Tree Structures

In order to illustrate overfitting for a binary tree structure, an experiment was carried out using the data set generated for the Terceira island, which is properly described in Chapter 4. The considered security index was Δf_{min} – the minimum value reached by the negative frequency deviation resulting from a short-circuit that leads to wind power loss.

By running the developed software of the Hybrid Regression Tree approach described in Chapter 5, several hybrid Regression Trees structures were generated with a decreasing number of nodes. This set of generated trees resulted from applying the pruning algorithm described in Section 5.1.3.

For the extracted hybrid Regression Trees, it was assigned a kernel regression model as being the predicting function to use in the tree leafs. These structures are called in this document as Kernel Regression Trees (KRT). Their accuracy was evaluated by applying testing error estimation (see Section 3.7.1) and learning error estimation (see Section 3.7.2). For learning error estimation the used predicting error was the root mean squared error, $RMSE(KRT)^{LS} = \sqrt{MSE(KRT)^{LS}}$, whereas for testing error estimation the used predicting error was $RMSE(KRT)^{TS} = \sqrt{MSE(KRT)^{TS}}$. The graphical evolution of these two predicting errors, as a function of the complexity $|T|$ of each extracted KRT structure, is presented in Figure 3.6 for $RMSE(KRT)^{LS}$, and in Figure 3.7 for $RMSE(KRT)^{TS}$.

As it can be seen in Figure 3.6, the more splitted the tree is, the lowest is the learning predicting error $RMSE(KRT)^{LS}$. In the most overfitting case, where the tree has 2329 nodes, since each leaf has stored only learning samples with the same B value, all the leafs are pure, and therefore learning error estimation gives a zero predicting error. This example is according to [4], where it is mentioned that, in general, more splits result in lower values of the learning estimating errors.

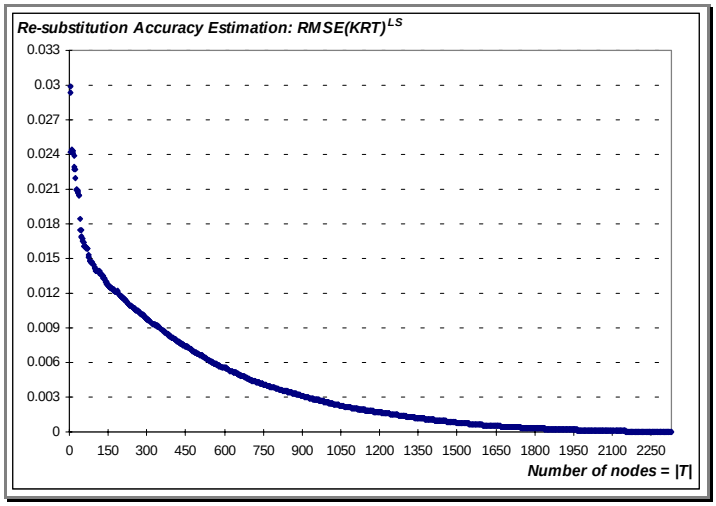


Figure 3.6 – Graphical evolution of $RMSE(KRT)^{LS}$ versus $|T|$, for the extracted KRT structures (Terceira Island)

On the other hand, as it can be seen in Figure 3.7, considering the testing predicting error $RMSE(KRT)^{TS}$, the KRT that maximizes accuracy will no longer correspond to the most splitted tree. As previously said $RMSE(KRT)^{LS}$ gives an overly optimistic picture of the structure accuracy, being thus $RMSE(KRT)^{TS}$ a more accurate estimation of the KRT true predicting error. Thus, the results obtained with testing error estimation presented in Figure 3.7 show that, starting with the most splitted tree, and as the tree initially decreases in size, the true KRT predicting error decreases slowly. Then, at the tree with 133 nodes, the KRT structure hits a minimum. From this point forward, as the tree gets smaller the true KRT predicting error has a fast increase.

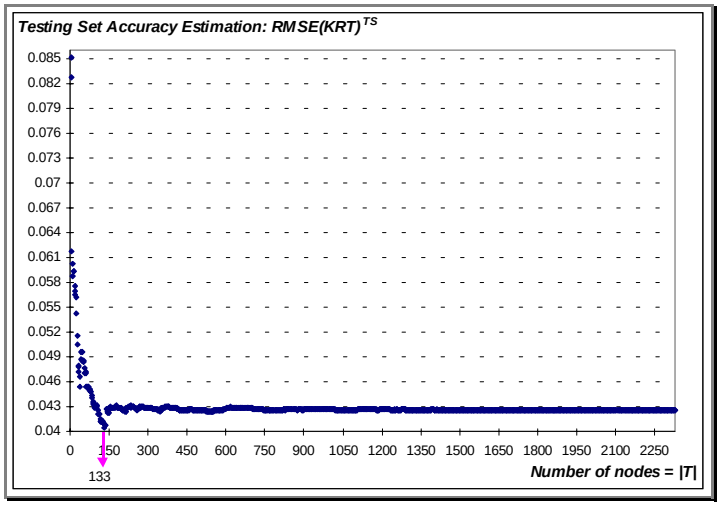


Figure 3.7 – Graphical evolution of $RMSE(KRT)^{TS}$ versus $|T|$, for the extracted KRT structures (Terceira Island)

The behavior described in Figure 3.7 is well known and is also according to [4], where it is mentioned that too large trees will overfitt the LS and therefore will have higher true predicting error than the right sized tree. Too small trees will not use some of the information available in the LS, and therefore will also result in a higher true predicting error than the right sized tree.

In practice, there are many different ways to fight against overfitting, being some specific to a particular type of method and others generic. In the implemented HRT approach described in Chapter 5, two techniques were applied. First was applied the direct use of stop-splitting rules during the growing algorithm of the tree structure. This first technique, although avoiding the tree to grow until having only pure leafs, does not look for the right sized tree. For this reason, subsequently, another more efficient technique was applied. This technique consists on a pruning algorithm that, starting with the most splitted tree, generates a set of pruned trees according to a minimum error-complexity criterion. Then, among this set, the right sized tree is selected according to an accurate estimation of the performance of the designed structures. These two techniques are properly explained in Chapter 5.

3.9 Main Steps to Apply AL Techniques in the Field of DSA

During the reported work, four main steps were considered in order to apply automatic learning techniques to perform dynamic security assessment (DSA) of power systems. All these steps are performed off-line. The final product of the procedure – the security structures – are to be used in an on-line environment in the power system control center, or to obtain physical interpretation of the system behavior. These steps, which are presented in Figure 3.8, are synthetically described below.

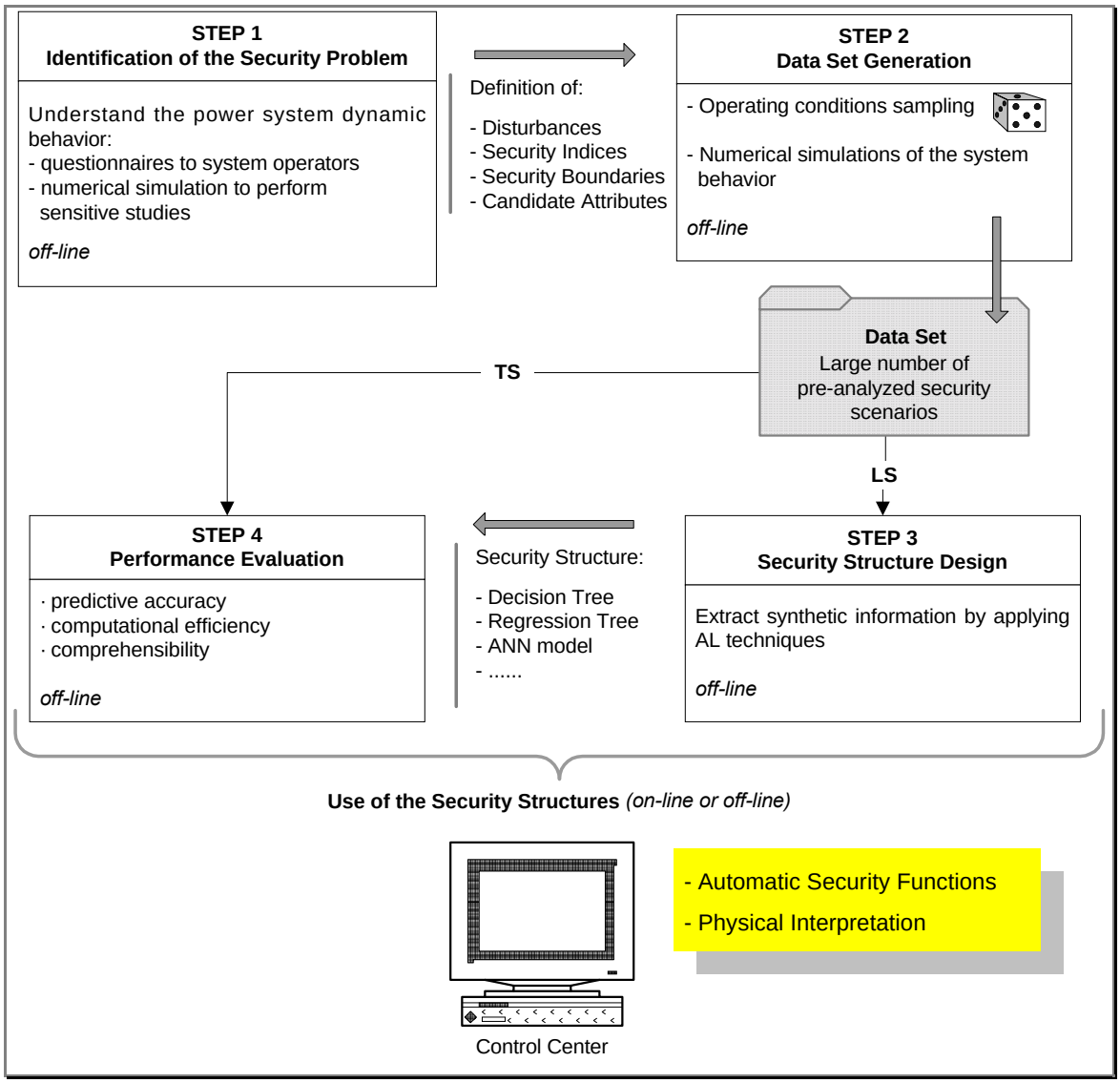


Figure 3.8 – Main steps to apply AL techniques in DSA

3.9.1 STEP1 – Identification of The Security Problem

The first thing to do when applying AL techniques to perform DSA of power systems is to identify the dynamic security problem to evaluate. This analysis involves a procedure of understanding the power system dynamic behavior, namely to identify the potential situations for which the system may lose security. This typically requires making questionnaires to the system

operators, and also performing sensitivity studies by running analytical tools of dynamic simulation.

This first step defines the structure of the data set to generate, namely:

- the disturbances for which is important to known the expected behavior of the system;
- the security indices B to predict and corresponding security boundaries;
- the measurement vector of candidate attribute $OP=[a_1, a_2,..., a_{Na}]$ to use in order to characterize the system OPs.

3.9.1.1 Disturbances Selection

A complete DSA should include all the disturbances that are eminent to occur and might endanger the power system security. Some examples of possible disturbances to considered are the following:

- Fault in the system;
- Loss of generation;
- Sudden increase of large loads;
- Disconnection of transmission or tied lines;
- Wind power disturbances.

3.9.1.2 Security Indices Selection

The selection of the security indices, must be made having in mind that what is important to predict is the "distance" to the security boundary if a pre-defined disturbance occurs. Some examples of typical security indices used for common security problems of power systems are the following:

- Transient instability: critical clearing time and energy margin;
- Frequency instability: maximum and minimum values reached by transient frequency deviation, maximum value reached by the rate of frequency changes;
- Voltage instability: total load increase tolerable by the system before voltage collapse (i.e., load power margin).

3.9.1.3 Candidate Attributes Selection

The selection of the candidate attributes is a very important issue in the procedure because, in order to achieved good results, it is required to use as candidate attributes the power system operating parameters that have influence on the type of dynamic behavior B to predict. Having in mind this fact, all possible influential operating parameters must be considered as candidate attributes.

However, for a practical power system, by considering all possible influential operating parameters, the dimension of the measurement vector $[a_1, a_2,..., a_{Na}]$ can become too large. The use of too many parameters will surely increase the computational effort of the design and predicting stages. Besides, in some methods, the performance of the designed structure can be damaged by the use of irrelevant parameters.

Therefore, after the data set generation, a *feature extraction* process must be performed. This procedure allows the elimination of insignificant phenomena or redundancies, by reducing the dimension of the measurement vector. It consists on applying a statistical mathematical method that chooses the most influential parameters in $[a_1, a_2,..., a_{Na}]$.

For instance, when performing security classification, the significant parameters to find are the ones that are able to distinguish, in the most effective way, the two groups of "secure" and "insecure" OPs. In this field, different statistical mathematical methods have been proven to be effective for performing feature extraction [44].

Candidate attributes are operating parameters that can be directly or indirectly measured from the power system. They can be of the following two main categories [44]:

- Pre-disturbance steady state variables, which characterize the operating conditions before the occurrence of a disturbance (like power generations, voltage magnitudes, consumption, spinning reserve, wind margin, and wind penetration);
- Post-disturbance transient state variables, which characterize the conditions after disturbance occurrence and at disturbance clearance moment (like post-fault network configuration and fault clearing time).

3.9.2 STEP2 – Data Set Generation

This step concerns with the generation of a large data set (DS) of samples of the system behavior (i.e., pre-analyzed security scenarios). These samples will be the input data to the design and performance evaluation steps. In fact, to build a security structure, a learning set (LS) is required, whereas to evaluate its performance characteristics an independent testing set (TS) is also required. The LS and TS, although independent, must result from the same distribution. Therefore, they must be obtained by randomly dividing the DS, resulting in the following sets:

$$LS = \{(OP_1, B_1),..., (OP_{N(LS)}, B_{N(LS)})\} \text{ and } TS = \{(OP'_1, B'_1),..., (OP'_{N(TS)}, B'_{N(TS)})\}$$

As referred in CART [4], the TS is frequently taken as approximately 1/3 of the samples in the DS, belonging the remaining samples to the LS. This partition is made having in mind that if the majority of the DS is used for testing purposes in order to ensure good error estimates, then the quality of the security structure will be reduced. On the other hand, if the majority of the DS is

used for training purposes, then the testing errors will confer a wrong idea about the quality of the designed structure.

In the context of the pruning algorithm applied to Regression Trees, Luis Torgo [50], based on extensive experimentation, claims that to have a sufficient amount of samples to ensure reliable estimates, the following method must be used to decide the size of the TS:

$$\#\{TS\} = min(0.3 \times \#\{DS\}, 1000) \quad (3.14)$$

When the available DS is too small, in order to have enough information to construct the structure and to estimate its accuracy, instead of using a TS, another method, called *V-fold Cross Validation*, needs to be performed [4]. By applying this method, every sample of the DS is used to design the structure. The accuracy estimation is made by using *V* different testing samples, $TS_1,...,TS_V$, obtained by randomly dividing the DS into *V* subsets having, as possible, a nearly equal size. For every $i, i = 1,...,V$, a structure is designed using a learning sample $LS_i = DS - TS_i$, whereas its accuracy is estimated using the TS_i testing sample. The predicting error of the final extracted structure is estimated as being the mean value of the *V* obtained predicting errors. For more information about V-fold cross validation see [4] and [50].

Regarding V-cross validation, the use of an independent TS is computational much more efficient and, therefore, it is the preferable method when the DS contains a large number of samples. Other techniques exist to deal with this issue, however their description is out of the scope of this thesis.

3.9.2.1 DS Generation Method

The data set generation procedure can be summarized as follows:

> *Given an operating range and resolution, a data set of samples (OP,B) is created that reflects the dependency of the system behavior (i.e., security index B) with the variation in its operating conditions (i.e., measurement vector* $OP = [a_1, a_2,..., a_{Na}]$*).*

The data set consists on a large number of samples, where each sample can be considered as a static picture of the system (i.e., there is no time dependency between samples). For the problem under analysis, the operating conditions that are usually considered to change between samples are the following:

- system load level;
- penetration of renewable power sources;
- network configuration;
- unit commitment scheme;
- generation dispatching scheme.

These operating conditions must have high influence on the dynamic behavior B to predict. Otherwise, they will unnecessarily increase the number of samples to generate, without improving the information contained in the DS.

In the generation procedure, among the operating conditions to change, the ones that are independent parameters (i.e., their values do not depend on other operating conditions) are randomly sampled by a systematic method, according to a pre-defined operating range and resolution. Then, for each sample, a unit commitment and economic dispatch module prepare the unit commitment and generation dispatch scenarios. Finally, both measurement vector $OP=[a_1, a_2,..., a_{Na}]$ and dynamic behavior B of each sampled operating scenario are provided by running a proper analytical tool that simulates the system behavior.

In the generation of the Terceira data set, the systematic method used to sample the operating conditions was the structured Monte Carlo sampling method [51]. This method is described in Chapter 4. In this network, the concern was to evaluate its dynamic behavior in the sense of frequency instability. Thus, the analytical tool used to generate the DS was one that involves a power-flow resolution and the numerical resolution of the non-linear differential equations that model the dynamic behavior of the power system components.

3.9.2.2 DS Requirements

The data set generation step is a very important part of the procedure. If the information contained in the data set does not *reflect the mechanism of the system behavior in a proper way*, then, in spite of having a good testing accuracy, there is no assurance that the extracted structures will be accurate enough when making prediction to real life operating scenarios.

For the same reason, the data set should consist on an enough number of samples to *cover all possible states* of the power system under study. Therefore, the generated OPs must cover the breadth of the system operating range and with the best possible resolution. Specially, in order to obtain good accuracy when predicting security classification, the data set must have good resolution in the neighborhood of the security boundary.

To reflect the mechanism of the system behavior in a proper way, when defining the operating scenarios to create the samples, it is necessary *to consider the actual operating practices that are performed in the power system*. Examples of those constraints are the following:

- schedule and dispatch strategies;
- maximum and minimum operating limits of thermal units;
- maintenance programs of thermal units;
- spinning reserve criterion;
- typical load curve models;
- automatic action of voltage regulators in the transformers;
- automatic action of power factor regulators in the power stations.

The operating range and resolution requirements can be improved by generating more samples. However, the computational time for the generation procedure will always introduce some limitation to this number. In the context of Hybrid Regression Trees, there is still another reason to limit the number of samples to generate. In fact, the higher the number of samples in the LS, which does not necessarily lead to a predicting accuracy improvement, the higher the number of samples stored in the Kernel Regression Tree structure, and therefore, the higher will be the time spent to make prediction.

In fact, all the specification of the data set generation must be defined, trying to find a compromise between the DS quality and computational time.

3.9.3 STEP3 – Security Structure Design

After the LS and TS being generated, it is then possible to apply an automatic learning (AL) technique. Like previously referred in Section 3.3, this step extracts from the LS the security structure that is the best approximation to the function $B=f(OP)$. Several AL techniques can be applied, such as Pattern Recognition methods, Artificial Neural Networks, Decision Trees, Regression Trees, or a hybrid model like the Kernel Regression Tree approach described in Chapter 5 of this document.

3.9.4 STEP4 – Performance Evaluation

To select the best security structure within the set of the extracted ones, the designed structures are applied to the TS to evaluate their performances. According to the control center requirements, the security structures can be evaluated by looking into account three main issues:

- predictive accuracy;
- computational efficiency;
- comprehensibility.

As already said, this evaluation is mandatory to be performed since it is the only way that allows comparing predicting performance between different AL methods, and between security structures extracted by a same AL method.

4 Data Sets for the Power Systems of Terceira and Crete Islands

4.1 Introduction

In Section 4.2 of this Chapter, a technical description of the procedure developed to generate a data set (DS) for the Terceira power system is presented. This task was mandatory to be performed, in order to derive synthetic security information to include functions of security assessment in the control center that is expected to be installed in the power system.

This procedure involved the identification of the security problem, the development of a software tool to generate the data set, and finally the partition of the DS to create a learning set (LS) and a testing set (TS). The software tool was developed with M. A. Mitchell (another researcher of INESC Porto), within the framework of its Master thesis requirements. This software tool provides a general methodology to generate data sets for Diesel-wind isolated power systems.

A brief description of the scenario considered for the Terceira electrical network is made in Section 4.2.1. In Section 4.2.2, some issues related to the identification of the Terceira security problem are presented, including the description of the selected disturbance, security indices, candidate attributes, and security criterion. The method implemented to generate the DS is described in Section 4.2.3. For the sake of the better understanding of the methodology developed to deal with the data set generation, the description of the approach is presented in this Section through the example of the Terceira island case. The resulting DS, LS and TS are described in Section 4.2.4. To get a more detailed description of the Terceira electrical network see [14].

For the Crete case study, researchers of NTUA provided the data set required for the extraction of AL security structures. A brief description of the considered scenario for the Crete electrical network is made in Section 4.3.1. In Section 4.3.2, some issues related to the identification of the Crete security problem are presented, including the description of the selected disturbances, security indices, candidate attributes, and security criterion. A short description of the method implemented to generate the DS is described in Section 4.3.3. The provided data set is described in Section 4.3.4. To get a more detailed description of the Crete electrical network see [11]. An explanation of the method used to generate the data set can be found in [1].

The data set generation procedure is an off-line procedure, which concerns with the generation of a large number of pre-analyzed security scenarios of the power system to study, which is performed by running analytical tools that simulate the system behavior. Therefore this procedure requires a high computational effort. In Section 3.9.2., an introduction to the data set generation procedure is provided.

4.2 Data Set of Terceira

4.2.1 Terceira Power System

For the generation of a data set for the Terceira power system, a load consumption level and topology scenario of the year 1999 was considered. This electrical network consists on an isolated Diesel-hydro power system, where it a peak load of approximately 20 MW and an light load of 9.2 MW were considered. Figure 4.1 describes the main topological configuration of the network. This single line diagram presents a reduced version of the network, which was used in this work to make the analytical model of the power system.

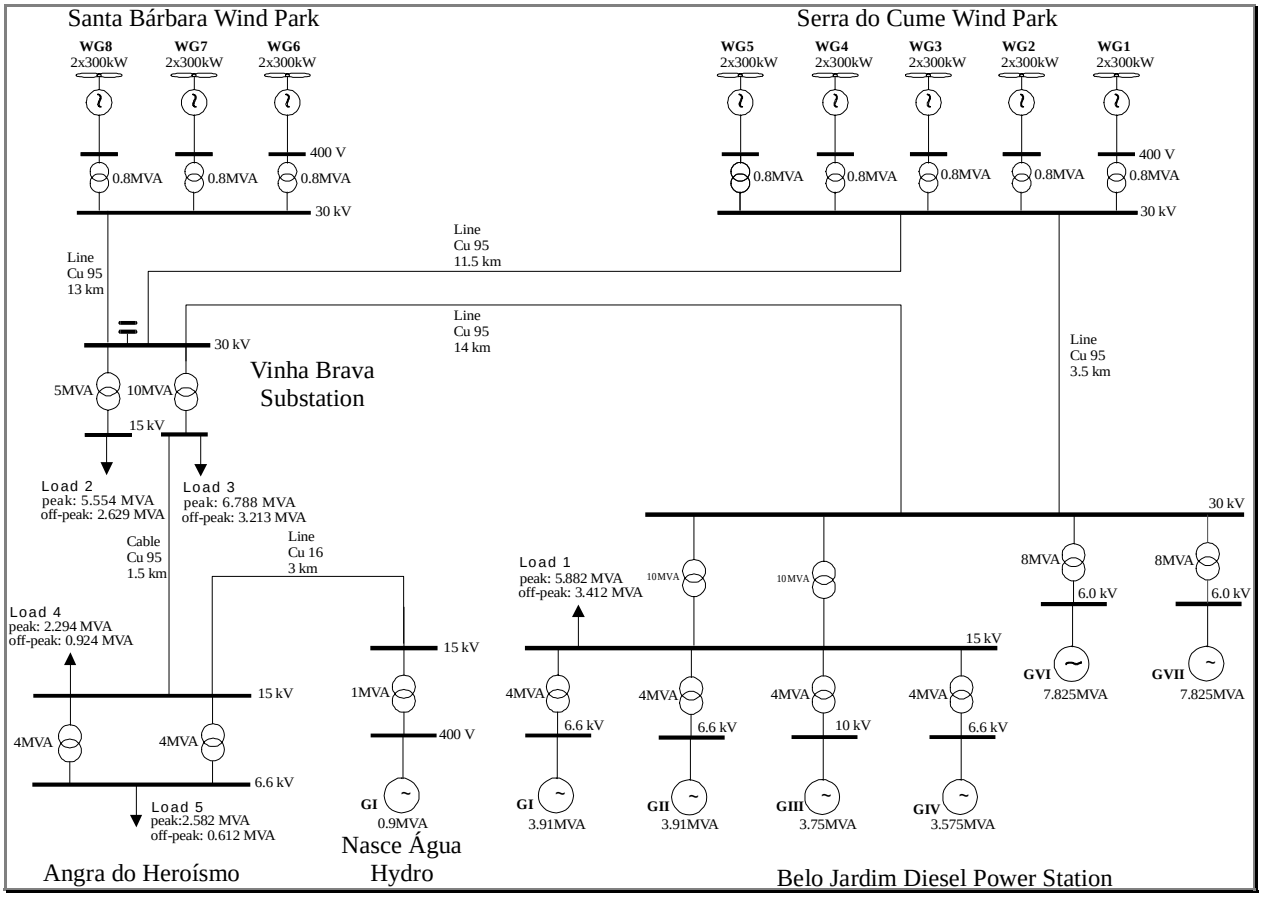


Figure 4.1 – Single line diagram of the electrical network of Terceira in the year 1999

The transmission and distribution network comprises 3 medium voltage levels (30 kV, 15 kV and 6.6 kV). The 30 kV level plays the role of the transmission system in the island. The system includes one Diesel power station (Belo Jardim) and three mini-hydro plants. In the Diesel power station, there are 6 thermal units with a total capacity of 26.88 MW. The hydro power production of the network is represented by the 0,72 MW unit, situated at the power station of Nasce Água. The other hydro units were not considered in operation because of their small contribution.

Although presently there are no wind parks in operation in the island, a considerable amount of wind power production is foreseen to be installed by the year of 1999. Two wind parks, located in two opposite regions of the island, are foreseen:

- Serra do Cume with an installed capacity of about 3 MW (utility owned);
- Santa Bárbara with about 1.8 MW of installed capacity (private owned).

Wind asynchronous generators of 300 kW were considered as the machines to be in operation in the wind parks. The reactive consumption of these machines was considered to be partially compensated through local capacitors connected in parallel with the generators.

The hydro units of the system do not participate in the speed regulation, and therefore this task is assigned only to Diesel machines. In the Diesel power station, there is a management system that monitors and controls the distribution of the reactive power among the machines. According to the investment plan of EDA (the electrical utility of Azores), a SCADA system will be installed soon to help in monitoring and controlling the network.

4.2.2 STEP 1: Identification of the Security Problem for Terceira

As it was already referred in Section 3.9.1, the identification of the security problem is mandatory to be performed in order to specify a proper structure for the data set to generate. This required the realization of sensitive studies to understand the power system behavior, which was made by running extensive numerical simulations, including a steady-state and a dynamic behavior analysis.

To perform these simulations an analytical tool developed by INESC Porto was used, especially designed to deal with this type of analysis. It performs a Newton-Raphson power-flow and a dynamic simulation analysis through the conventional step-by-step integration approach, using a 4^{th} order Runge-Kutta method to solve the differential equations that model the dynamic behavior of the power system components. Regarding the component modeling, the following approach was used:

- Asynchronous wind generators modeled through a 3^{rd} order transient model;
- Synchronous generators represented by the 4^{th} transient model;
- For the exciter and the AVR, the IEEE model 1 was adopted (see block diagrams of Figure 4.2);
- The speed regulator and turbine were represented by a simple model, which includes a proportional and integral control actions (see block diagram of Figure 4.3);
- The loads were modeled, during the dynamic simulation, as constant impedances.

A detailed description of the analytical models and numerical methods used in this simulation tool can be found in [52].

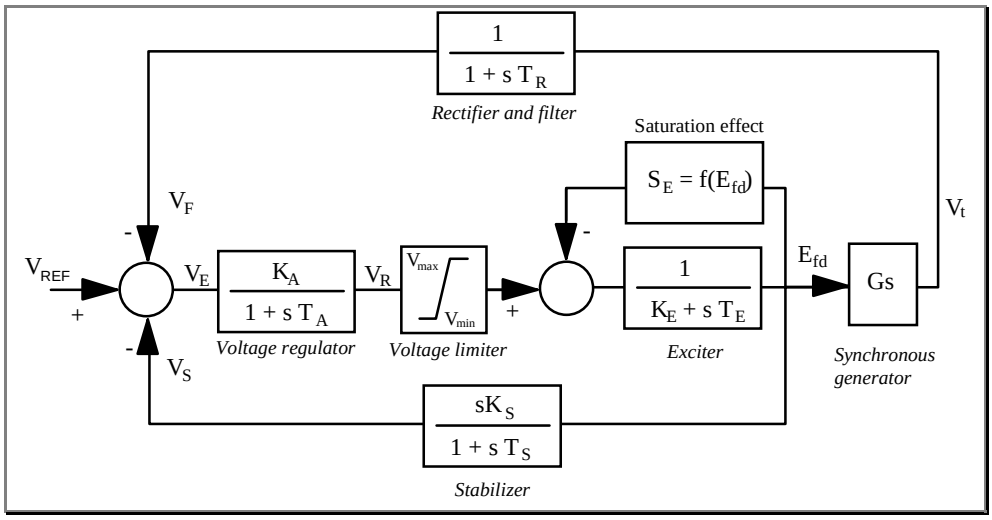


Figure 4.2 – Block diagram of the voltage regulators of Belo Jardim and Nasce Água

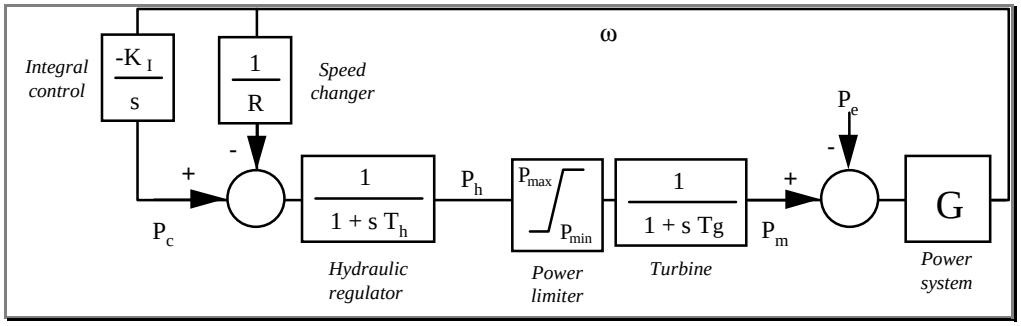


Figure 4.3 – Block diagram of the speed regulators of Belo Jardim

The performed studies showed that the system presents a quite reasonable dynamic behavior regarding wind power disturbances. However, during light load hours with maximum wind power penetration, the degree of dynamic security of the system is reduced. In fact, if, during this load-wind scenario, a short-circuit takes place near the wind park provoking the disconnection of a large amount of wind generators, the system might lose security. In those cases, although the system demonstrated to be dynamically stable, large frequency excursion might lead to the operation of some protection devices that, afterwards, would provoke the system collapse. These studies also showed that, during light load hours, the dynamic behavior clearly depends on the number and type of Diesel units in operation.

From those studies it was concluded that, after the wind parks being installed in the network, an advanced control system would be quite helpful to manage the system operation. This system would be especially necessary during light load hours with large wind power penetration situations, namely by suggesting the most adequate Diesel machines to be in operation or also suggesting the disconnection of some wind generators.

Some of the dynamic responses of the system, resulting from those studies, are presented in Figure 4.4 and Figure 4.5. These figures show the time evolutions of the system frequency deviation, resulting from the simulation of a three-phase short-circuit in Angra do Heroísmo, eliminated after 180 ms, for different operating scenarios. Figure 4.4 refers to the light load scenario with no wind power penetration. Figure 4.5 refers to the same load scenario with maximum wind power penetration. It presents the situation where, due to the actuation of the under-voltage protection of wind generators, Santa Bárbara wind park is disconnected during the fault occurrence. For this scenario two different dispatch schemes are considered:

1) only one large Diesel unit in operation (a 7.825 MVA unit);
2) two small Diesel units in operation (a 3.91 MVA units).

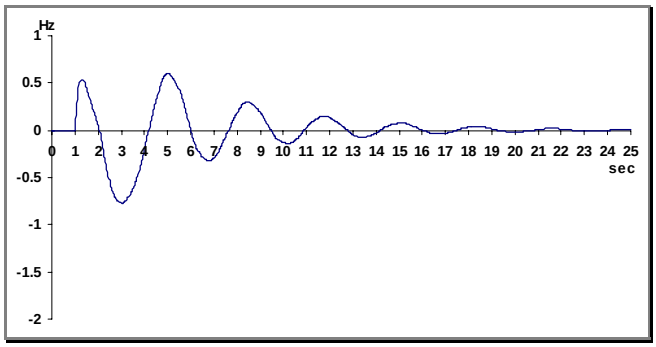


Figure 4.4 – Change of frequency deviation due to short-circuit
(Scenario: light load with no wind power penetration)

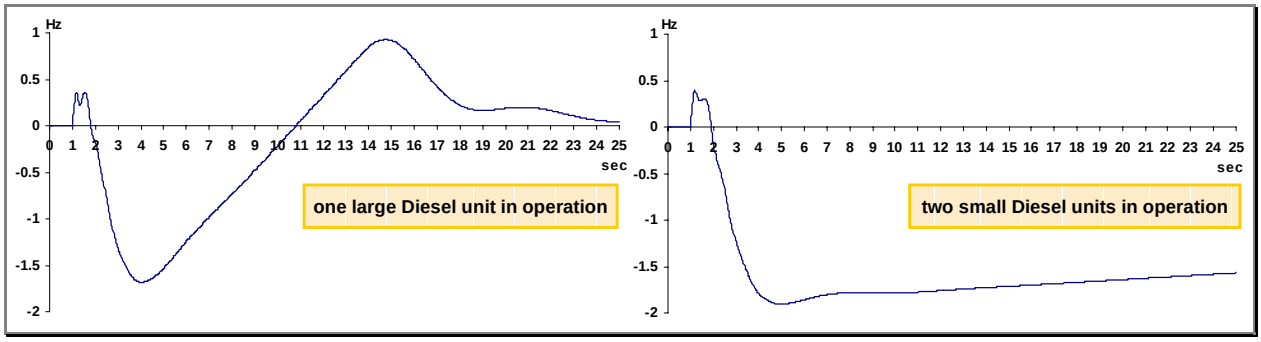


Figure 4.5 – Change of frequency deviation due to short-circuit with disconnection of Santa Bárbara WP
(Scenario: light load with maximum wind power penetration)

From the analysis of the dynamic response of the system to this disturbance for several load scenarios, it was possible to conclude that, if a short-circuit occurs when there are no wind generators in operation, then the frequency deviations are more or less the same, independently on the load scenario. In these situations the system presents a behavior like the one of Figure 4.4, which does not lead to dynamic security loss.

However, when all the wind generators are connected to the network, operating near their nominal power, large excursions on frequency might be observed. During light load scenarios these frequency deviations are particularly severe, having a behavior like the one presented in

Figure 4.5. Namely, when there are two small Diesel units in operation, although the system inertia being higher, the available spinning reserve is smaller and, therefore, frequency excursions can reach values near -2 Hz. These values can lead to the operation of other frequency relays, like the ones installed in the other wind park, which might provoke afterwards to the system collapse.

4.2.2.1 Disturbance Selection

According to the dynamic behavior of Terceira power system, it was considered necessary to known the expected behavior of the system to the following disturbance:

- Tree-phase short-circuit in the 6.6 kV bus of Angra do Heroísmo, eliminated after 180 ms, followed by the disconnection of Santa Bárbara wind park at 100 ms after the occurrence of the default.

4.2.2.2 Security Indices Selection

The security indices selected to measured the dynamic behavior of the network are the following:

- $B_1 = \Delta f_{min}$: minimum value reached by the negative frequency deviation (Hz);
- $B_2 = \Delta f_{max}$: maximum value reached by the positive frequency deviation (Hz);
- $B_3 = df/dt_{max}$: maximum value reached by the rate of frequency change (Hz/s).

4.2.2.3 Security Boundaries Selection

The considered security boundaries are the following:

- $B_{01} = -1\ Hz$ (according to Δf_{min}, a OP is "secure" if $\Delta f_{min}(OP) > -1\ Hz$, otherwise is "insecure");
- $B_{02} = 0.65\ Hz$ (according to Δf_{max}, a OP is "secure" if $\Delta f_{max}(OP) < 0.65Hz$, otherwise is "insecure");
- $B_{03} = 3.4\ Hz/s$ (according to df/dt_{max}, a OP is "secure" if $df/dt_{max}(OP) < 3.4Hz/s$, otherwise is "insecure").

4.2.2.4 Candidate Attributes Selection

In the selection of candidate attributes for the Terceira data set, only pre-disturbances steady-state continuous parameters were considered. The technique used in this work to model generator unit status, was to consider the generator unit as *off* when its generation level is zero, and *on* otherwise. In the case of circuit status, a fixed topology was considered to the network. A fixed value was also considered to the transformers tap and hydro active generation level.

For the measurement vector that characterizes each OP of the Terceira data set, the following candidate attributes were selected:

General Attributes

a_1 = WM[6] in Serra do Cume Wind Park;

a_2 = WM in Santa Bárbara Wind Park;

a_3 = Total WM;

a_4 = Total WP[7] (%);

a_5 = Total SR[8] (MW);

a_6 = Total active generation of Diesel power (MW);

a_7 = Total reactive generation of Diesel power (MVAr);

a_8 = Total active generation of wind power (MW);

a_9 = Total reactive consumption in the wind generators (MVAr);

a_{10} = Total active load (MW);

a_{11} = Total reactive load (MVAr);

a_{12} = Total reactive generation in capacitor banks (kVAr);

a_{13} = Total active losses (kW);

a_{14} = Total reactive losses (kVAr);

Attributes of Belo Jardim Diesel Power Station

a_{15} = Active generation in Diesel unit GI (MW);

a_{16} = Reactive generation in Diesel unit GI (MVAr);

a_{17} = S.R. in Diesel unit GI (MW);

a_{18} = Voltage magnitude in Diesel unit GI (perceptual value to the nominal value);

a_{19} = Active generation in Diesel unit GII (MW);

a_{20} = Reactive generation in Diesel unit GII (MVAr);

a_{21} = S.R. in Diesel unit GII (MW);

a_{22} = Voltage magnitude in Diesel unit GII (perceptual value to the nominal value);

a_{23} = Active generation in Diesel unit GIII (MW);

a_{24} = Reactive generation in Diesel unit GIII (MVAr);

a_{25} = S.R. in Diesel unit GIII (MW);

a_{26} = Voltage magnitude in Diesel unit GIII (perceptual value to the nominal value);

a_{27} = Active generation in Diesel unit GIV (MW);

a_{28} = Reactive generation in Diesel unit GIV (MVAr);

a_{29} = S.R. in Diesel unit GIV (MW);

a_{30} = Voltage magnitude in Diesel unit GIV (perceptual value to the nominal value);

a_{31} = Active generation in Diesel unit GVI (MW);

---

[6] $\mathrm{WM} \rightarrow \textit{Wind Margin} = \dfrac{\textit{Total SR (MW)}}{\textit{Total Active Generation in the Wind Park (MW)}}$

[7] $\mathrm{WP} \rightarrow \textit{Wind Penetration} = \dfrac{\textit{Total Wind Power of Active Generation (MW)}}{\textit{Total Active Load (MW)}} \times 100\%$

[8] $\mathrm{SR} \rightarrow \textit{Spinning Reserve}$

a_{32} = Reactive generation in Diesel unit GVI (MVAr);

a_{33} = S.R. in Diesel unit GVI (MW);

a_{34} = Voltage magnitude in Diesel unit GVI (perceptual value to the nominal value);

a_{35} = Active generation in Diesel unit GVII (MW);

a_{36} = Reactive generation in Diesel unit GVII (MVAr);

a_{37} = S.R. in Diesel unit GVII (MW);

a_{38} = Voltage magnitude in Diesel unit GVII (perceptual value to the nominal value);

Attributes of Serra do Cume Wind Park

a_{39} = Active generation in wind generator WG1 (MW);

a_{40} = Reactive consumption in wind generator WG1 (MVAr);

a_{41} = Active generation in wind generator WG2 (MW);

a_{42} = Reactive consumption in wind generator WG2 (MVAr);

a_{43} = Active generation in wind generator WG3 (MW);

a_{44} = Reactive consumption in wind generator WG3 (MVAr);

a_{45} = Active generation in wind generator WG4 (MW);

a_{46} = Reactive consumption in wind generator WG4 (MVAr);

a_{47} = Active generation in wind generator WG5 (MW);

a_{48} = Reactive consumption in wind generator WG5 (MVAr);

Attributes of Santa Bárbara Wind Park

a_{49} = Active generation in wind generator WG6 (MW);

a_{50} = Reactive consumption in wind generator WG6 (MVAr);

a_{51} = Active generation in wind generator WG7 (MW);

a_{52} = Reactive consumption in wind generator WG7 (MVAr);

a_{53} = Active generation in wind generator WG8 (MW);

a_{54} = Reactive consumption in wind generator WG8 (MVAr).

4.2.3 STEP 2: Data Set Generation Method Applied for Terceira

In the procedure applied to generate the DS of Terceira, the technique used to change the operating conditions between samples was the structured Monte Carlo sampling method. This method was developed by McCalley et al. [51], being in this research work adapted and extended for the generation of DS for isolated systems. As this procedure is part of the research work of another Master student, in this thesis only the relevant parts of the approach are described.

The applied procedure required an initial specification, which covers the definition of the following aspects:

- Operating conditions to change between samples;
- Monte Carlo parameters;
- DS operating range and resolution;
- Operating constraints;
- Maximum number of samples to generate.

For the reasons already explained, in the specification of the DS a compromise between the DS quality and computational time was necessary to be defined, which imposed a limit to all the DS specification.

4.2.3.1 Operating Conditions to Change

For the creation of the Terceira DS, the change of operating conditions between samples was obtained by changing:

- the load level in each load busbar;
- the wind power in each wind park, where the same generating level was considered for the wind generators of the same wind park;
- and the unit commitment scheme of the Diesel units (i.e., the status *on/off* of the Diesel generator units).

4.2.3.2 Monte Carlo Parameters

The Monte Carlo parameters (a_{MC} *parameters*) consist on the operating parameters in which the sampling method is applied in order to change the operating conditions between samples. Obviously, the value of the Monte Carlo parameters has to be independent from the operating conditions, because otherwise they could not be sampled. For the Terceira DS, there were identified the following Monte Carlo parameters:

- active load level in each load busbar (P_{load});
- mechanical power in each wind park (P_{mec});

which gives a total of 7 Monte Carlo parameters (5 for the load busbars and 2 for the wind parks).

4.2.3.3 DS Operating Range and Resolution

After identifying the Monte Carlo parameters (a_{MC} *parameters*), the definition of the *DS operating range* was included. This was done by identifying the operating range for each a_{MC} parameter, i.e., the credible minimum $a_{MC,min}$ and maximum $a_{MC,max}$ values.

The definition of the desired *DS resolution* was also performed. This was done by defining the resolution for each a_{MC} parameter, i.e., the number *"n"* of intervals of each a_{MC} operating range where each interval has a range of $(a_{MC,max} - a_{MC,min})/n$.

For the Terceira DS, the operating range and resolution defined for each one of the a_{MC} parameters are presented in Table 4.1.

Table 4.1 – Operating range and resolution considered for the Data Set of Terceira

Monte Carlo parameter	Operating range	Resolution
a_{MC1}: P_{mec} in Santa Bárbara wind park	[0;1.8] MW	$n_1 = 5$
a_{MC2}: P_{mec} in Serra do Cume wind park	[0; 3] MW	$n_2 = 5$
a_{MC3}: $P_{load,1}$	[3; 6] MW	$n_3 = 8$
a_{MC4}: $P_{load,2}$	[2.5; 6] MW	$n_4 = 2$
a_{MC5}: $P_{load,3}$	[3; 7] MW	$n_5 = 2$
a_{MC6}: $P_{load,4}$	[0.5; 2.5] MW	$n_6 = 1$
a_{MC7}: $P_{load,5}$	[0.5; 3] MW	$n_7 = 1$

By defining the resolution of each a_{MC} parameter, the DS operating range was divided into *intervals* (in one dimension), *cells* (in two dimensions), *irregular cubes* (in tree dimensions) and *hypercells* (four or more dimensions).

4.2.3.4 Generation Procedure

In the generation procedure, the following steps were applied:

1. The load levels and wind powers (i.e., the a_{MC} parameters) were randomly sampled by the structured Monte Carlo method, according to the pre-defined operating range and resolution. This method consists on a step-by-step procedure, in which, for each hypercell, a *load/wind operating scenario* was randomly sampled.

2. For each sampled *load/wind scenario*, a unit commitment (UC) module prepared a set of possible *scheduling schemes* for the Diesel power station.

3. For each *scheduling scheme*, an economic dispatch (ED) module prepared an *operating point* corresponding to the minimum cost dispatch scenario.

4. Finally, for each defined *operating point*, both measurement vector $OP = [a_1, a_2, ..., a_{54}]$ and security indices $[\Delta f_{min}, \Delta f_{max}, df/dt_{max}]$ were provided by running the previously referred analytical tool of power-flow calculation and dynamic simulation developed by INESC Porto.

This data set generation procedure included a decision of which numerical values in each hypercell to sample. This decision was accomplished by the "Monte Carlo" part of the sampling method, which consists on making a random selection.

One simple approach, named *structured sampling*, would be to sample the center of each hypercell. However, by applying a random selection, a maximum resolution was possible to achieve for each a_{MC} parameter (i.e., the number of different values sampled for each a_{MC} parameter equals the number of operating scenarios to generate). This can be seen in Figure 4.6, which illustrates the *structured sampling* and *structured Monte Carlo sampling* applied in a two-dimension problem.

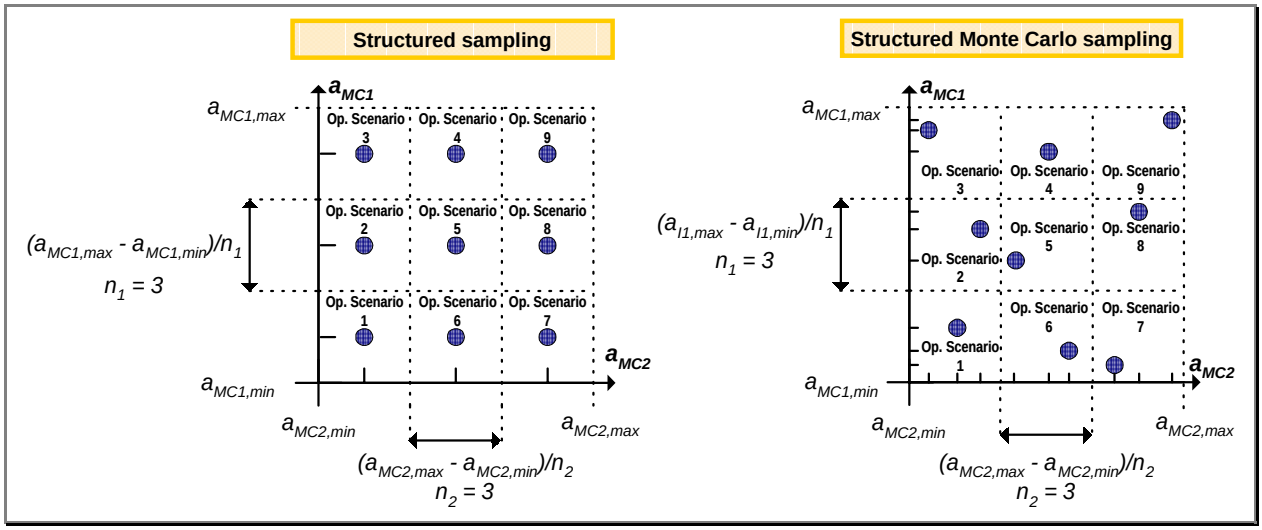


Figure 4.6 – Illustration of *structured sampling* and *structured Monte Carlo sampling* in a two-dimension problem

As we can see in this figure, as the problem under analysis has 9 operating scenarios to generate:

– by applying the *structured sampling*, only 3 different values per parameter are sampled;
– whereas, by applying the *structured Monte Carlo sampling* 9 different values are possible to sample per parameter.

The flowchart of the software tool, developed to generate the data set of Terceira, is presented in Figure 4.7. Note that it is possible that the defined pre-disturbance operating condition result in a non-converged power-flow or dynamic simulation solution. To consider this possibility in the generation algorithm, when a non-convergent solution is detected, the correspondent sample is not kept in the output file and the algorithm automatically starts the generation of the next sample.

For a given wind and load level, there is a wide range of unit commitment possibilities[9]. Since each unit commitment solution adds one sample to the data set, a limit must be performed to this number. To accomplish this, at the end of the economic dispatch calculations, the algorithm orders these solutions by associated increasing cost, and then selects only the first n_{ED} solutions (i.e., the dispatch solutions with less production cost). For instance, a $n_{ED} = 5$ was considered for the generation of the Terceira DS.

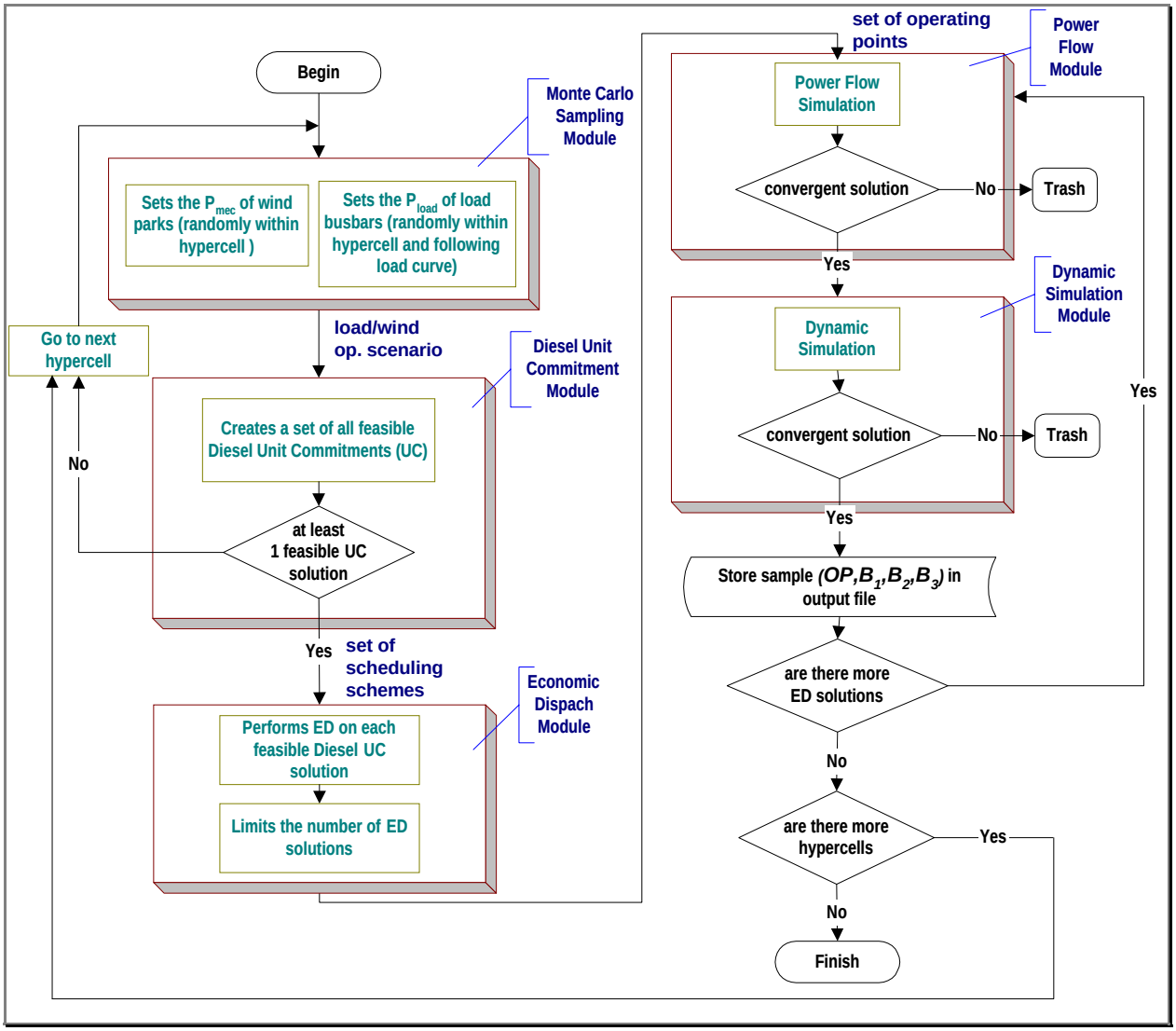


Figure 4.7 – Flowchart of the developed software to generate the Data Set of Terceira

4.2.3.5 Operating Constraints

During the definition of the operating points, the following constraints were included:

- typical load curve models;
- maximum and minimum operating limits of the Diesel units;

[9] Theoretically, there are $2^K - 1$ possible combinations, being K the number of existent Diesel units. In practice, not all combinations are feasible because of the imposed practical operating constraints.

- spinning reserve criterion;
- maintenance programs of the Diesel units;
- production cost of the Diesel units;
- automatic action of the local capacitors in the wind parks;
- automatic action of the management system that controls the distribution of the reactive power among the Diesel machines in Belo Jardim power station.

These constraints were included within the Monte Carlo, unit commitment, economic dispatch and power-flow modules. A description of these constraints is presented bellow.

1 - Constraint of the Monte Carlo Method - Typical Load Curve Model

The load buses of the transmission and distribution network normally have typical daily load curve models, that can be obtained via historical data.

This typical behavior was included in the DS generation algorithm, by associating to each load busbar a control string like the one presented in Figure 4.8. The example of this figure presents the load curve model for a hypothetical P_{load} Monte Carlo parameter, to which it was considered a resolution of $n = 8$. According to this curve, the control string associates a binary variable to each interval of the P_{load} operating range. If the binary variable is 0 then the interval will not be considered in the sampling process, otherwise the interval will be considered.

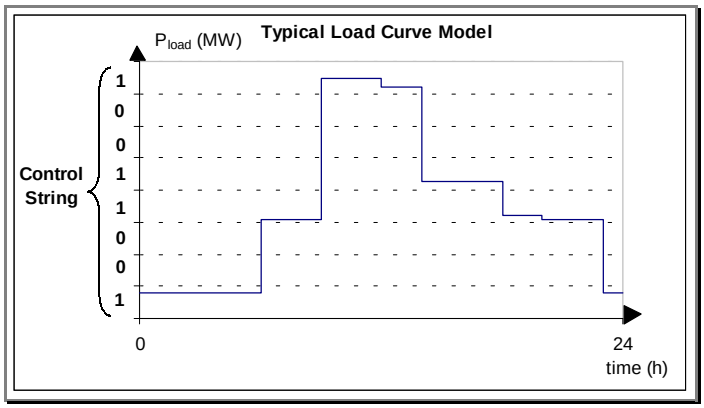


Figure 4.8 – Control string used to model a typical load curve in the DS generation algorithm

For the Terceira DS generation, a typical load curve model was assigned for the $P_{load,1}$ Monte Carlo variable, by considering the control string presented in Figure 4.9.

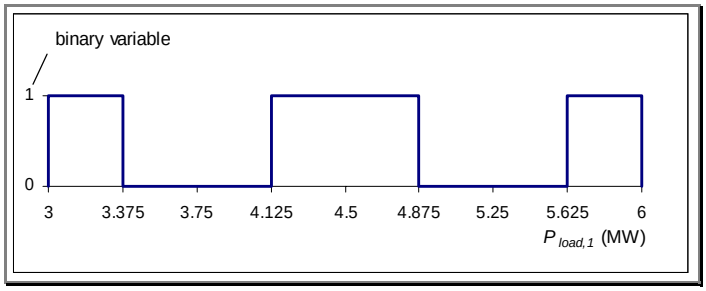


Figure 4.9 – Control string considered to model the typical load curve of the $P_{load,1}$ parameter of Terceira

2 - Constraints of the Diesel Unit Commitment Module

The unit commitment module calculates all the feasible combination of generating Diesel units to serve the Diesel power demand, given by:

$$P_{Diesel} = P_{load} - P_{wind} - P_{hydro} \quad (4.1)$$

where:

> P_{load}*: active load level;*
>
> P_{wind}[10]*: wind generation level;*
>
> P_{hydro}*: hydro generation level.*

This module takes into account the following practical operating constraints:

- maximum and minimum operating limits of the Diesel units;
- spinning reserve criterion;
- maintenance program of the Diesel units.

For the spinning reserve (SR) criterion, the following minimum requirement was used:

$$SR \geq P_{load} \times Load\ Margin + P_{wind} \times Wind\ Margin\ (MW) \quad (4.2)$$

where:

> $Load\ Margin = 10\%$
>
> $Wind\ Margin = 40\%$

During the system operation, there are times when some Diesel unit of the system can be out of service due to maintenance procedures. To capture this aspect, the following model was used:

- To consider no maintenance procedures during most of the operating time, a cycle of maintenance status for the Diesel power station was defined, where a large perceptual time of this cycle was set as having all the Diesel units available to operate.

- The remaining time of the cycle was divided between all the units, giving a mean time when each unit is out of service due to maintenance procedures. From this procedure resulted a cycle of maintenance status like the one presented in Figure 4.10.

- Having defined the cycle of maintenance status for the Diesel power station, when calculating the set of feasible unit commitments, a number is randomly selected within 0 and 100%, which gives the maintenance status of the Diesel units to consider.

[10] In the unit commitment part of the algorithm, the wind generation level is not know (once it is an output of the power-flow calculation), being thus set equal to the total mechanical power level in the wind parks.

To model the maintenance program of the Belo Jardim Diesel units, the cycle of maintenance status of Figure 4.10 was used.

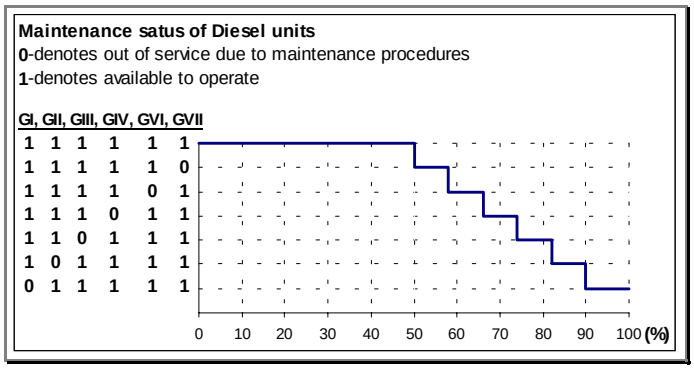


Figure 4.10 – Cycle of maintenance status considered for the Diesel power station of Terceira

3 - Constraints of the Economic Dispatch Module

Once the set of feasible unit commitment solutions is determined, to each UC solution, the generation algorithm must define a value for the active generation of the operating Diesel machines. To accomplish this, an economic dispatch module was considered that performs a ED solution for each feasible UC solution, by considering the following constraints:

- The sum of the output powers in the operating Diesel units, $\sum P_i$, must equal the Diesel power demand $P_{Diesel} = P_{load} - P_{wind} - P_{hydro}$.
- Each Diesel unit is within its minimum and maximum operating limits.

In this module, the following quadratic curve was assumed for the production cost function of the Diesel machines:

$$F_i(P_i) = A_i P_i^2 + B_i P_i + C_i \ \ (\$/h) \qquad (4.3)$$

where:

> *A_i, B_i and C_i: constant values;*
>
> *P_i: power output of generator i (MW).*

For the Diesel units of Belo Jardim power station, the following production cost functions were considered:

$$F_{GI} = F_{GII} = F_{GIII} = 0.085P_i^2 + 6.5P_i + 200 \ \ \$/h$$
$$F_{GIV} = 0.095P_i^2 + 7.5P_i + 300 \ \ \$/h$$
$$F_{GVI} = F_{GVII} = 0.075P_i^2 + 5.5P_i + 100 \ \ \$/h$$

4 - Constraints of the Power Flow Module

Automatic Action of the Local Capacitors in the Wind Parks:

Typically, when the wind generator is an asynchronous machine, its reactive consumption is partially compensated through local capacitor banks that are connected in parallel with the generator. In those systems, the MVAr value of the local capacitor bank may depend on the generator load level.

To consider this behavior in the data base generated for Terceira, just before the power-flow calculation, the algorithm sets the MVAr value of the local capacitor banks of each wind generator, according to the curve of Figure 4.11.

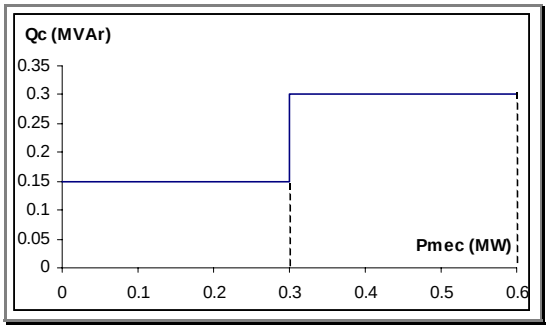


Figure 4.11 – MVAr value of local capacitor bank *versus* Pmec in wind generators considered for the Terceira DS

Control of the Distribution of Reactive Power Among the Diesel Machines:

To model the action of the management system that controls the distribution of reactive power among the Diesel machines of Belo Jardim power station, an automatic reactive compensation feedback loop was introduced after each power-flow calculation. This loop guaranties that none of the Diesel machines is consuming reactive power.

4.2.3.6 Maximum Number of Generated Samples

Considering the previously described specifications for the generation of Terceira data set, the maximum number of samples to generate was therefore set equal to:

$$[n_1 \times n_2 \times (n_3 - 4) \times n_4 \times n_5 \times n_6 \times n_7] \times n_{ED} = 400 \times 5 = 2000 \text{ samples}$$

4.2.4 Data Set Results for Terceira

Using the approach described in this Section, 1976 acceptable samples were obtained. This DS was divided into the LS and TS by sending sequentially 3 samples to the LS and 2 to the TS, resulting in 1186 samples in the LS and 790 in the TS. This data set took 4 hours and 57 minutes to be generated in a Pentium II Processor at 64 MB RAM, which gives about 9 seconds for the generation of each sample. Regarding the security boundaries presented in Section 4.2.2.3, the obtained number of "insecure" and "secure" OPs in the LS and TS are presented in Table 4.2 and Table 4.3.

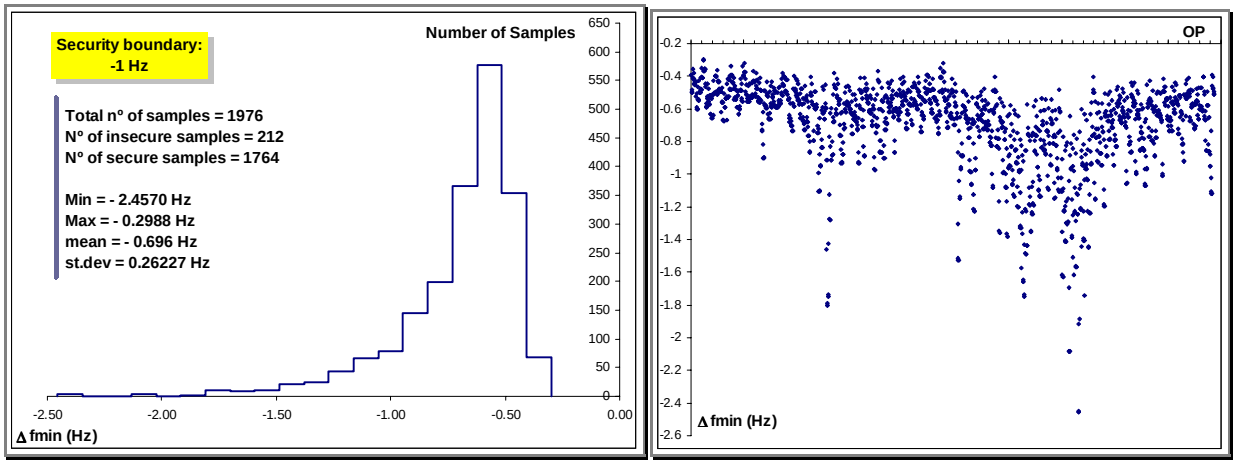
Table 4.2 – Number of "insecure" and "secure" OPs in the LS of Terceira

Disturbance	Short-Circuit with Wind Power Loss		
Security Index	$B_1 = \Delta f_{min}$	$B_2 = \Delta f_{max}$	$B_3 = df/dt_{max}$
Nº of Insecure OPs	125	35	135
Nº of Secure OPs	1061	1151	1051

Table 4.3 – Number of "insecure" and "secure" OPs in the TS of Terceira

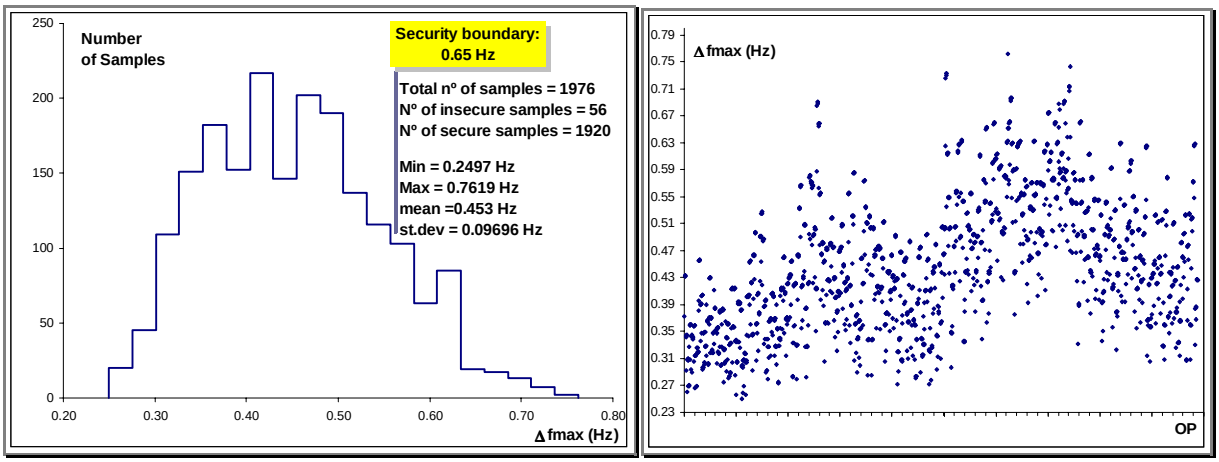
Disturbance	Short-Circuit with Wind Power Loss		
Security Index	$B_1 = \Delta f_{min}$	$B_2 = \Delta f_{max}$	$B_3 = df/dt_{max}$
Nº of Insecure OPs	87	21	93
Nº of Secure OPs	703	769	697

The frequency distributions (histograms) of the 3 security indices Δf_{min}, Δf_{max}, and df/dt_{max} are presented in Figure 4.12(a), Figure 4.13(a) and Figure 4.14(a). The obtained values of the security indices for each OP of the DS are presented in Figure 4.12(b), Figure 4.13(b) and Figure 4.14(b). In these last three figures, each point represents an operating point of the DS.



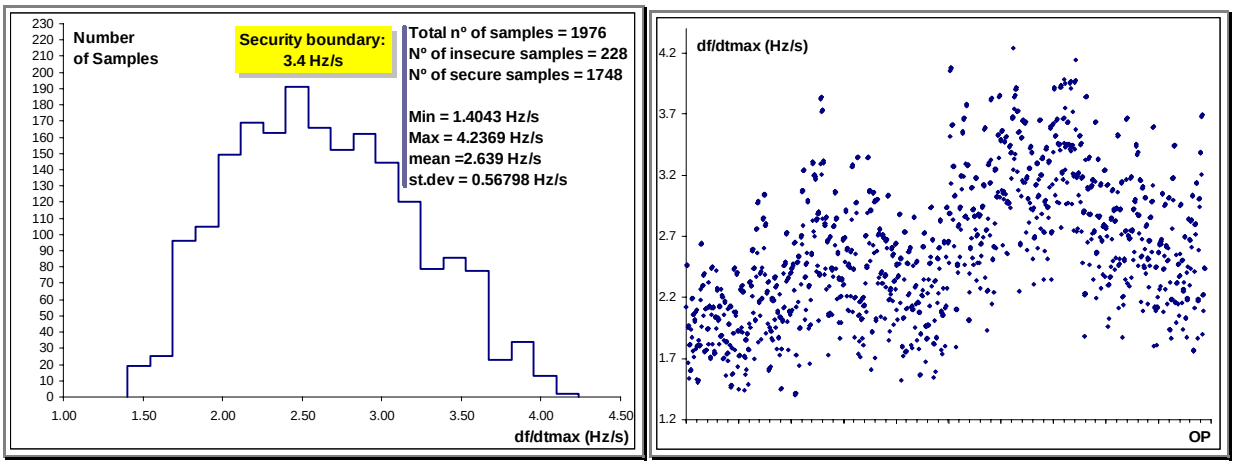
(a) (b)

Figure 4.12 – Histogram of B_1 security index / Value of B_1 for each OP – Terceira DS



(a) (b)

Figure 4.13 – Histogram of B_2 security index / Value of B_2 for each OP – Terceira DS



(a) (b)

Figure 4.14 – Histogram of B_3 security index / Value of B_3 for each OP – Terceira DS

As it can be seen in the previously presented figures, the severity of the considered disturbance is much more observed in the Δf_{min} and df/dt_{max} security indices than in the Δf_{max} security index. However, the OPs of the DS that resulted to be "insecure" according to Δf_{max} are "secure" according to Δf_{min}, Therefore, the security assessment of Δf_{max} was also considered necessary to be performed.

4.3 Data Set of Crete

4.3.1 Crete Power System

The power system of the island of Crete is the largest autonomous power system in Greece. It comprises two power stations, one in Linoperamata and the other in Hania, equipped with Diesel engines, gas and steam turbines. The data set generated for the Crete power system was projected for the year 2000. Figure 4.15 describes the main topological configuration of the network foreseen to the year 2000. In this network all the HV transmission lines operate at 150 kV. For this scenario, a peak load of approximately 360 MW was considered. The annual peak load demand occurs on a summer day, and overnight loads can be assumed to be approximately equal to 25% of the corresponding daily peak loads.

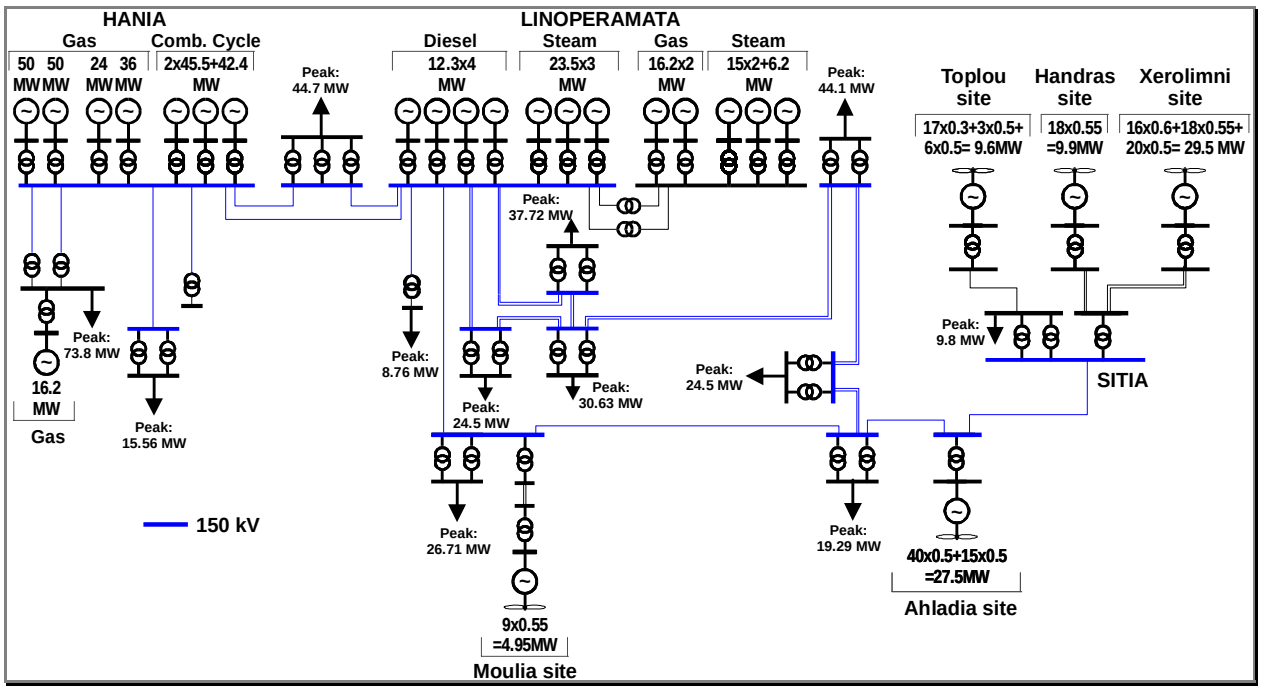


Figure 4.15 – Single line diagram of the Crete generation and transmission system in the year 2000

The conventional generation system consists on two major power plants with twenty generating units installed. These are 6 steam units of total capacity 106.7 MW, 4 Diesel units with 49.2 MW, 7 gas turbines with 208.6 MW and one combined cycle plant with 133.4 MW. The plants are located near to the major load points. The steam and Diesel units mainly supply the base-load. The gas turbine units normally supply the peak load at a high running cost, which increases significantly the average cost of the electricity production.

A total of 10 wind parks, consisting on 162 wind turbines with an installed capacity of 81.45 MW, are or will be installed (have been approved) in Crete by the year 2000. All the generators of the wind turbines will be asynchronous ones, except for those that will be installed in Ahladia site that will be synchronous. Each wind turbine is equipped with induction generator comprising power factor correction capacitors.

4.3.2 STEP 1: Identification of the Security Problem for Crete

Extensive dynamic simulations on the power system model have been performed by NTUA using EUROSTAG software. These simulations showed that for the most common wind power variations, the system remains satisfactorily stable if sufficient spinning reserve is provided. On the other hand, for various short-circuits and conventional unit outages, the system frequency undergoes fast changes and might reach very low values.

In any case, the dynamic security of the system depends critically on the amount of spinning reserve provided by the conventional machines and the response of their speed governors. As an example of this behavior, Figure 4.16 shows the change of the system frequency that follows from the disconnection of three wind parks to a scenario of high wind power penetration. Two different dispatch schemes were considered:

1) Operation with fast thermal units, such as gas turbines and Diesel machines to provide spinning reserve (fast spinning reserve).

2) Operation with slower machines, such as the steam turbines to cover mainly the spinning reserve plus some Diesel machines (slow spinning reserve).

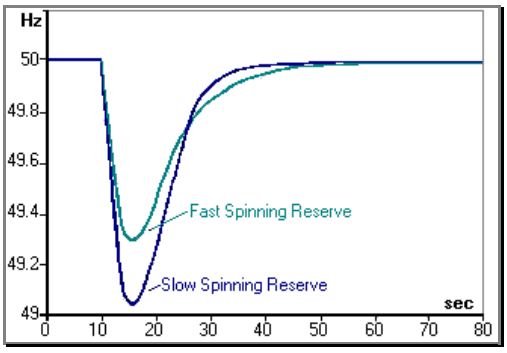


Figure 4.16 – Frequency change to the disconnection of three wind parks

For the simulated situation presented in Figure 4.16, when fast spinning reserve is provided, the lower value of the frequency reaches 49.31 Hz. However, in the case where only slow spinning reserve is available, the lower frequency value, which reaches 49.04 Hz, will cause the operation of the protection devices of the rest of the wind parks. The total wind power disconnection might lead the system to collapse and certainly will trigger the load shedding mechanisms that will disconnect part of the system load.

4.3.2.1 Disturbances Selection

For each one of the produced OPs, a number of possible disturbances has been simulated, where EUROSTAG was used to obtain the system dynamic behavior. The following two major disturbances were selected:

- *Machine Loss:* outage of a major gas turbine;
- *Short-Circuit:* three-phase short-circuit at a critical bus near the wind parks.

These disturbances were selected according to utility criterion. In fact, a unit disconnection is an event that occurs frequently in the Crete power system, whereas a tree-phase fault, although rare, is a severe event that can occur during stormy conditions.

4.3.2.2 Security Indices Selection

For each OP, the following two security indices were recorded:

- $B_1 = f_{min}$: minimum value reached by the system frequency (Hz);
- $B_2 = df/dt_{max}$: maximum value reached by the rate of frequency change (Hz/s).

4.3.2.3 Security Boundaries Selection

To classify an OP as "secure/insecure", the following security boundaries were used:

- $B_{01} = 49\ Hz$ (according to f_{min}, a OP is "secure" if $f_{min}(OP) > 49\ Hz$, otherwise is "insecure");
- $B_{02} = 0.4\ Hz/s$ (according to df/dt_{max}, a OP is "secure" if $df/dt_{max}(OP) < 0.4Hz/s$, otherwise is "insecure").

4.3.2.4 Candidate Attributes Selection

In the selection of candidate attributes for the Crete data set, only pre-disturbances steady-state continuous parameters were considered. For the vector of candidate attributes that characterizes each OP, 22 operating parameters were selected, which include:

- Total active and reactive load: $\sum P_L$ *and* $\sum Q_L$;
- Total conventional active generation: $\sum P_C$;
- Active and reactive power in the wind parks, being considered 4 aggregated wind parks: $P_{W\,i}$, $\sum P_W$, $\sum Q_W$;
- Spinning reserve and active generation in the conventional power plants, being considered 4 aggregated power plants: SR_i and P_{Ci} ;
- Total reactive generation in the capacitor banks: $\sum Q_{cap}$;
- Wind power penetration: $WP = \sum P_W / \sum P_L$;
- Wind margin: $WM = \sum SR / \sum P_W$.

4.3.3 STEP 2: Data Set Generation Method Applied for Crete

For the creation of the Crete data set, different OPs were obtained by varying:

- the load in each one of the 11 load busbars;
- the wind power in each one of the 4 aggregate wind park;
- and the wind margin.

In the generation procedure developed by NTUA, the following steps were applied:

1. The load level, wind power and wind margin were assumed to follow a normal distribution around three operating profiles:

   - Low-load operating condition with a total load $P_L = 100\ MW$;
   - Medium-load operating condition with a total load $P_L = 180\ MW$;
   - High-load operating condition with a total load $P_L = 280\ MW$.

2. For each one of the 11 load busbars and each one of the 4 aggregated wind parks, a perturbation of approximately *±10%* was applied around each one of the above operating profiles.

3. A dispatch algorithm approximating actual operating practices followed in the control center of Crete was applied, using several wind margins.

4. Finally, for each defined *operating point*, both measurement vector and security indices were provided by solving the power-flow solution and by using EUROSTAG to make the dynamic analysis.

4.3.4 Data Set Results for Crete

Using the approach described in this Section, 2765 acceptable samples were obtained. This DS was divided into the LS and TS by sending sequentially 2 samples to the LS and 1 to the TS, resulting in 1844 samples in the LS and 921 in the TS. Regarding the security boundaries presented in Section 4.3.2.3, the obtained number of "insecure" and "secure" OPs in the LS and TS are presented in Table 4.4 and Table 4.5.

Table 4.4 – Number of "insecure" and "secure" OPs in the LS of Crete

Disturbance	Machine Loss		Short-Circuit	
Security Index	$B_1 = f_{min}$	$B_2 = df/dt_{max}$	$B_3 = f_{min}$	$B_4 = df/dt_{max}$
Nº of Insecure OPs	31	59	768	698
Nº of Secure OPs	1813	1785	1076	1146

Table 4.5 – Number of "insecure" and "secure" OPs in the TS of Crete

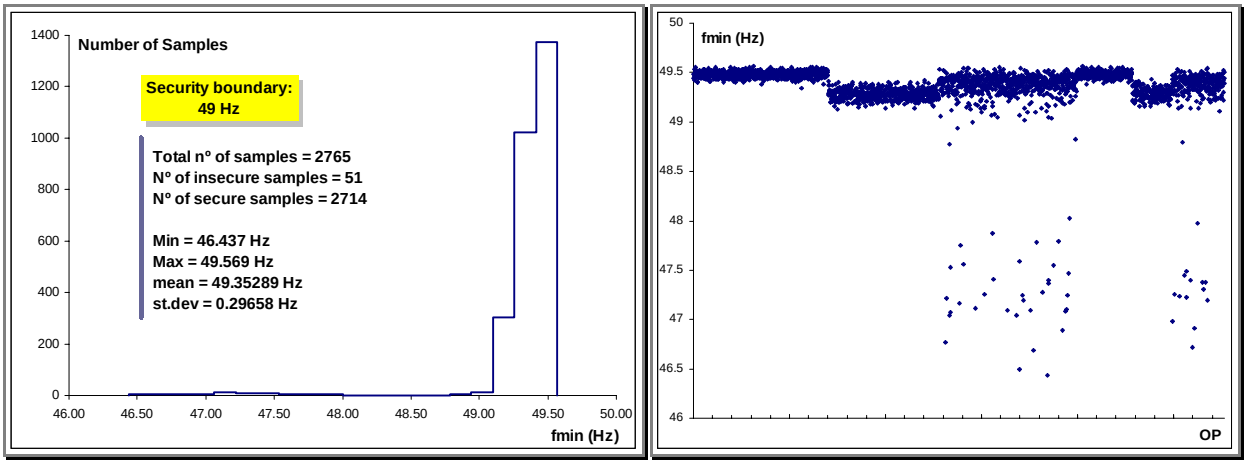
Disturbance	Machine Loss		Short-Circuit	
Security Index	$B_1 = f_{min}$	$B_2 = df/dt_{max}$	$B_3 = f_{min}$	$B_4 = df/dt_{max}$
Nº of Insecure OPs	20	32	375	354
Nº of Secure OPs	901	889	546	567

The frequency distributions (histograms) of the DS security indices, are presented in:

- Figure 4.17(a) for B_1 (f_{min} resulting from machine loss);
- Figure 4.18(a) for B_2 (df/dt_{max} resulting from machine loss);
- Figure 4.19(a) for B_3 (f_{min} resulting from short-circuit);
- Figure 4.20(a) for B_4 (df/dt_{max} resulting from short-circuit).

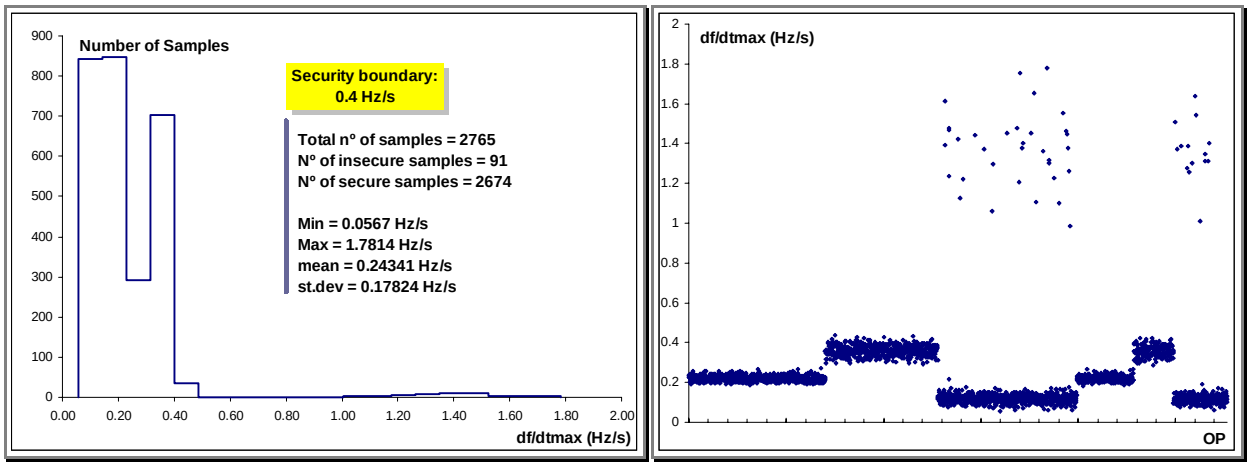
The obtained values of the security indices for each OP of the DS, are presented in:

- Figure 4.17(b) for B_1 (f_{min} resulting from machine loss);
- Figure 4.18(b) for B_2 (df/dt_{max} resulting from machine loss);
- Figure 4.19(b) for B_3 (f_{min} resulting from short-circuit);
- Figure 4.20(b) for B_4 (df/dt_{max} resulting from short-circuit).



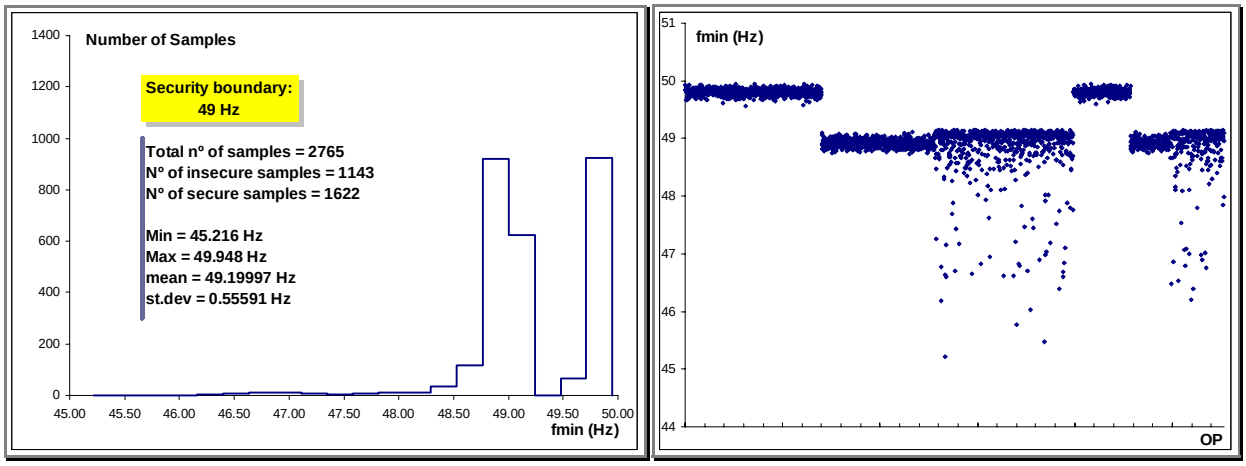
(a) (b)

Figure 4.17 – Histogram of B_1 security index (f_{min}, Machine Loss) / Value of B_1 for each OP – Crete DS



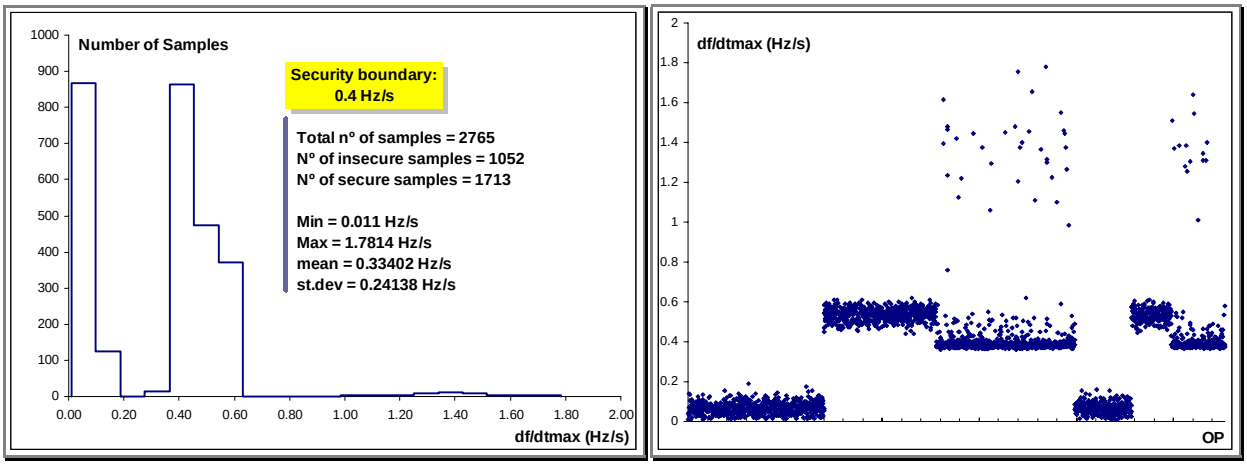
(a) (b)

Figure 4.18 – Histogram of B_2 security index (df/dt_{max}, Machine Loss) / Value of B_2 for each OP – Crete DS



(a) (b)

Figure 4.19 – Histogram of B_3 Security Index (f_{min}, Short-Circuit) / Value of B_3 for each OP – Crete DS



(a) (b)

Figure 4.20 – Histogram of B_4 security index (df/dt_{max}, Short-Circuit) / Value of B_4 for each OP – Crete DS

4.4 Conclusions

With the data sets generated for the Terceira and Crete islands, whose procedure is described in this Chapter, it was possible to derive security assessment structures to those power systems, by applying the Hybrid Regression Tree, Artificial Neural Network and Decision Tree methods. The obtained results with these automatic learning approaches are presented in Chapter 6.

As it is usual to occur, the procedure applied to generate the data set of Terceira required a high computational effort. In fact, the data set generation procedure is an off-line procedure that concerns with the generation of a large number of pre-analyzed security scenarios of the power system to study, which is performed by running analytical tools that simulate the system behavior.

This generation procedure was designed in such a way that, through it, is possible to include the usual system operating strategy, the breadth of the system operating range and with a good resolution. In fact, this technique provided that the OPs are well distributed and highly resolved throughout the pre-defined operating range. The developed software tool provides a general methodology to generate data sets for Diesel-wind isolated power systems.

5 Hybrid Regression Trees

5.1 Introduction

The aim of this Chapter is to give a technical description of a new hybrid automatic learning technique, called in this document as Kernel Regression Trees, implemented to make, for the first time, fast dynamic security assessment (DSA) of power system in the field of frequency stability problems. The results of the application of this algorithm to Crete can be found in [1] and [2].

The Kernel Regression Tree (KRT) is an hybrid method, presented by Luís Torgo [3] in 1997, which integrates Regression Trees (RTs) ([4] – Breiman et al., CART, 1984) with kernel regression ([5] – Watson, 1964; [6] –Nadaraya, 1964).

The first application of the RT approach in DSA, used in the field of voltage stability problems, is due to Wehenkel [42], in 1995. Recently, an application of a KRT approach in the same security assessment problem was presented in [43] by Peças Lopes et al..

The RT is a non-parametric statistical methodology that deals with continuous goal variables (i.e., consists on a method to solve regression problems). Thus, the output of a RT security structure is a security index B that measures the security degree of a hypothetical OP to a pre-defined disturbance. The RT consists on a machine learning (ML) method. Thus it provides security structures that can be translated into interpretable security rules. An overview to the security structure provided by this method is presented in Section 3.5.

Kernel regression is also a non-parametric statistical methodology that belongs to a research field usually called local modeling [3]. Kernel regression models provide quite opaque security structures, but on the other hand, are able to model with good accuracy non-linear functions. Thus, by integrating this regression procedures in the tree leafs, a security structure with better accuracy can be obtained, by increasing the non-linearity of the functions used at the leafs [3]. Furthermore, in highly non-linear problems, by integrating kernel regression models in the tree leafs, it is possible to overcome the limitations of the individual kernel regression model, both in terms of accuracy and computational efficiency [53].

In the work of Luís Torgo presented in [53], the performance of several functional models was studied to make prediction on the RT leafs, to several artificial and real life domains. These studied alternatives included:

- Parametric models, namely, the traditional mean value and linear regression;
- Non-parametric models, namely, K-nearest neighbors and kernel regression.

From those experiences it was concluded that, in highly non-linear problems[11] the use of the two non-parametric models in the tree leafs clearly give better accuracy, whereas kernel regression proved to be the best predicting function. On the other hand, these two non-parametric approaches also proved to be computationally more expensive. This is natural to occur since, for each prediction, these models need to re-calculate the regression model, whereas by using the mean value or the linear regression, the prediction is assigned by a constant model.

In this Chapter, the implemented Hybrid Regression Tree (HRT) algorithm is described. It provides tree structures that can be of the following main types:

- a RT security structures, by considering the mean value as the model to use at the tree leafs;
- a KRT security structure, by considering a kernel regression as the model to use at the tree leafs.

5.2 Design of a Hybrid Regression Tree

In the implemented HRT algorithm, the method used to design the tree structure of a RT and a KRT structure is exactly the same. In fact, to extract a KRT security structure, the following two stages are performed:

- Design of the regression tree (RT) structure;
- Assign a kernel regression model to make prediction in the tree leafs.

Two approaches were considered to design the RT, which differ in the way applied to avoid overfitting problems. The first one, which is described in Section 5.2.1, fights overfitting by applying directly stop-splitting rules during the growing algorithm of the RT. This first technique, presented by Breiman et al. in CART [4], although avoiding the tree to grow until having only pure leafs, does not look for the right sized tree. In fact, much work was made centered on finding the appropriate stop-splitting rules for generating the tree with the right size (i.e., with a good trade-off between *bias* and *variance*), where many variants were invented and tested. From this work it was concluded that searching for the right stopping rule was the wrong way of looking at the problem. A more satisfactory procedure was found that consists on pruning instead of stopping. For this reason subsequently another more efficient technique was applied, which is a pruning algorithm based in the one presented in CART [4] and that is described in Section 5.2.3.

[11] which is characteristic to occur in power systems security problems

5.2.1 Design of a Regression Tree Using Stop-Splitting Rules

In this approach, the design of a RT is determined by the following two issues:

- the optimal splitting test;
- the stop-splitting rules.

5.2.1.1 Optimal Splitting test

Starting with the root node, which corresponds to the LS, the growing of the RT is made by successively splitting their nodes. In the implemented algorithm, it was considered that all candidate attributes are numerical values. In those cases the splitting of a node is performed by a test applied in the measurement hyperspace A defined as:

$$\{a_k(sample) > u_k\}? \tag{5.1}$$

where:

$a_k(sample)$: *value of candidate attribute k in the sample*
u_k : *optimal threshold value of the chosen candidate attribute* a_k

By applying this test to all the samples in the node, two successor nodes are created, which correspond to the two possible instances of the test:

$$\{a_k(sample) > u_k\} \text{ and } \{a_k(sample) \le u_k\} \tag{5.2}$$

When dividing a node, the optimal splitting test has to be selected from the samples stored in the node. In this procedure, for each candidate attribute, the set of candidate splitting tests is defined by the halfway value between consecutive distinct values and by the lower value of the attribute range.

The splitting of each node must be performed according to an *optimal splitting test*, which corresponds to the one that provides a maximum amount of information. As already referred in Section 3.5, the design of a RT consists on explain as much as possible the mean squared error of the security index B observed in the LS. This corresponds to dividing the samples of the LS into disjoint regions, in such a way that in each region the security index B is as constant as possible, being this partition defined by the leafs of the designed tree.

According to this goal, the optimal splitting test *"s"*, which divides a node *"t"* into *"t_L"* and *"t_R"*, is the one that most decreases the learning error estimator $MSE(RT)^{LS}$ (see equation 3.13), i.e., that maximizes:

$$\Delta R(s,t) = R(t) - R(t_L) - R(t_R) \tag{5.3}$$

where:

$$R(t) = \frac{1}{N(LS)} \sum_{OP_i \in t} (B_i - f_t(OP_i))^2$$

Note that

$$MSE(RT)^{LS} = \sum_{t \in \tilde{T}} R(t) \qquad (5.4)$$

The error measure $R(t)$ as the property that for any split of *"t"* into *"t_L"* and *"t_R"* then:

$$R(t) \geq R(t_L) + R(t_R) \qquad (5.5)$$

As already referred, in the implemented algorithm, the mean value was considered as the predicting function to be used in the leafs of the RT. Therefore, the optimal splitting test was selected as being the one that most decreases the RT variance $s^2(RT)^{LS}$ (see equation 3.5). According to this goal, at each node *"t"*, the selected splitting test *"s"* was the one that maximizes:

$$\Delta s^2(s,t) = s^2(t) - P_L \times s^2(t_L) - P_R \times s^2(t_R) \qquad (5.6)$$

where:

$s^2(t)$ *: variance of B formed by the learning samples stored in node t (see equation 3.4)*
P_L *and* P_R *: proportion of learning samples at the left and right successor nodes*
$s^2(t_L)$ *and* $s^2(t_R)$ *: variance of B at the left and right successor nodes*

These splitting rules are described in CART [4].

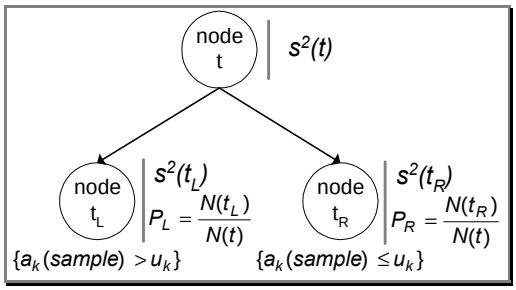


Figure 5.1 – Splitting a node *t* of a *RT*

5.2.1.2 Stop-Splitting Rule

The procedure continues splitting the created successor nodes, until a stop-splitting criterion is met for all the non-splitted nodes. The criterion used, which was described by Luís Torgo, is defined by the two stop-splitting rules:

- Rule 1: It is not possible to further reduce the variance of B in a statistically significant way. This corresponds to verify if a minimum number of examples, N_{min}, has been reached in the node [53][54].

- Rule 2: The variance of B has been sufficiently reduced. This corresponds to verify if a minimum value of a statistic call *coefficient of variation*, CV_{min}, has been reached in the node [53], where:

$$CV(node) = \frac{\sqrt{s^2(node)}}{\overline{B}_{node}} \tag{5.7}$$

$CV(node)$ captures the spread of the set of B values in the node. As an alternative, it can be verified if a minimum value of the variance of B, s^2_{min}, as been reached in the node, where s^2_{min} corresponds to a perceptual value of the variance of B in the root node [54], i.e.:

$$s^2_{min} = \frac{percentual\ value}{100}\% \times s^2(root) \tag{5.8}$$

This procedure captures the spread of the set of B value in the node relatively to the root node.

When, in a node, one of these rules is verified, it becomes a terminal node, i.e., a leaf.

In the literature, there are still other alternatives to stop splitting a regression tree. Namely, in CART [4] it is referred that, a node t can be declared as terminal if:

$$\max_s \Delta R(s,t) \leq 0.006 \times R(root) \tag{5.9}$$

where "s" is a splitting test

This procedure captures the capability of the splitting tests to reduce variance.

5.2.2 Predicting, with Kernel Regression Models in the Tree Leafs

Once the design of the RT, to obtain a KRT structure, a kernel regression model is assigned to make prediction at the tree leafs. According to the used kernel regression model, a prediction of the security index B for any new unseen operating point Q of the system (a query point) is obtained by performing the following steps:

1) Find the leaf that verifies the Q operating conditions. At this stage Q becomes within the measurement hyperspace A, defined by the learning samples stored in that leaf (see example presented in Figure 5.2 for a three-dimension measurement hyperspace).

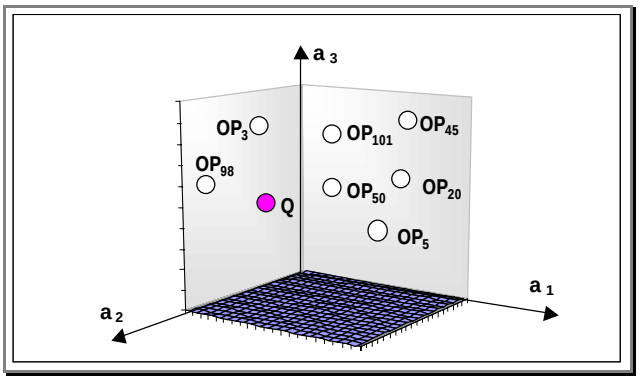


Figure 5.2 – Example of a new unseen operating point *Q* in the measurement hyperspace *A* of a leaf

2) Predict the numerical value of *B(Q)*, i.e. *B'(Q)*, by applying a kernel regression model to the learning samples stored in the leaf. Kernel regression models [5][6] make prediction by a weighted mean of the response *B* of the form:

$$B'(Q) = \frac{\sum_{i=1}^{neighbors} K_h[D(Q,OP_i)] \times B_i}{\sum_{i=1}^{neighbors} K_h[D(Q,OP_i)]} \qquad (5.10)$$

where:

D(Q,OP) : Distance function
h : Bandwidth value
$K_h[x] = K\left[\frac{x}{h}\right]$, *being K(.) the Kernel function*
OP_i *- Operating Point of neighbor* $_i$
B_i *- Security index value of neighbor* $_i$

Distance Function

The prediction is obtained by using the samples in the leaf (also named *neighbors*) that are "most similar" to *Q*. This similarity is measured by means of the *distance function D(Q,OP)*. This function measures the normalized distance between two samples in the measurement hyperspace *A*.

One design issue of kernel regression models includes the choice of the distance function. In the implemented model, an Euclidean distance was used, which is defined as:

$$D(Q,OP) = \sqrt{\sum_{i=1}^{Na} d_i(Q,OP)^2} \qquad (5.11)$$

where:

$d_i(Q,OP) = \dfrac{|a_i(Q)-a_i(OP)|}{M_{diff}}$: *Normalized difference between the values of candidate attribute* a_i

$d_i(Q,OP) \in [0,1] \rightarrow \Re^+$

M_{diff} : *Maximal value among the set of* $|a_i(Q)-a_i(OP)|$ *differences*

$D(Q,OP)$: *Normalized distance* $\in [0,1] \rightarrow \Re^+$

Na : *Number of candidate attributes*

Bandwidth Value

Another important design decision, when applying kernel regression models, is the choice of the *bandwidth value h*, where many alternatives exist [55]. In the implemented model, a K-Nearest Neighbor (KNN) rule[12] was used. The KNN rule sets the bandwidth value *h* as the distance *D* to the K-nearest neighbor of *Q*. It also sets that only the K-nearest neighbors are used to make prediction.

Kernel Function

The kernel function *K(.)* (also called weight function) estimates the weight of each neighbor to *Q*, giving more weight to neighbors that are nearest to *Q*. In the implemented model, a Gaussian one was applied, which is presented in Figure 5.3 and given by:

$$K(d) = e^{-d^2} \qquad (5.12)$$

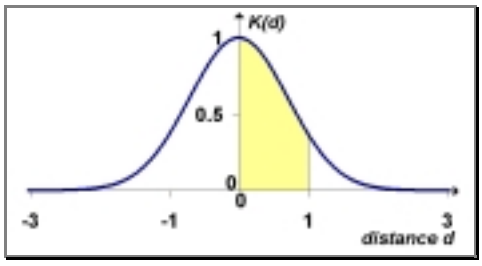


Figure 5.3 – Kernel function

Atkeson et al. [56] claims that the choice of the kernel function is not a critical design issue, as long as the function is reasonably smooth. These authors provide an extensive list of alternative kernel functions and discuss some of their merits. Kernel regression (and generally local modeling) can be very sensitive to the presence of irrelevant features, so weighing can help to reduce this influence [3].

[12] See Section 3.6 where the application of K-Nearest Neighbor rule in classification problems is synthetically explained.

5.2.3 Design of a Regression Tree/Kernel Regression Tree by Applying a Pruning Algorithm

The implemented pruning algorithm, applied to design a RT or a KRT structure, comprises the following stages:

1) Design of a very large regression tree, $RT_{\max}$, which is supposed to overfit the LS.

2) Generation of a sequence of pruned trees with decreasing complexity, $T_1 \succ T_2 \succ \ldots \succ root$ where $T_1 \prec = T_{\max}$, by progressively pruning $T_{\max}$ upward in the "right way" until being reached the root node. Note that a subtree T_i of T is referred as a pruned tree of T if $root(T_i) = root(T)$, which can be denoted by $T \succ T_i$.

3) Selection, among that sequence of generated pruned trees $\{T\} = \{T_1, T_2, \ldots, root\}$, the right sized one, based on an accurate estimation of the true predicting error of the corresponding security structures, i.e., by using a testing set to estimate the structures performances.

5.2.3.1 Design of RTmax

To grow $RT_{\max}$, the previously described design procedure that exploits stop-splitting rules is applied. By using this process, a very large tree is designed by letting the splitting procedure continue until each terminal node is sufficient small or contains only identical B values. The size of this initial tree is not critical as long as it is large enough to overfitt the LS. For all the results presented in Chapter 6, the following stop-splitting rules were used:

- rule 1: The number of examples in the node is 1 (i.e., $N(node) = 1$);
- rule 2: The variance of B in the node is 0 (i.e., $s^2(node) = 0$).

5.2.3.2 Pruning Process

Minimum Error-Complexity Criterion

Even for a moderate sized $RT_{\max}$, there is an extremely large number of pruned trees of $T_{\max}$ and an even larger number of distinct ways of pruning up it to the root node. Regarding this, a selective pruning process is applied, that generates a reasonable number of pruned trees of $T_{\max}$, with decreasing size, such that each subtree is the "best" pruned tree in its size range.

To select the "best" pruned tree in its size range, a *minimum error-complexity criterion* is applied to each last pruned tree. Considering that T is the binary tree structure of a regression tree RT, the *error-complexity measu*re of RT is defined by:

$$MSE_\alpha(RT)^{LS} = MSE(RT)^{LS} + \alpha \times \left|\tilde{T}\right| \qquad (5.13)$$

where:

> $MSE(RT)^{LS}$: *learning estimation of the RT error (see equation 3.13)*
> $\left|\tilde{T}\right|$: *number of leafs in T (complexity of RT)*
> α : *real number ≥ 0 (penalty for the complexity of RT)*

Thus, $MSE_\alpha(RT)^{LS}$ is formed by adding to the error of the *RT*, a cost penalty of its complexity.

One more time, the mean value was considered as the predicting function to use in the *RT* leafs. Thus, to calculate the error $MSE(RT)^{LS}$ of the *RT*, the variance measure $s^2(RT)^{LS}$ was used (see equation 3.5).

Starting with $\alpha = 0$, for each increasing value of α the pruning algorithm finds the subtree $T_\alpha \prec T_{\max}$ that minimizes $MSE_\alpha(RT)^{LS}$, i.e.:

$$MSE_\alpha(RT_\alpha)^{LS} = \min_{T \prec T_{\max}} MSE_\alpha(RT)^{LS} \qquad (5.14)$$

Note that when α is a small value, the penalty of having a large number of leafs is low and therefore it will result on a large T_α.

Although α runs through a continuous value, the pruning process produces a finite sequence of pruned trees $T_1, T_2, \ldots, root$ with progressively fewer terminal nodes. This is because each T_α is the *minimizer* subtree of $MSE_\alpha(RT)^{LS}$ for a range of values of α, and therefore as α increases it continues being the *minimizer* subtree until a jump point α' is reached, where a new smaller subtree $T_{\alpha'}$ becomes the *minimizer*. The pruning process stops when the *minimizer* subtree becomes the root node of $T_{\max}$.

A direct search through all possible subtrees $T_\alpha \prec T_{\max}$ to find the *minimizer* of $MSE_\alpha(RT)^{LS}$ is computationally expensive. Moreover, there are no guaranties that there is a unique subtree $T \prec T_{\max}$ that minimizes $MSE_\alpha(RT)^{LS}$, and that in the pruning process the nesting $T_1 \succ T_2 \succ \ldots \succ \{root\}$ holds.

To overcome this difficulties and obtain and effective algorithm for generating the sequence of subtrees, the pruning process uses the following procedures:

- for each α value, the pruning algorithm finds the smallest subtree $T_\alpha \prec T_{\max}$ that minimizes $MSE_\alpha(RT)^{LS}$;
- each subtree of $T_{\max}$ is obtained by pruning upwards from the previous subtree.

Thus, for each value of α, the pruning algorithm looks for the *smallest minimizing* subtree $T_\alpha \prec T_{max}$, which is defined by the following conditions:

$$\begin{aligned} &(i)\ \ MSE_\alpha(RT_\alpha)^{LS} = \min_{T \prec T_{max}} MSE_\alpha(RT)^{LS} \\ &(ii)\ \ If\ MSE_\alpha(RT_\alpha)^{LS} = MSE_\alpha(RT)^{LS}, then\ T_\alpha \prec T \end{aligned} \qquad (5.15)$$

According to this previous definition, for every value of α there is a *smallest minimizing* subtree $T(\alpha) \prec T_{max}$, which is unique (the proof of this proposition can be found in CART [4]).

Generation of T_1

T_1 is the smallest pruned tree of T_{max} that minimizes $MSE_\alpha(RT)^{LS}$ for $\alpha = \alpha_1 = 0$, i.e., T_1 is the smallest pruned tree of T_{max} that satisfies:

$$MSE(RT)^{LS} = MSE(RT_{max})^{LS} \qquad (5.16)$$

Letting "t_L" and "t_R" to be any two successor nodes resulting from the split of a node "t" of a tree, then as previously said in Section 5.2.1.1:

$$R(t) \geq R(t_L) + R(t_R) \qquad (5.17)$$

Regarding this, T_1 is gotten from T_{max} by pruning all the pair of leafs $\{t_L, t_R\}$ that verify $R(t) = R(t_L) + R(t_R)$, where in the implemented approach the predicting function $f_t(OP_i)$ was considered to be the mean value.

In the resulting T_1, for any non-terminal node t:

$$R(t) > MSE(RT_t)^{LS} \qquad (5.18)$$

where:

$$MSE(RT_t)^{LS} = \sum_{t \in \tilde{T}_t} R(t)$$

Generation of T_2, ..., root node

Starting with T_1, the remaining set of pruned trees of T_{max}, $\{T_2, T_3, \ldots, root\}$, is obtained by progressively replacing the *weakest non-terminal node* by a leaf.

Setting

$$MSE_\alpha(RT_t)^{LS} = MSE(RT_t)^{LS} + \alpha \times \left|\tilde{T}_t\right| \qquad (5.19)$$

and

$$R_\alpha(t) = R(t) + \alpha \text{ : error-complexity measure of node } t \tag{5.20}$$

For every non-terminal node t of T_1, as long as $R_\alpha(t) > MSE_\alpha(RT_t)^{LS}$ is preferable to have the subtree T_t in T_1 instead of replacing the non-terminal node t by a leaf. However, at some critical α_2 value of α, for some non-terminal node t_2 happens that $R_\alpha(t_2) = MSE_\alpha(RT_{t2})^{LS}$. At this point, t_2 is smaller than T_{t2} having the same error-complexity, and therefore is preferable. Thus, for $\alpha = \alpha_2$, t_2 is the weakest non-terminal node of T_1. At this point:

$$\alpha = \alpha_2 = \frac{R(t_2) - MSE(RT_{t2})^{LS}}{\left|\tilde{T}_{t2}\right| - 1} \tag{5.21}$$

To find automatically α_2 and t_2, let us define a function $g(t)$ for all non-terminal nodes of T_1 as:

$$g(t) = \frac{R(t) - MSE(RT_t)^{LS}}{\left|\tilde{T}_t\right| - 1} \tag{5.22}$$

Regarding that for $\alpha = \alpha_2$ then $\alpha_2 = g(t_2)$ and $\alpha_2 < g(t_i)\ \forall\, t_i \in \left\{T_1 - \tilde{T}_1 - t_2\right\}$, the weakest non-terminal node t_2 of T_1 is the one that minimizes $g(t)$, i.e., that verifies:

$$g(t_2) = \min_{t \in \left\{T_1 - \tilde{T}_1\right\}} g(t) \tag{5.23}$$

being

$$\alpha_2 = g(t_2)$$

Having detected t_2, then T_2 is generated by replacing, in T_1, t_2 by a leaf, i.e.:

$$T_2 = T_1 - T_{t2} \tag{5.24}$$

Continuing with this process, a decreasing sequence of subtrees $T_1 \succ T_2 \succ \ldots \succ root$ is obtained.

Summarizing:

The result of the implemented pruning process is a decreasing sequence of binary trees $\{T\} = \{T_1, T_2, \ldots, root\}$ where $T_1 \succ T_2 \succ \ldots \succ root$ with $T_1 \prec = T_{\max}$, and a corresponding increasing sequence of α values $\alpha_1 = 0 < \alpha_2 < \ldots$, such that for $\alpha_k < \alpha < \alpha_{k+1}$, T_k is the smallest pruned tree of $T_{\max}$ that minimizes $MSE_\alpha(RT)^{LS}$. By considering the mean value to make prediction at the tree leafs of $\{T\}$, the result of the pruning process is a set of regression trees $\{RT\} = \{RT_1, RT_2, \ldots, root\}$. By considering the kernel regression described in Section 5.2.2 to make prediction at the tree leafs of $\{T\}$, the result of the pruning process is a set of kernel regression trees $\{KRT\} = \{KRT_1, KRT_2, \ldots, root\}$.

Here is necessary to remark that the sequence of *minimum error-complexity* trees is a subsequence of the one constructed by a *minimum error* criterion, whose pruning process is the following:

- Supposing that $T_{\max}$ has L leafs, for every value of $H: 1 \le H \le L$, the subtree $T_H \prec T_{\max}$ with $L-H$ leafs that minimizes $MSE(RT)^{LS}$ is selected.

Although this *minimum error* pruning procedure is intuitive and possible to be implemented, comparing with the *minimum error-complexity* pruning process, it is computationally much more expensive. Moreover, in regression problems, the sequence of pruned trees that results from the *minimum error-complexity* criterion is usually almost the same as the one that results from the *minimum error* criterion. In fact, the pruning process based in the $MSE_{\alpha}(RT)^{LS}$ error-complexity measure usually takes off only two terminal nodes at a time [4]. For instance, this behavior can be seen in the results presented in Figure 3.6 of Chapter 3 (Section 3.8), which were obtained from an experience carried out with the Terceira data set. From this experience, having T_1 2329 nodes, a set $\{T\}$ with 1141 trees was generated, being the maximum possible number of pruned trees in $\{T\}$ equal to $\left|\widetilde{T}_1\right| + 1 = \left(\left|T_1\right| - 1\right)/2 + 1 = (2329 - 1)/2 + 1 = 1165$.

5.2.3.3 Selection of the Right Sized Tree

According to the control center requirements, the user may want to exploit a specific KRT or a RT structure from the set of pruned trees $\{T\}$.

Note that, for the KRT and RT approaches, the tree structures in $\{T\}$ are exactly the same. The only difference between the two approaches stays in the model used to make prediction in the tree leafs. In the implemented algorithm, to make prediction, a kernel regression model is assigned to the KRT structures, whereas the mean value is assigned to the RT structures.

Among $\{T\}$, the user must select the right sized tree based on an accurate estimation of the true predicting error of the corresponding security structures. Thus, instead of a learning estimation, a testing estimation must be used to evaluate the performances of the design structures. Therefore, according to the control center requirements, the right sized tree must be selected by following the criterions described below.

1) Regarding the cases when ***producing fast security evaluation*** is the main goal, then maximizing accuracy and minimizing prediction time are the main requirements to the user. To maximize accuracy, the right sized tree must be selected among $\{T\}$ according to the minimization of a TS error estimator of the S structure, where S can be a RT or a KRT structure. Within fast security evaluation, the following two situations can occur:

1.a) If the goal is to make on-line evaluation of the system security degree, the $MSE(S)^{TS}$ error estimator (see equation 3.8) can be used to estimate the accuracy of the structure. According to the minimization of this error, the selected structure S_{MMT} is such that:

$$MSE(S_{MMT})^{TS} = \min_{T_i \in \{T\}} MSE(S_i)^{TS} \qquad (5.25)$$

In this document, the *KRT* and *RT* structures selected according to this last criterion are called KRT_{MMT} and RT_{MMT} (*MMT – Minimum MSE Tree*).

1.b) If the goal is to make on-line classification of the system as "secure/insecure", the selected structure must reach a small *Global Classification TS error* (see equation 3.10). Besides this last requirement, two other important issues should be considered: to have small *False Alarm* and *Missed Alarm TS errors* (see equation 3.11 and 3.12). Obviously, missed alarm error is a misclassification rate with higher importance than false alarm error, since they correspond to actually "insecure" OPs for which the structure failed to warn.

2) Regarding the cases where the main goal is to ***extract interpretable security rules*** from the tree structure, both interpretability and accuracy are important to the user. Obviously a constant model must be considered in the tree leafs. Therefore, the use of a *KRT* structure is not feasible since, for each prediction, these structures need to re-calculate a kernel regression function. Thus, among the KRT and RT approaches, only the last one can be used to extract those security rules. Within the extraction of interpretable security rules, the following two situations can occur:

2.a) If the goal is to extract interpretable security rules that explain the *B* value as a function of the system operating conditions (i.e., extract regression rules), it must be chosen a *RT* that achieves a good compromise between the regression errors and complexity. In this case, complexity depends on the number of leafs (since it gives the number of *If rules*) and on their depth to the root node (since it gives the maximum number of attributes that can be included in the *If rule*). In CART [4], to obtain a RT structure with a good trade-off between accuracy and comprehensibility, it is suggested to select the right sized tree according to the *k SE rule* for $k = 1$ (i.e., the *1 SE rule*). By following the *1 SE rule*, the selected regression tree RT_{SET} is the smallest one such that:

$$MSE(RT_{SET})^{TS} \leq MSE(RT_{MMT})^{TS} + k \times SE \qquad (5.26)$$

where:

> *k=1*
>
> *SE is the standard error estimation of* $MSE(RT_{MMT})^{TS}$*, used to define the uncertainties of the* $MSE(RT_{MMT})^{TS}$ *predicting error estimation.*

For any structure S:

$$SE(MSE(S)^{TS}) = \frac{1}{\sqrt{N(TS)}}\left[\frac{1}{N(TS)}\sum_{OP'_i \in TS}\left(B'_i - f_S(OP'_i)\right)^4 - \left(MSE(S)^{TS}\right)^2\right]^{1/2} \quad (5.27)$$

In this document, the RT that verifies the *1 SE rule* is called RT_{SET} *(SET – Standard Error Tree)*. The *1 SE rule* allows choosing the simplest tree whose accuracy is comparable to the one that minimizes $MSE(RT)^{TS}$ (i.e., the RT_{MMT}).

2.b) If the goal is to extract interpretable security rules that explain the *system security class ("secure/insecure")* as a function of the system operating conditions (i.e., extract classification rules), it must be chosen a RT that achieves a good compromise between the classification errors and complexity. In these cases, the number of leafs do not necessarily measure the number of extracted *If rules*. To illustrate, let us suppose that a tree has a large number of leafs, but only two of them are assigned by an "insecure" class. Thus, although being large, this tree can be translated into a very simple set of security rules of the form:

```
If [operating conditions from the root node to the first " insecure" leaf ]
or If [operating conditions from the root node to the second " insecure" leaf ] Then "Insecure"
else "Secure"
```

5.3 Experiments to Compare Performance Between RT and KRT

In order to compare the performance of the implemented regression tree and kernel regression tree approaches, an experiment was carried out for the case study of Terceira island already referred in Section 3.8.1.

The resulting sequence of pruned trees $T_1 \succ T_2 \succ \ldots \succ T_{1140} \succ root$ was evaluated in the sense of predictive regression accuracy and predictive computational efficiency, where two situations were considered:

- the use of a mean value as the function to make prediction in the leafs – RT approach;
- the use of the kernel regression model described in Section 5.2.2 as the function to make prediction in the leafs – KRT approach;

The obtained results are shown in Figure 5.4 in the sense of regression accuracy performance, and in Figure 5.5 in the sense of computational efficiency performance. To estimate the predictive regression accuracy, the $RMSE(S)^{TS}$ error was used. To estimate predictive computational efficiency, the execution time that the program takes to make prediction for all the TS was used.

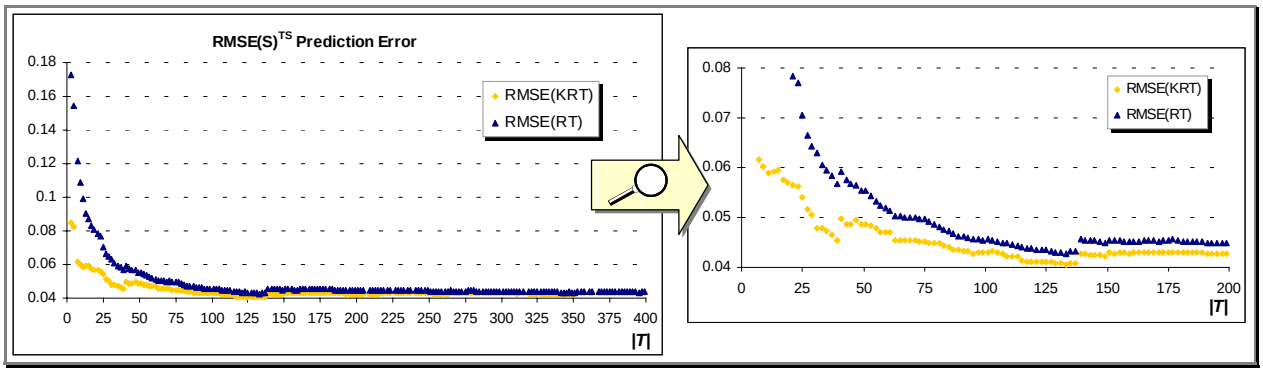


Figure 5.4 – Comparing predictive accuracy between RT and KRT structures

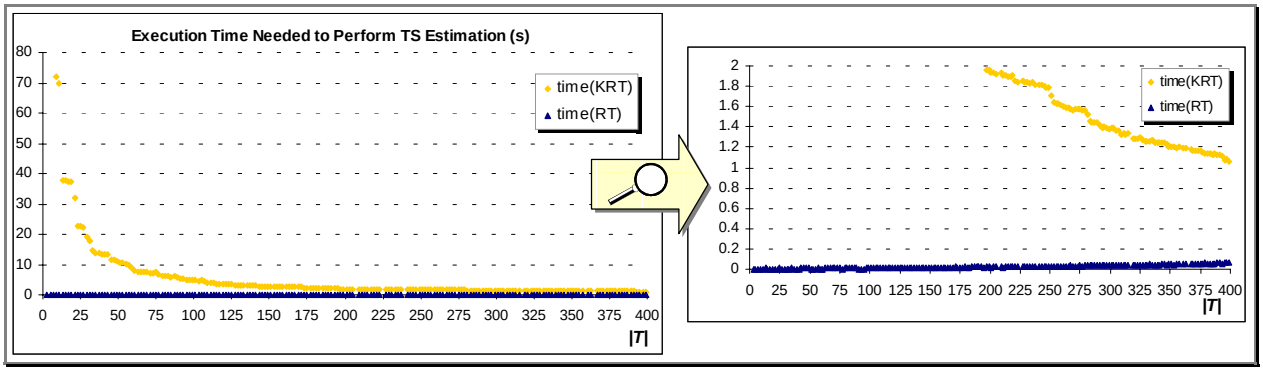


Figure 5.5 – Comparing predictive computational efficiency between RT and KRT structures

Regarding predictive regression accuracy, as it can be seen in Figure 5.4, KRT always obtained the best score.

Regarding predictive computational efficiency, as it can be seen in Figure 5.5, KRT always obtains the higher execution times. This is due to the prediction in KRT being a two-stage process. First, like in RT, the OP is dropped down the tree. Then, instead of assigning the mean value of B as being the predicting value, kernel regression is applied using the learning samples stored in the leaf. The more complex the tree is, the lower is the number of samples stored in the leafs, and therefore, the lower is the price in terms of time needed for local prediction tasks. Thus, as it can be seen in Figure 5.5, as the tree grows in size, the predictive time of KRT gets closer to the one obtained by RT. Also note that the more complex the tree, the higher is the time needed to drop an OP down the tree, being this the reason for the execution time of the RT presented in Figure 5.5, although slowly, increases with its complexity.

5.4 Some Issues Related to Tree Structure Methods

When using the tree structure to extract interpretable rules, it is necessary to have in mind that the fact that a candidate attribute does not appear in any of the tree splits does not mean that it is not relevant for the problem. In fact, a relevant attribute can be constantly masked by another and thus never be chosen for splitting the nodes.

At any given node, there may be a number of splits on different variables, all of them giving almost the same decreasing of the *MSE* learning error. Since data is noisy, the choice between these competing splits may be almost random. If one attribute masks another, then small changes in the DS, or even in how the LS and TS result from randomly separating the DS, may shift the split from one attribute to the other, which will lead to a different evolution of the tree from that node downwards. This behavior, which is remarked in CART [4], leads to the tree structure being unstable. Some examples can be seen in [4], which illustrate tree structure instability, i.e., that small changes in the DS may lead to much different tree structures, however achieving almost the same accuracy.

This issue, which applies both to decision and regression trees, is very important because if it is not considered then the tree structure may lead to misinterpretation.

Another frequently mentioned characteristic, and also referred in [4], of the tree growing procedure is that it is only one-step optimal and not overall optimal. To illustrate this behavior, lets suppose that the tree growing procedure produces 11 terminal nodes. If one could search, within all possible partitions of the learning samples into 11 disjoint groups, for the one that minimizes the *MSE* learning error, the two results might be quite different. However, looking for the optimal tree structure hasn't been an issue to reach by the researchers of this field, since an overall optimal tree growing procedure does not appear computational feasible for any reasonably sized data set. Besides, in the sense of the application to power systems security assessment, the main goal is to construct a good predicting structure whose performance stands up to the ones of other methods and that is useful and practical.

6 Results

6.1 Introduction

In this Chapter, the results obtained with the Hybrid Regression Tree (HRT) method described in Chapter 5, for the cases of the Terceira and Crete island networks, are presented. The available Artificial Neural Network (ANN) and Decision Tree (DT) results are also presented for comparative assessment purposes. The performance evaluation results were obtained by applying the testing set (TS) to the obtained structures.

ANN structures were obtained to deal with this problem within the framework of the EU research project, using software already available. These structures were designed by other researches of the Power Systems Unit of INESC Porto [7]. Researchers of NTUA provided the DT structures. These researchers were the ones involved in the CARE project responsible for developing the DT module, and for constructing the DT structures to be used in the Crete advanced control system. The obtained DTs were derived through the use of an ID3 algorithm [45].

The security problem evaluated for the power system of Terceira is described in Chapter 4. However, just to remind, in Table 6.1 the security indices selected that define the dynamic security of this network are presented. The obtained number of "insecure" and "secure" samples in the TS is presented in Table 6.2.

Table 6.1 – Security indices of the Terceira case

Disturbance		
Short-Circuit + Wind Power Loss: Tree-phase short-circuit in Angra do Heroísmo, eliminated after 180 ms, followed by the disconnection of Santa Bárbara wind park at 100 ms after de occurrence of the default		
Security Indices		**Security Boundary**
B_1	Δf_{min}: minimal value reached by the negative frequency deviation (Hz)	-1Hz
B_2	Δf_{max}: maximal value reached by the positive frequency deviation (Hz)	0.65 Hz
B_3	df/dt_{max}: maximal value reached by the rate of frequency change (Hz/s)	3.4 Hz/s

Table 6.2 – Number of "insecure" and "secure" OPs in the TS of Terceira

Disturbance	Short-Circuit with Wind Power Loss		
Security Index	$B_1 = \Delta f_{min}$	$B_2 = \Delta f_{max}$	$B_3 = df/dt_{max}$
Nº of Insecure OPs	87	21	93
Nº of Secure OPs	703	769	697

The security problem evaluated for the power system of Crete is also described in Chapter 4. Just to remind, the security indices selected to define the dynamic security of this network are presented in Table 6.3. The obtained number of "insecure" and "secure" samples in the TS is presented in Table 6.4.

Table 6.3 – Security indices of the Crete case

Disturbance 1		
Machine Loss **:** outage of a major gas turbine		
Security Indices		**Security Boundary**
B_1	f_{min} : minimum value reached by the system frequency (Hz)	49Hz
B_2	df/dt_{max} : maximal rate reached by the frequency change (Hz/s)	0.4 Hz/s
Disturbance 2		
Short-Circuit **:** three-phase short-circuit at a critical bus near the wind parks		
Security Indices		**Security Boundary**
B_3	f_{min} : minimum value reached by the system frequency (Hz)	49Hz
B_4	df/dt_{max} : maximal rate reached by the frequency change (Hz/s)	0.4 Hz/s

Table 6.4 – Number of "insecure" and "secure" OPs in the TS of Crete

Disturbance	Machine Loss		Short-Circuit	
Security Index	$B_1 = f_{min}$	$B_2 = df/dt_{max}$	$B_3 = f_{min}$	$B_4 = df/dt_{max}$
Nº of Insecure OPs	20	32	375	354
Nº of Secure OPs	901	889	546	567

For the case of Terceira, the implemented HRT method was applied for the 3 security indices to obtain the corresponding AL structures. In order to evaluate the performance of this approach, researchers of NTUA provided a DT structure for the B_3 security index of the Terceira case.

For the case of Crete, by applying the implemented HRT and ANN methods, AL structures were obtained for the 4 considered security indices. Therefore, for this study case it was possible to compare accuracy performance between the implemented approach and the ANN approach. The provided ANNs were trained to provide security monitoring by emulating the expected values of f_{min} and df/dt_{max}. Therefore, they weren't trained for classification purposes. The two multi-layer ANNs were trained (one for each disturbance) using an adaptive back propagation algorithm [40]. For each ANN, the structure presented in Figure 6.1 was adopted, which includes:

- one input layer with 22 attributes as inputs;
- one hidden layer with 8 neurons;
- one output layer with the two security indices as outputs.

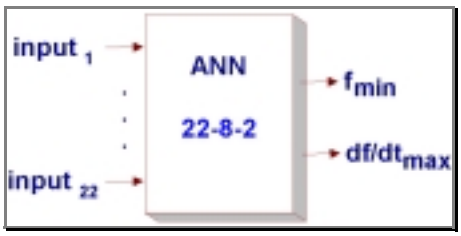


Figure 6.1 – Structure selected for the trained ANNs

The two provided ANNs have a response time in the order of some microseconds to predict, for one sample, both f_{min} and df/dt_{max} that succeed from a disturbance.

Also in order to compare the performance of HRT and ANN approaches, researchers of NTUA provided a DT structure for the B_3 security index of the Crete case.

6.2 Results for the Case of Terceira Island

6.2.1 B_1 of Terceira Case - Results Obtained with the HRT Method

In order to extract AL security structure for this case, the pruning algorithm described in Chapter 5 was applied to the B_1 security index of the Terceira LS. From this procedure a set $\{T\}$ of 1141 pruned trees was generated, where $T_1 \succ T_2 \succ \ldots \succ T_{1140} \succ root$, having T_1 (the most complex tree) 2329 nodes.

Regarding the use of the extracted RT and KRT structures to produce the emulation of the B_1 security index, after analyzing the obtained set of pruned trees $\{T\}$ one can derive the following main conclusions:

1. For each tree structure of $\{T\}$, the kernel regression tree (KRT) approach was able to provide security structures with smaller $RMSE^{TS}$ error than the regression tree (RT) approach (see Figure 6.2). Thus, among the RT and KRT approaches, in the sense of accuracy, the last one is the most suitable to produce on-line evaluation of the system security degree. In fact, as it can be seen in Figure 6.10, Figure 6.17, Figure 6.26, Figure 6.32, Figure 6.39 and Figure 6.49, this was observed for all the cases studied of the Terceira and Crete power systems.

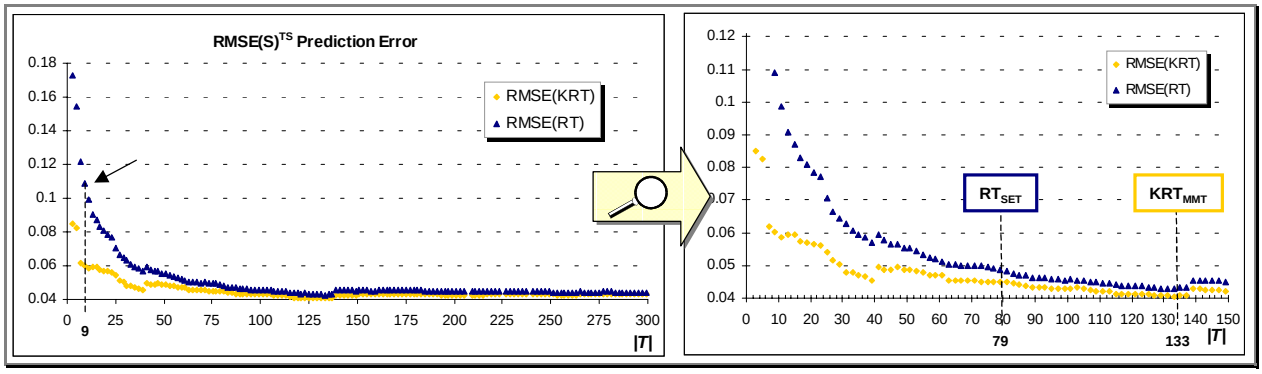


Figure 6.2 - Comparing $RMSE^{TS}$ error between the obtained {*KRT*} and {*RT*} ($B_{1,Terceira}$)

2. Among the extracted structures, the most suitable one to produce on-line evaluation of the system security degree is KRT_{MMT}, which has 133 nodes (see Figure 6.2). The results of the TS performance evaluation for this regression structure are presented in Figure 6.3.

   The parameter *prediction time* presented in the Table of Figure 6.3 is an estimation of the time that the structure takes to predict the security index value for one OP. It is measured as the mean value of the time that the structure takes to make prediction to the testing samples in a Pentium II Processor at 64 MB RAM. In the Graphic of Figure 6.3, each point represents one TS sample, where its vertical distance to the diagonal presents the predicting error of applying the KRT_{MMT} to that sample.

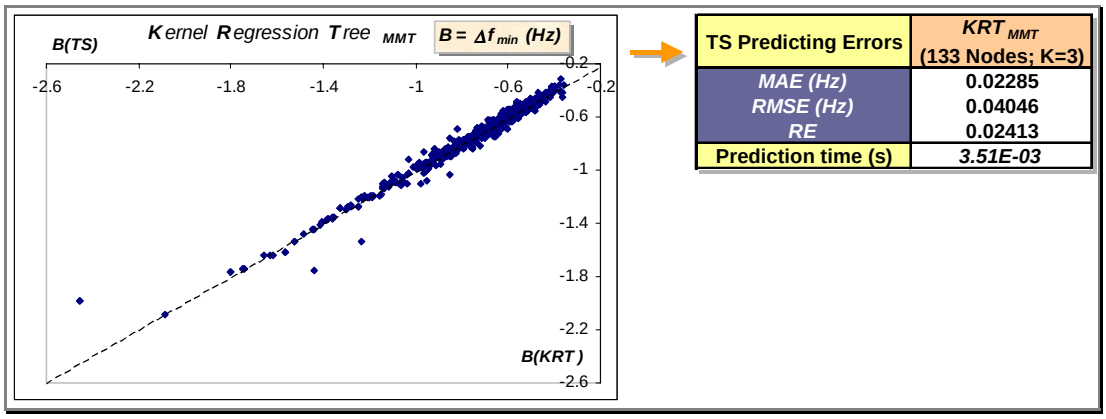


Figure 6.3 – TS performance evaluation results for the obtained KRT_{MMT} ($B_{1,Terceira}$)

3. In order to extract interpretable regression rules, the *1 SE rule* was applied to the set of RT structures $\{RT\}$. The selected RT, called in Figure 6.2 as RT_{SET}, has 79 nodes (40 leafs). Therefore, regarding the high number of nodes, RT_{SET} is too complex to be translated into comprehensible regression rules. The RT with 9 nodes (5 leafs) is considered much more suitable to extract simple regression rules. This structure is the one that verifies the *18 SE rule*. Its equivalent regression rules and TS regression errors are presented in Figure 6.4.

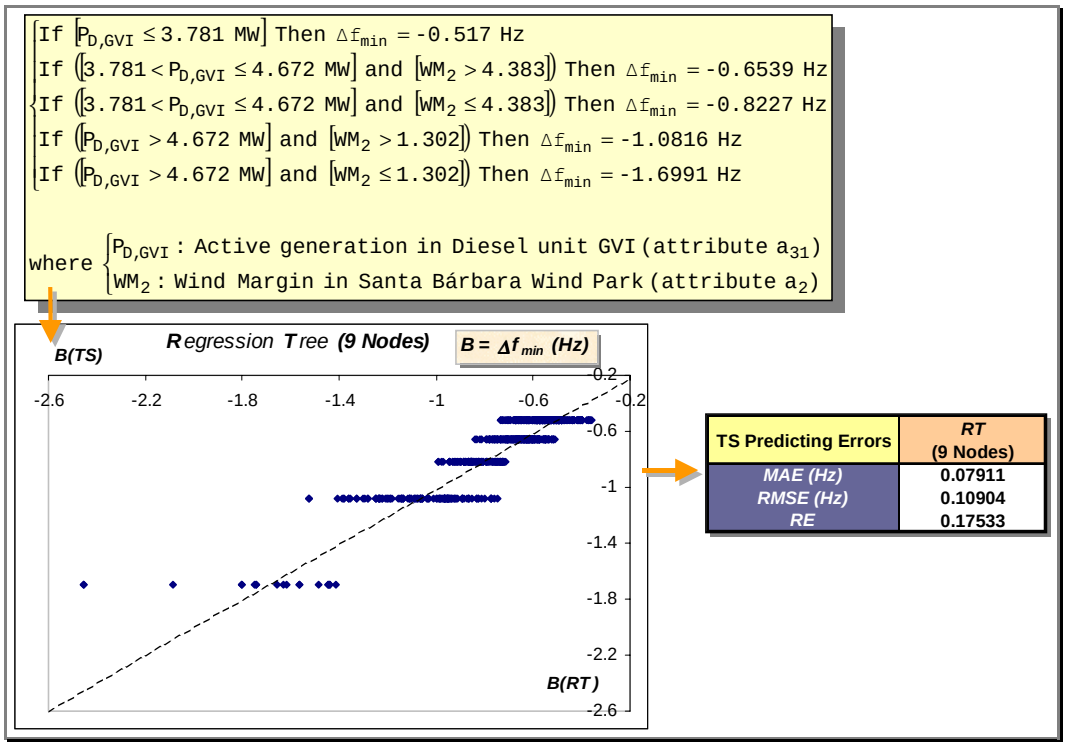


Figure 6.4 – Regression rules and TS regression errors for the obtained RT with 9 nodes ($B_{1,Terceira}$)

Regarding the use of the extracted RT and KRT structures to classify the B_1 security index as "secure/insecure", after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. As it can be seen in Figure 6.5, from the pruned tree structure with 97 nodes to the one with 133 nodes, the resulting KRT and RT structures achieve minimum *Global Classification Error*. For the KRT approach, the simpler the tree structure is the higher is the prediction time. Regarding this and also the values of the false and missed alarm presented in Figure 6.6, among {*KRT*}, the *KRT* with 133 nodes is considered a suitable structure to produce fast security classification. For the RT approach, the simpler the tree structure is the lower is the prediction time. Regarding this and also the values of the classification errors presented in Figure 6.7, among {*RT*}, the *RT* with 97 nodes is considered a suitable structure to produce fast security classification. Although these two structures have the same classification errors, the *RT* provides smaller predicting time. Therefore, among the extracted structures, the *RT* with 97 nodes is considered suitable to produce fast security classification. The results of the TS performance evaluation for this classification structure are presented in Figure 6.7.

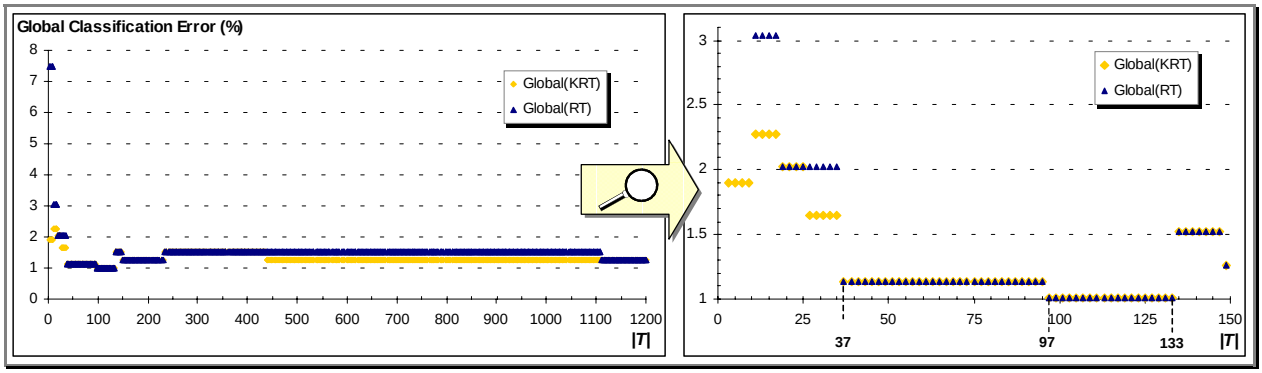


Figure 6.5 - Comparing *Global Classification Error* between the obtained {*KRT*} and {*RT*} ($B_{1,\text{Terceira}}$)

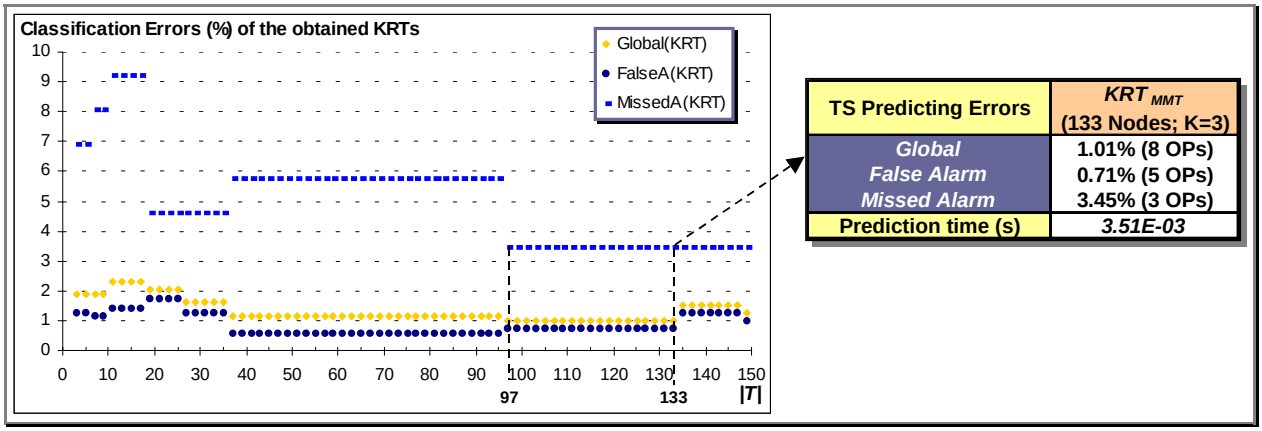


Figure 6.6 – TS classification errors for the obtained {*KRT*} ($B_{1,\text{Terceira}}$)

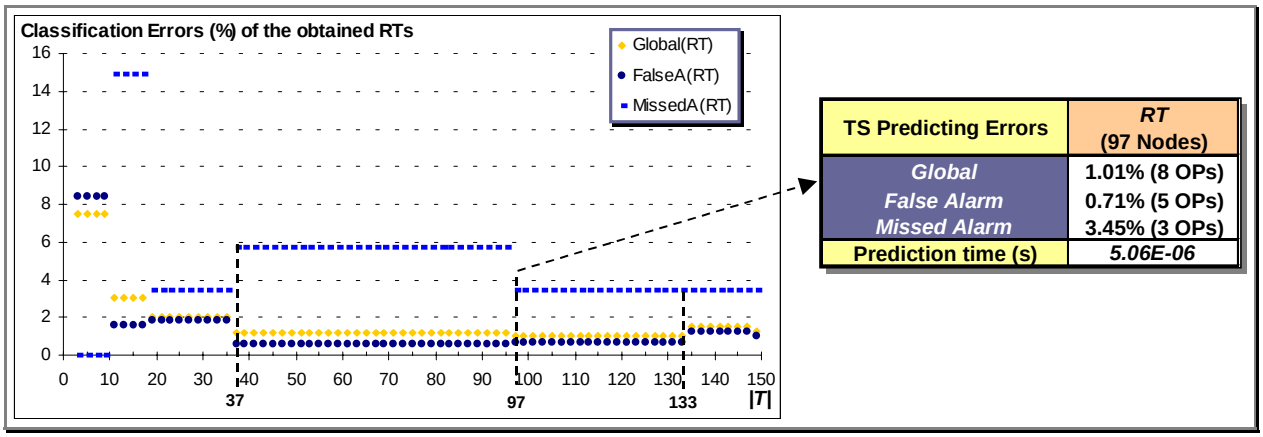


Figure 6.7 – TS classification errors for the obtained {*RT*} ($B_{1,Terceira}$)

2. As it can be also seen in Figure 6.7, among {*RT*}, the *RT* with 37 nodes achieves a good compromise between classification error and complexity. Moreover, its classification structure provides 3 *If rules*. Therefore, the *RT* with 37 nodes is considered suitable to extract classification rules. Its equivalent classification rules and TS classification errors are presented in Figure 6.8. Its tree structure is presented in Figure 6.9.

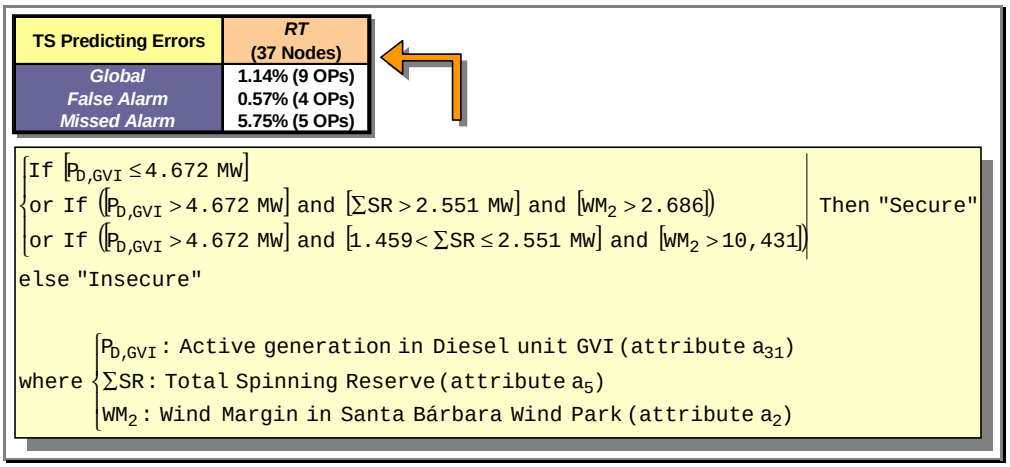


Figure 6.8 – Classification rules and TS errors for the obtained *RT* with 37 nodes ($B_{1,Terceira}$)

In Figure 6.9, nodes in the tree structure are of two types: non-terminal and terminal nodes (leafs). The root node (node number 1) includes information related to the total number of learning samples (*N*), the variance of the security index B_1 in the LS ($s^2(t)$) and the splitting test. Non-terminal nodes present the node number, containing also information related to the splitting test. The leaf nodes present information related to the node number, the number of learning samples stored there (*N*), and the mean (*Mean*) and variance ($s^2(t)$) of the security index B_1 in those samples.

6.2.1.1 *Interpretation of the Security Structures (B_1 of Terceira)*

As it can be seen in Figure 6.4 and Figure 6.8, the security rules extracted from the selected tree structures can be easily understood, discussed, and eventually adopted by the operators to define new operating strategies. Notice that the common practice of the engineers is to use, in an off-

line procedure, analytical tools of power system behavior simulation, together with their expertise, to run extensive scenarios. From those simulations, they extract the relevant security information in order to define operating guidelines and corrective measures. By using machine learning techniques, like the RT applied approach, much of the manual task to extract the operating guidelines can be performed automatically by a systematic methodology.

Regarding the extracted security rules and the considered disturbance, it seems obvious the influence of the total spinning reserve ($\sum SR$) and of the wind margin in Santa Bárbara wind park (WM_2). However, to understand the influence of the active generation in Diesel unit GVI ($P_{D,GVI}$) a closer examination is necessary to be performed. First of all, in the data set generation procedure, the Diesel unit GVI (one of the biggest machines of the Belo Jardim Diesel power station) was considered, by omission, as the slack machine. Moreover, the lowest production costs were assigned to the GVI and GVII Diesel units. As a result, these two machines were considered in service in all the DS operating scenarios. Furthermore, by taking a closer look to the generated data set, it was observed that when $P_{D,GVI} \leq 4.672\ MW$ then the total spinning reserve is within *3* and *6 MW*, being the wind power loss to compensate in the worst case of *1.8 MW* (i.e., the installed power in the disconnected wind park). Regarding these issues, it seems much more clear why $P_{D,GVI}$ has such an influence on the system security.

This example illustrates well how influent can the scheduling and dispatching strategies that are reflected in the data set be on the generalization capabilities of the extracted AL structure. In fact, as already highlighted in Section 3.9.2, if the considered operating strategies are not the ones actually performed on the network, then, in spite of having a good testing accuracy, there is no assurance that the structure will be accurate enough when making prediction to real life operating scenarios.

Summarizing:
According to the required function of the hybrid regression tree, among the generated set of pruned trees $\{T\}$, the structures presented in Table 6.5 were selected. In this table, the testing set performance evaluation results obtained for the security structures and the number of secure and insecure samples in the TS are also presented.

Table 6.5 – Selected hybrid regression trees ($B_{1,Terceira}$)

Short-Circuit + Wind Power Loss, Δf_{min}								
Function	**Selected structure**	**MAE error**	**RMSE error**	**RE error**	**Global error**	**False Alarm error**	**Missed Alarm error**	**Prediction time (s)**
on-line evaluation of security degree	KRT_{MMT} (133 nodes; K=3)	0.0228	0.0405	0.0241	-	-	-	*3.51E-03*
extract interpretable regression rules	*RT* with 9 nodes	0.0791	0.1090	0.1753	-	-	-	-
fast security classification	*RT* with 97 nodes	-	-	-	1.01% (8 OPs)	0.71% (5 OPs)	3.45% (3 OPs)	*5.06E-06*
extract interpretable classification rules	*RT* with 37 nodes	-	-	-	1.14% (9 OPs)	0.57% (4 OPs)	5.75% (5 OPs)	-
Nº of Insecure OPs in the TS		87						
Nº of Secure OPs in the TS		703						

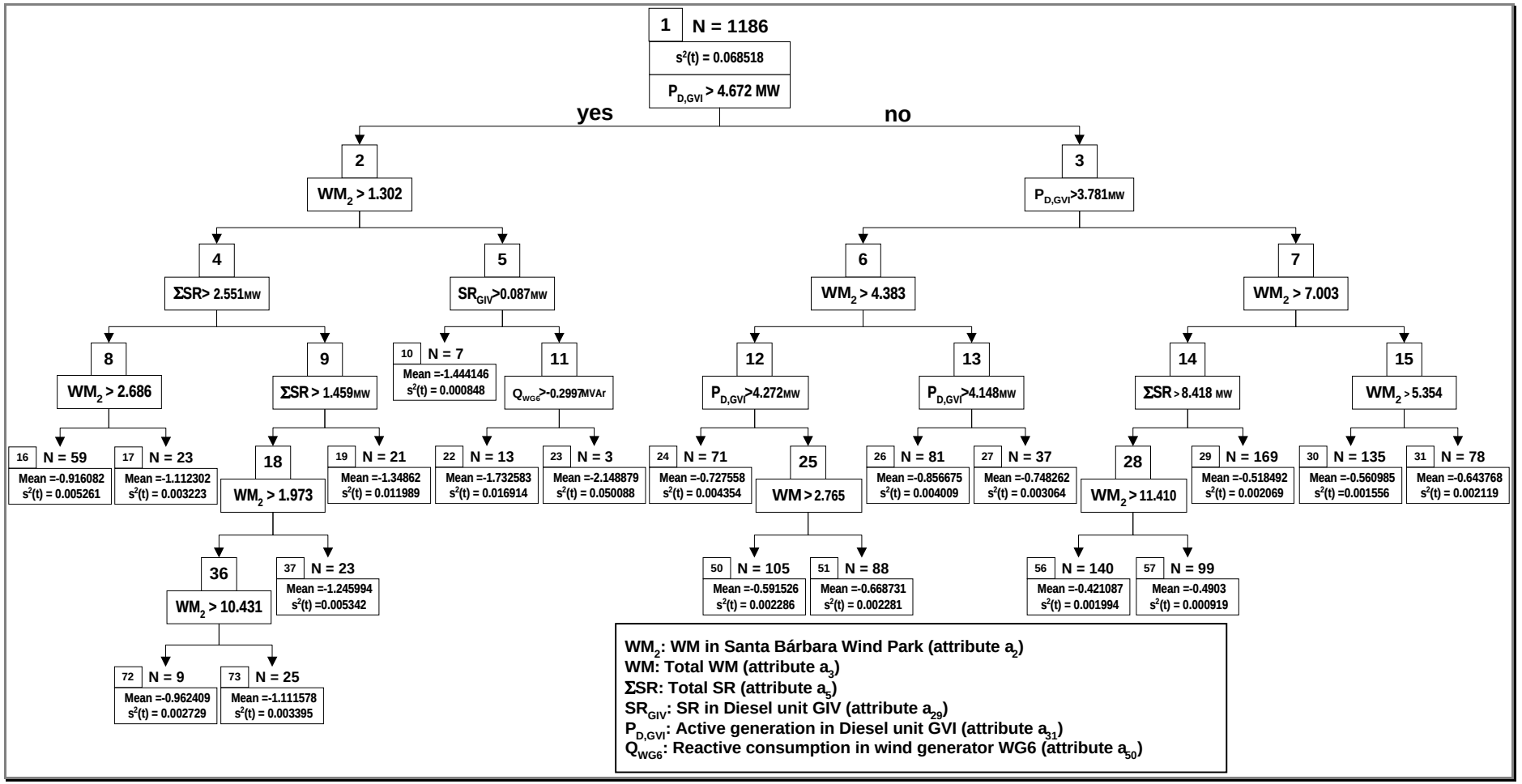


Figure 6.9 – Tree structure for the obtained *RT* with 37 nodes *($B_{1,Terceira}$)*

6.2.2 B_2 of Terceira Case - Results Obtained with the HRT Method

In this case, by applying the pruning algorithm described in Chapter 5, a set $\{T\}$ of 1130 pruned trees was generated, where $T_1 \succ T_2 \succ \ldots \succ T_{1129} \succ root$, having T_1 2329 nodes.

Regarding the use of the extracted RT and KRT structures to produce emulation of the B_2 security index, after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. For this case, KRT_{MMT} – the most suitable structure to produce on-line evaluation of security degree – has 203 nodes (see Figure 6.10). The results of the TS performance evaluation for this regression structure are presented in Figure 6.11.

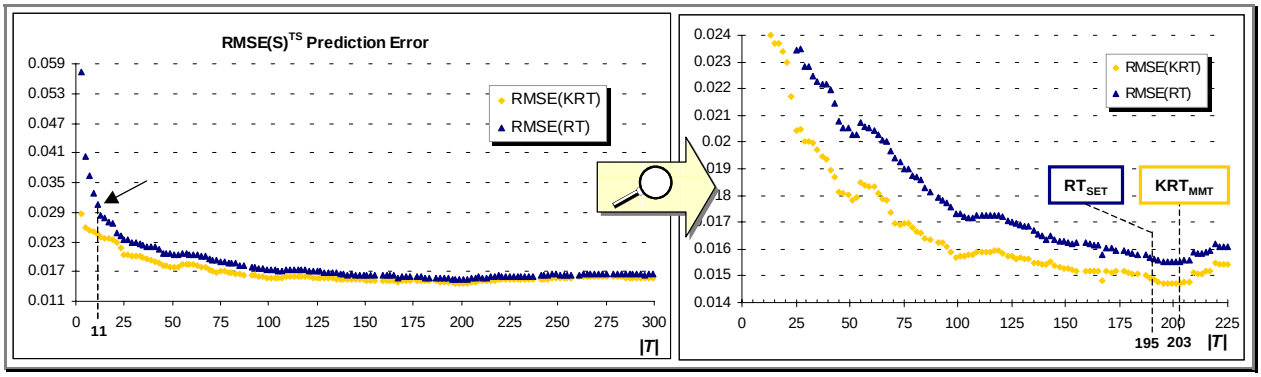


Figure 6.10 - Comparing $RMSE^{TS}$ error between the obtained {*KRT*} and {*RT*} ($B_{2,Terceira}$)

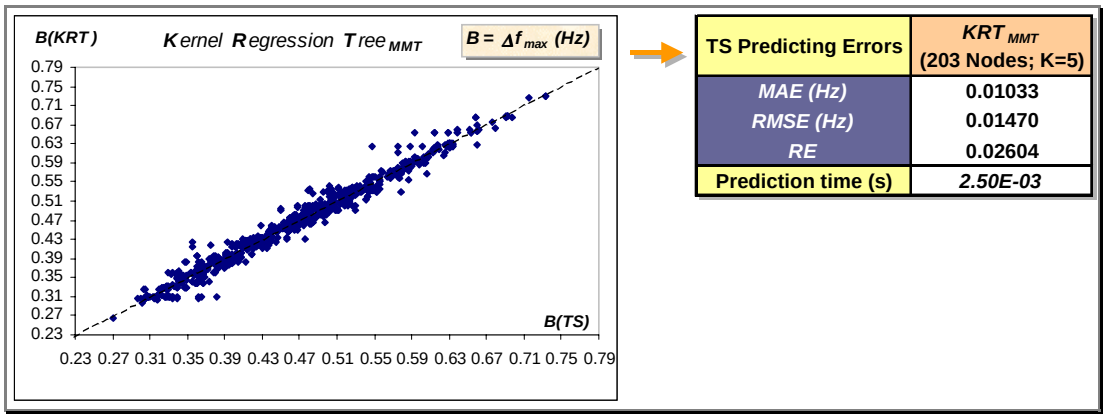


Figure 6.11 – TS performance evaluation results for the obtained KRT_{MMT} ($B_{2,Terceira}$)

2. As it can be seen in Figure 6.10, for this example RT_{SET} has 195 nodes (98 leafs), being thus too complex to be translated into comprehensible regression rules. The *RT* with 11 nodes (6 leafs) is considered much more suitable to extract simple regression rules. This structure is the one that verifies the *35 SE rule*. Its equivalent regression rules and TS regression errors are presented in Figure 6.12. Its tree structure is presented in Figure 6.16.

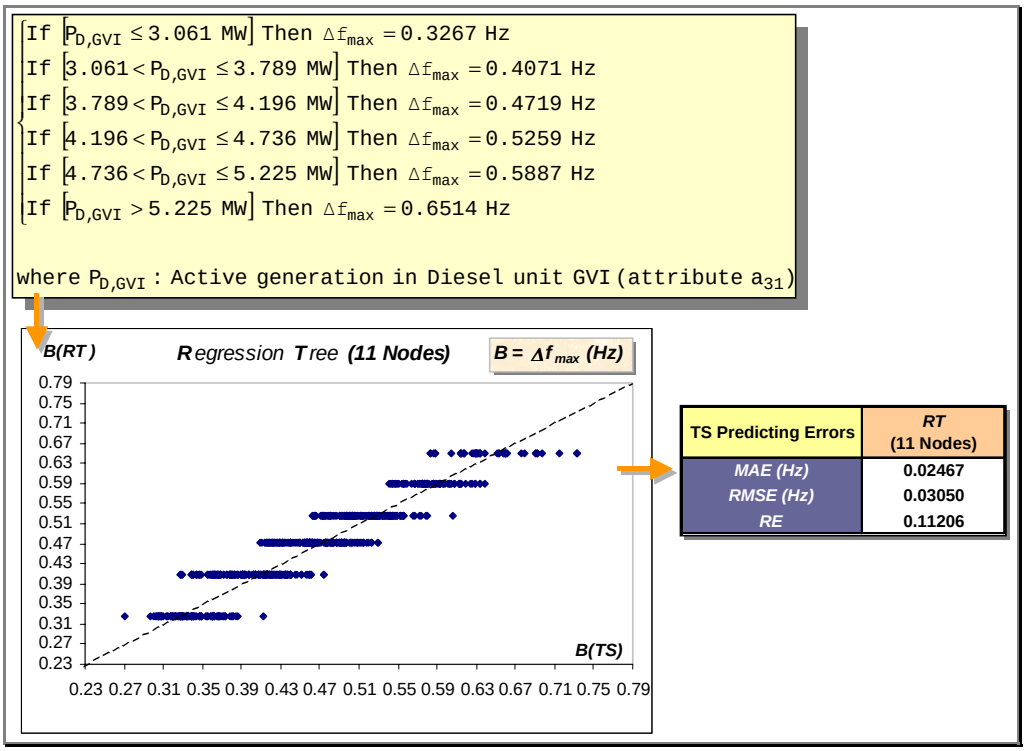


Figure 6.12 – Regression rules and TS regression errors for the obtained *RT* with 11 nodes ($B_{2,Terceira}$)

Regarding the use of the extracted RT and KRT structures to classify the B_2 security index as "secure/insecure", after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. As it can be seen in Figure 6.13, from the pruned tree structure with 47 nodes to the one with 53 nodes, the resulting KRT and RT structures achieve minimum *Global Classification Error*. Regarding this and the TS performance evaluation presented in Figure 6.14 and Figure 6.15, among the extracted structures, the *RT* with 47 nodes is considered suitable to produce fast security classification. The results of the TS performance evaluation for this classification structure are presented in Figure 6.15.

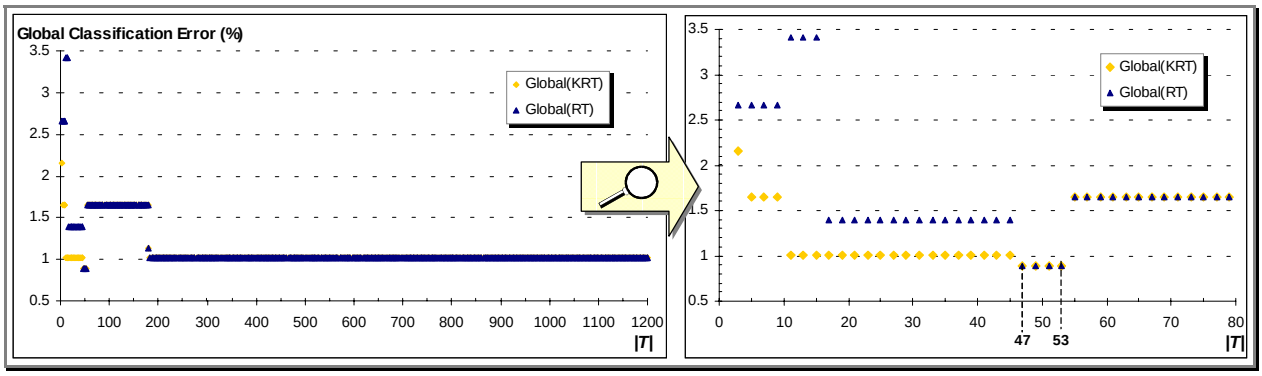


Figure 6.13 - Comparing *Global Classification Error* between the obtained {*KRT*} and {*RT*} ($B_{2,Terceira}$)

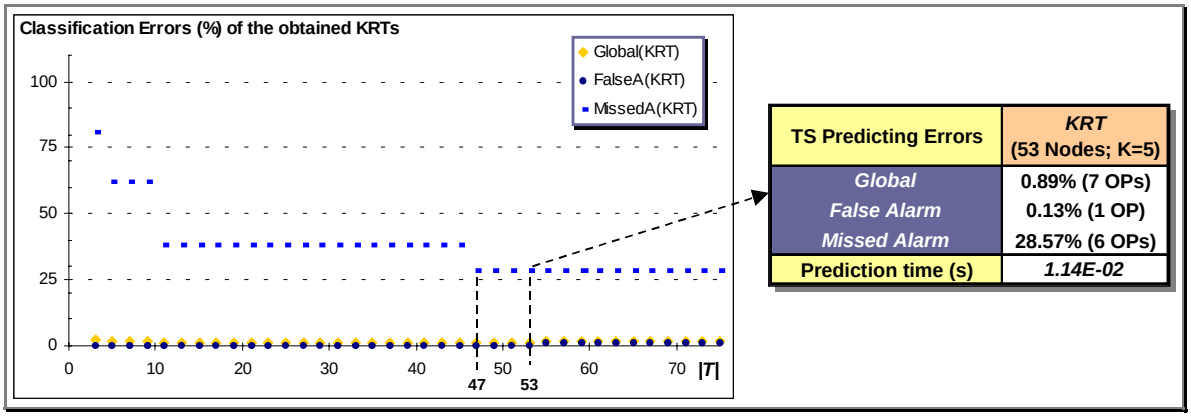


Figure 6.14 – TS classification errors for the obtained {*KRT*} ($B_{2,Terceira}$)

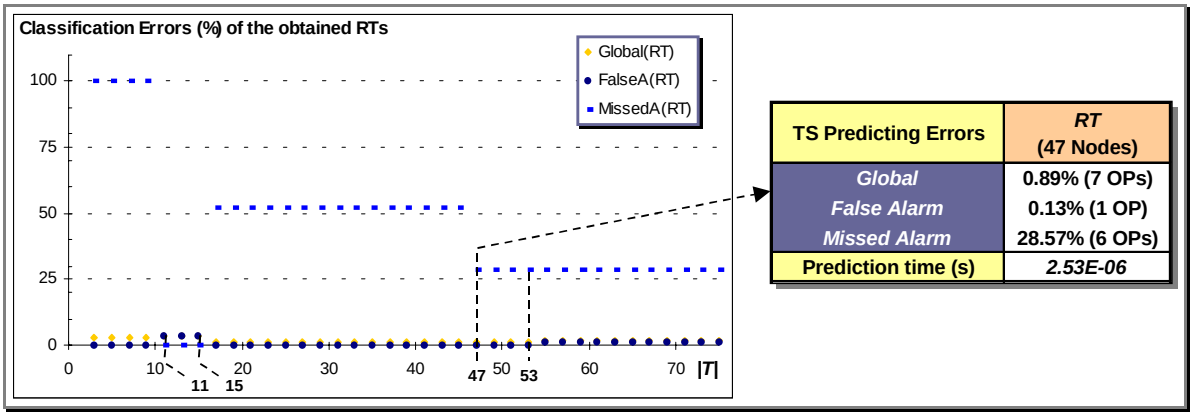


Figure 6.15 – TS classification errors for the obtained {*RT*} ($B_{2,Terceira}$)

2. As it can be also seen in Figure 6.15, among {*RT*}, the *RT* with 11 nodes achieves a good compromise between classification error and complexity. Moreover, its classification structure provides 1 *If rule*. Therefore, the *RT* with 11 nodes is considered suitable to extract classification rules. Its tree structure, extracted classification rules, and TS classification errors are presented in Figure 6.16.

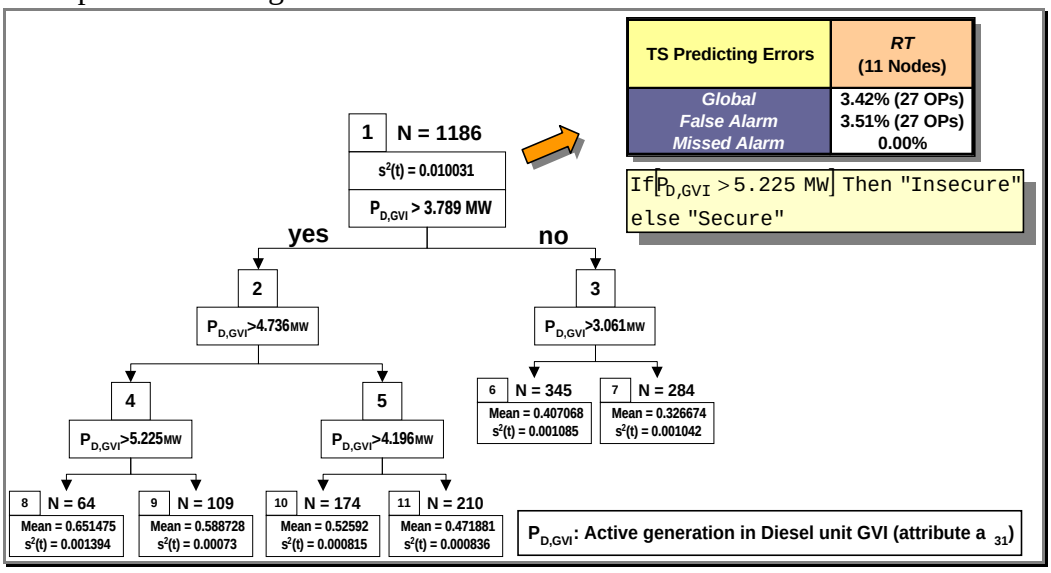


Figure 6.16 – Tree structure, classification rules and TS classification errors for the obtained *RT* with 11 nodes ($B_{2,Terceira}$)

Summarizing:

According to the required function of the hybrid regression tree, among the generated set of pruned trees $\{T\}$, the structures presented in Table 6.6 were selected.

Table 6.6 – Selected hybrid regression trees ($B_{2,Terceira}$)

Short-Circuit + Wind Power Loss, Δf_{max}								
Function	**Selected structure**	**MAE error**	**RMSE error**	**RE error**	**Global error**	**False Alarm error**	**Missed Alarm error**	**Prediction time (s)**
on-line evaluation of security degree	*KRT* $_{MMT}$ (203 nodes; K=5)	0.0103	0.0147	0.0260	-	-	-	*2.50E-03*
extract interpretable regression rules	*RT* with 11 nodes	0.0247	0.0305	0.1121	-	-	-	-
fast security classification	*RT* with 47 nodes	-	-	-	0.89% (7 OPs)	0.13% (1 OP)	28.57% (6 OPs)	*2.53E-06*
extract interpretable classification rules	*RT* with 11 nodes	-	-	-	3.42% (27 OPs)	3.51% (27 OPs)	0.00%	-
Nº of Insecure OPs in the TS		21						
Nº of Secure OPs in the TS		769						

6.2.3 B_3 of Terceira Case

6.2.3.1 Results Obtained with the HRT Method (B_3 of Terceira)

In this case, by applying the pruning algorithm described in Chapter 5, a set $\{T\}$ of 1132 pruned trees was generated, where $T_1 \succ T_2 \succ \ldots \succ T_{1131} \succ root$, having T_1 2327 nodes.

Regarding the use of the extracted RT and KRT structures to produce emulation of the B_3 security index, after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. For this case, KRT_{MMT} – the most suitable structure to produce on-line evaluation of security degree – has 751 nodes (see Figure 6.17). The results of the TS performance evaluation for this regression structure are presented in Figure 6.18.

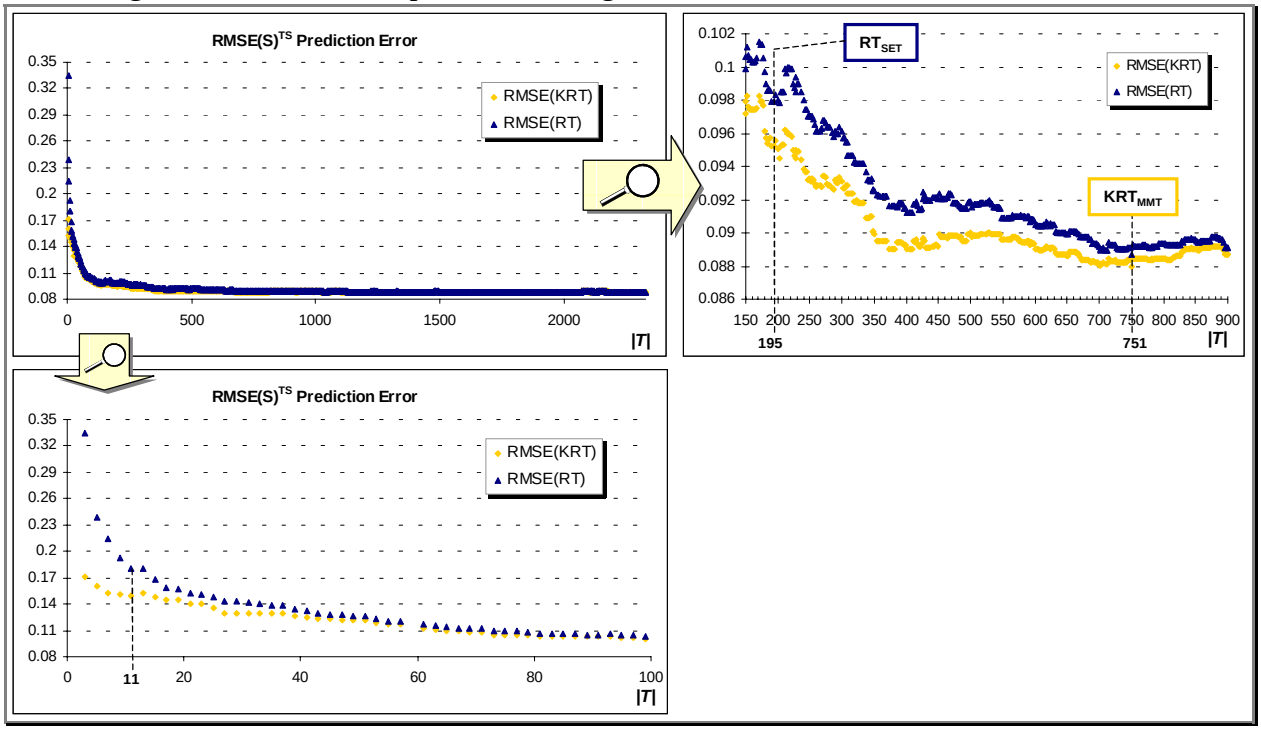


Figure 6.17 - Comparing $RMSE^{TS}$ error between the obtained {*KRT*} and {*RT*} ($B_{3,Terceira}$)

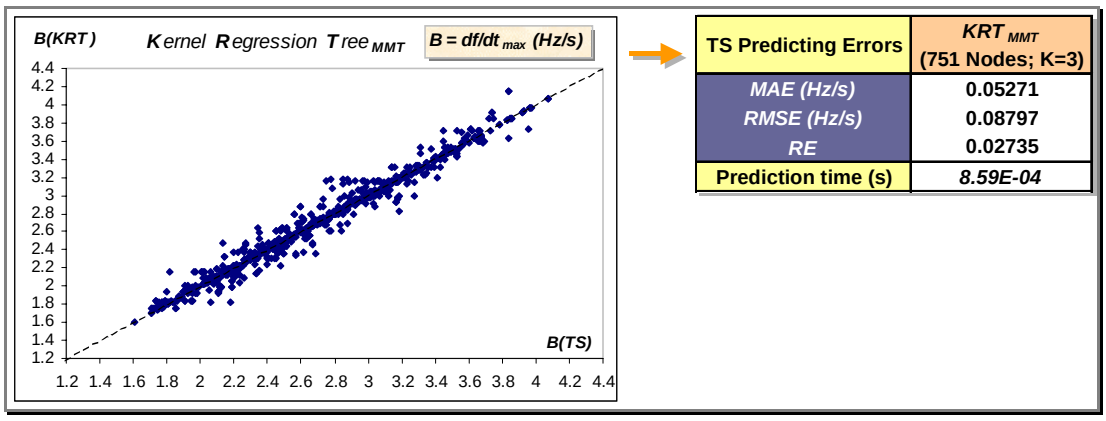


Figure 6.18 – TS performance evaluation results for the obtained KRT_{MMT} ($B_{3,Terceira}$)

2. As it can be seen in Figure 6.17, for this example RT_{SET} has 195 nodes (98 leafs), being thus too complex to be translated into comprehensible regression rules. The *RT* with 11 nodes (6 leafs) is considered much more suitable to extract simple regression rules. This structure is the one that verifies the *42 SE rule*. Its equivalent regression rules and TS regression errors are presented in Figure 6.19.

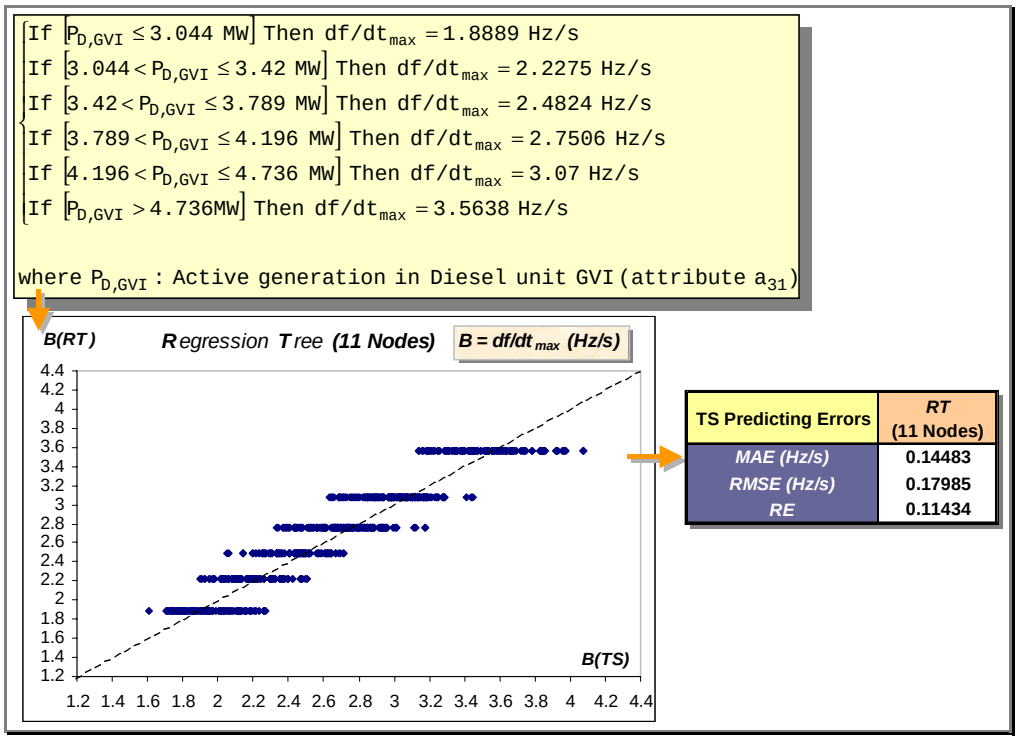


Figure 6.19 – Regression rules and TS regression errors for the obtained *RT* with 11 nodes ($B_{3,Terceira}$)

Regarding the use of the extracted RT and KRT structures to classify the B_3 security index as "secure/insecure", after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. As it can be seen in Figure 6.20, from the pruned tree structure with 177 nodes to the one with 181 nodes, the *Global Classification Error* of the KRT and RT structures achieve minimum values. Regarding this and also the false and missed alarms presented in Figure 6.21, among {*KRT*}, the *KRT* with 181 nodes is considered a suitable structure to produce fast security classification. Regarding the classification errors presented in Figure 6.22, among {*RT*}, the *RT* with 177 nodes is considered a suitable structure to produce fast security classification. As we can see by comparing Figure 6.21 with Figure 6.22, the *KRT* structure with 181 nodes provides smaller *Global* and *Missed Alarm Errors* than the *RT* with 177 nodes. Thus, among the extracted structures, the *KRT* with 181 nodes is considered suitable to produce fast security classification. The results of the TS performance evaluation for this classification structure are presented in Figure 6.21.

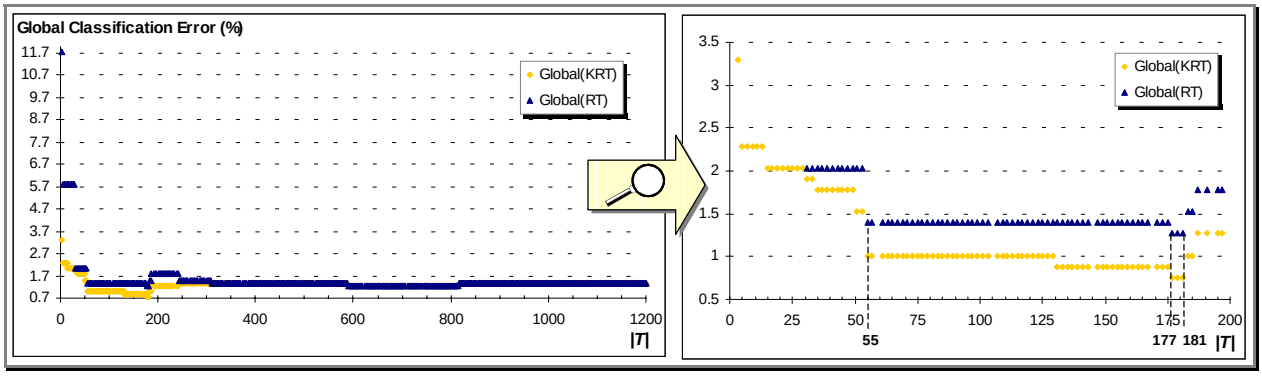
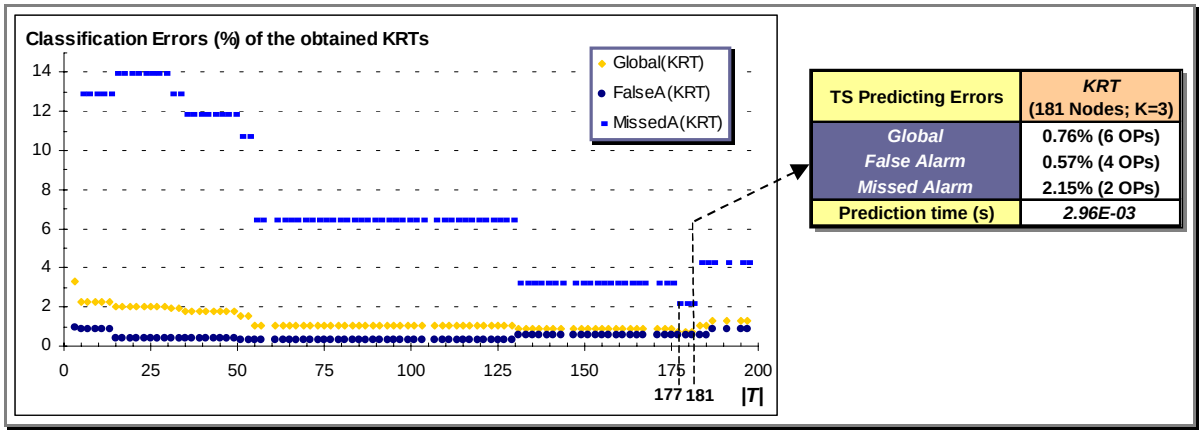


Figure 6.20 – Comparing *Global Classification Error* between the obtained {*KRT*} and {*RT*} ($B_{3,Terceira}$)



TS Predicting Errors	*KRT* (181 Nodes; K=3)
Global	0.76% (6 OPs)
False Alarm	0.57% (4 OPs)
Missed Alarm	2.15% (2 OPs)
Prediction time (s)	*2.96E-03*

Figure 6.21 – Classification errors for the obtained {*KRT*} ($B_{3,Terceira}$)

TS Predicting Errors	*RT* (55 Nodes)
Global	1.39% (11 OPs)
False Alarm	0.287% (2 OPs)
Missed Alarm	9.68% (9 OPs)

TS Predicting Errors	*RT* (177 to 181 Nodes)
Global	1.27% (10 OPs)
False Alarm	0.29% (2 OPs)
Missed Alarm	8.6% (8 OPs)

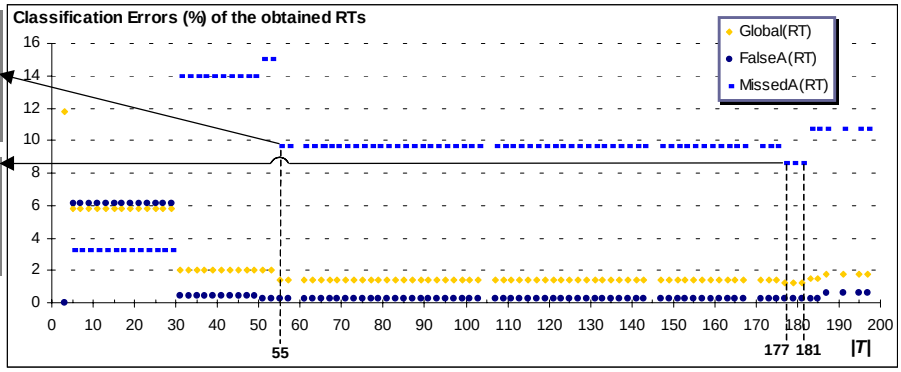


Figure 6.22 – Classification errors for the obtained {*RT*} ($B_{3,Terceira}$)

2. As it can be also seen in Figure 6.22, among {*RT*}, the *RT* with 55 nodes achieves a good compromise between classification error and complexity. Moreover, its classification structure provides 3 *If rules*. Therefore, the *RT* with 55 nodes is considered suitable to extract classification rules. Its equivalent classification rules are presented in Figure 6.23, and its TS classification errors are presented in Figure 6.22.

$$\begin{cases} \text{If} \left[P_{D,GVI} > 5.225 \text{ MW} \right] \\ \text{or If} \left(\left[5.076 < P_{D,GVI} \leq 5.225 \text{ MW} \right] \text{and} \left[SR_{GIV} > 0.218 \text{MW} \right] \right) \\ \text{or If} \left(\left[4.736 < P_{D,GVI} \leq 5.225 \text{ MW} \right] \text{and} \left[SR_{GIV} \leq 0.218 \text{MW} \right] \text{and} \left[\sum \text{active losses} > 395 \text{kW} \right] \right) \end{cases} \text{Then "Insecure"}$$

$$\text{else "Secure"}$$

$$\text{where} \begin{cases} \sum \text{active losses: Total active losses (attribute } a_{13}) \\ SR_{GIV} : \text{Spinning Reserve in Diesel unit GIV (attribute } a_{29}) \\ P_{D,GVI} : \text{Active generation in Diesel unit GVI (attribute } a_{31}) \end{cases}$$

Figure 6.23 – Classification rules extracted by the obtained *RT* with 55 nodes *($B_{3,Terceira}$)*

Summarizing:

According to the required function of the hybrid regression tree, among the generated set of pruned trees $\{T\}$, the structures presented in Table 6.7 were selected.

Table 6.7 – Selected hybrid regression trees *($B_{3,Terceira}$)*

Short-Circuit + Wind Power Loss, df/dt_{max}								
Function	**Selected structure**	**MAE error**	**RMSE error**	**RE error**	**Global error**	**False Alarm error**	**Missed Alarm error**	**Prediction time (s)**
on-line evaluation of security degree	***KRT*** $_{MMT}$ (751 nodes; K=3)	0.0527	0.0880	0.0274	-	-	-	*8.59E-04*
extract interpretable regression rules	***RT*** with 11 nodes	0.1448	0.1799	0.1143	-	-	-	-
fast security classification	***KRT*** with 181 nodes and K=3	-	-	-	0.76% (6 OPs)	0.57% (4 OPs)	2.15% (2 OPs)	*2.96E-03*
extract interpretable classification rules	***RT*** with 55 nodes	-	-	-	1.39% (11 OPs)	0.287% (2 OPs)	9.68% (9 OPs)	-
Nº of Insecure OPs in the TS		93						
Nº of Secure OPs in the TS		697						

6.2.3.2 Results Obtained with the Provided DT (B_3 of Terceira)

For comparative purposes, the tree structure and TS classification errors obtained for the DT provided by the NTUA researchers are presented in Figure 6.24.

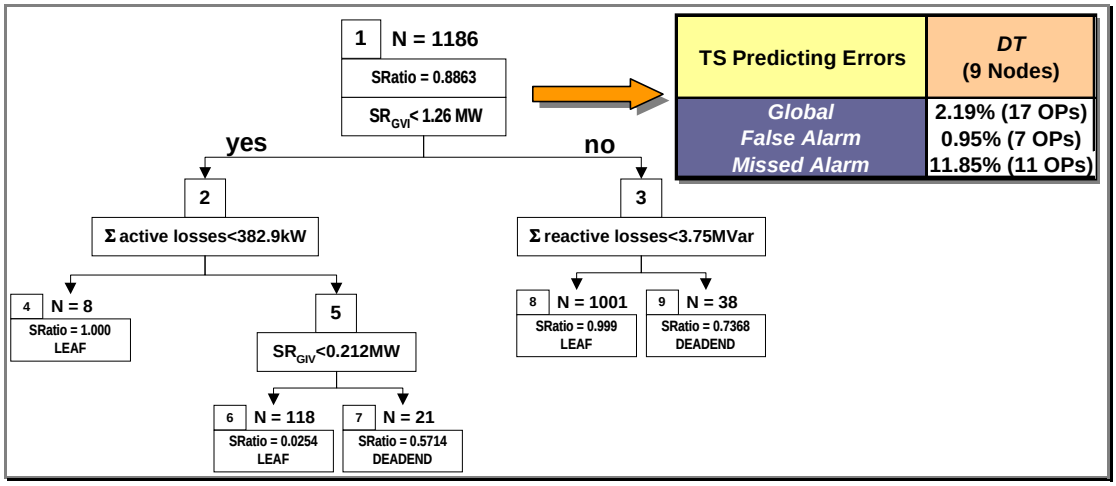


Figure 6.24 – Tree structure and TS classification errors for the *DT* provided by NTUA *($B_{3,Terceira}$)*

For the DT described in Figure 6.24, the nodes in the tree structure are of two types: non-terminal (including the root node) and terminal nodes (leafs or deadend). The contents of the root node are the following: the node number (1), the number of learning samples belonging to the node (*N*), the safety ratio (*SRatio = ratio of the number of LS "secure" samples over N*) and the splitting test. Non-terminal nodes present the node number and the splitting test. Terminal nodes present the node number, the number of learning samples belonging to the node (*N*), the safety ratio (*SRatio*), and the type of the node (*LEAF or DEADEND*). A terminal node becomes a *LEAF* if its entropy is lower than a pre-defined minimum value. Otherwise, a test T is applied to further split the node. If the node cannot be further splitted in a statistically significant way, then it becomes a *DEADEND*. Terminal nodes with a safety ratio larger than 0,5 correspond to "secure" nodes.

From the provided DT, the following classification rules can be extracted:

$$\text{If}\left(\left[SR_{GVI} < 1.26\text{MW}\right] \text{and} \left[SR_{GIV} < 0.212\text{MW}\right] \text{and} \left[\sum \text{active losses} \geq 382.9\text{kW}\right]\right) \text{Then "Insecure" else "Secure"}$$

$$\text{where} \begin{cases} \sum \text{active losses}: \text{Total active losses (attribute } a_{13}) \\ SR_{GIV}: \text{Spinning Reserve in Diesel unit GIV (attribute } a_{29}) \\ SR_{GVI}: \text{Spinning Reserve in Diesel unit GVI (attribute } a_{33}) \end{cases}$$

Figure 6.25 – Classification rules extracted by the *DT* provided by NTUA *($B_{3,Terceira}$)*

6.3 Results for the Case of Crete Island

6.3.1 B_1 of Crete Case

6.3.1.1 Results Obtained with the HRT Method (B_1 of Crete)

In this case, by applying the pruning algorithm described in Chapter 5, a set $\{T\}$ of 935 pruned trees was generated, where $T_1 \succ T_2 \succ \ldots \succ T_{934} \succ root$, having T_1 3085 nodes.

Regarding the use of the extracted RT and KRT structures to produce emulation of the B_1 security index, after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. For this case, KRT_{MMT} – the most suitable structure to produce on-line evaluation of security degree – has 39 nodes (see Figure 6.26). The results of the TS performance evaluation for this regression structure are presented in Figure 6.27.

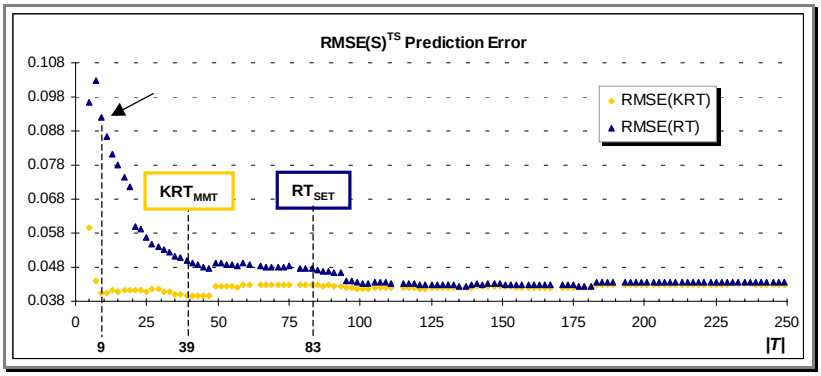
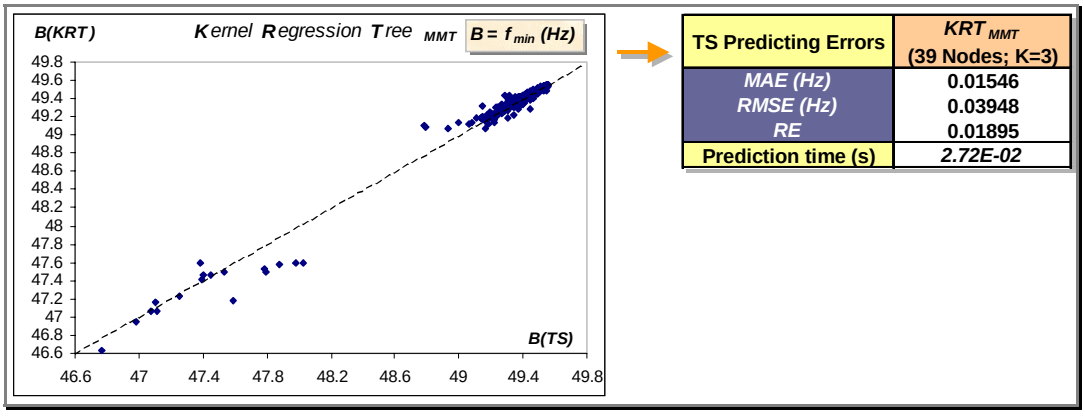


Figure 6.26 - Comparing $RMSE^{TS}$ error between the obtained $\{KRT\}$ and $\{RT\}$ $(B_{1,Crete})$



TS Predicting Errors	KRT_{MMT} (39 Nodes; K=3)
MAE (Hz)	0.01546
RMSE (Hz)	0.03948
RE	0.01895
Prediction time (s)	2.72E-02

Figure 6.27 – TS performance evaluation results for the obtained KRT_{MMT} $(B_{1,Crete})$

2. As it can be also seen in Figure 6.26, for this example RT_{SET} has 83 nodes (42 leafs), being thus too complex to be translated into comprehensible regression rules. The *RT* with 9 nodes (5 leafs) is considered much more suitable to extract simple regression rules. This structure is the one that verifies the *15 SE rule*. Its equivalent regression rules and TS regression errors are presented in Figure 6.28.

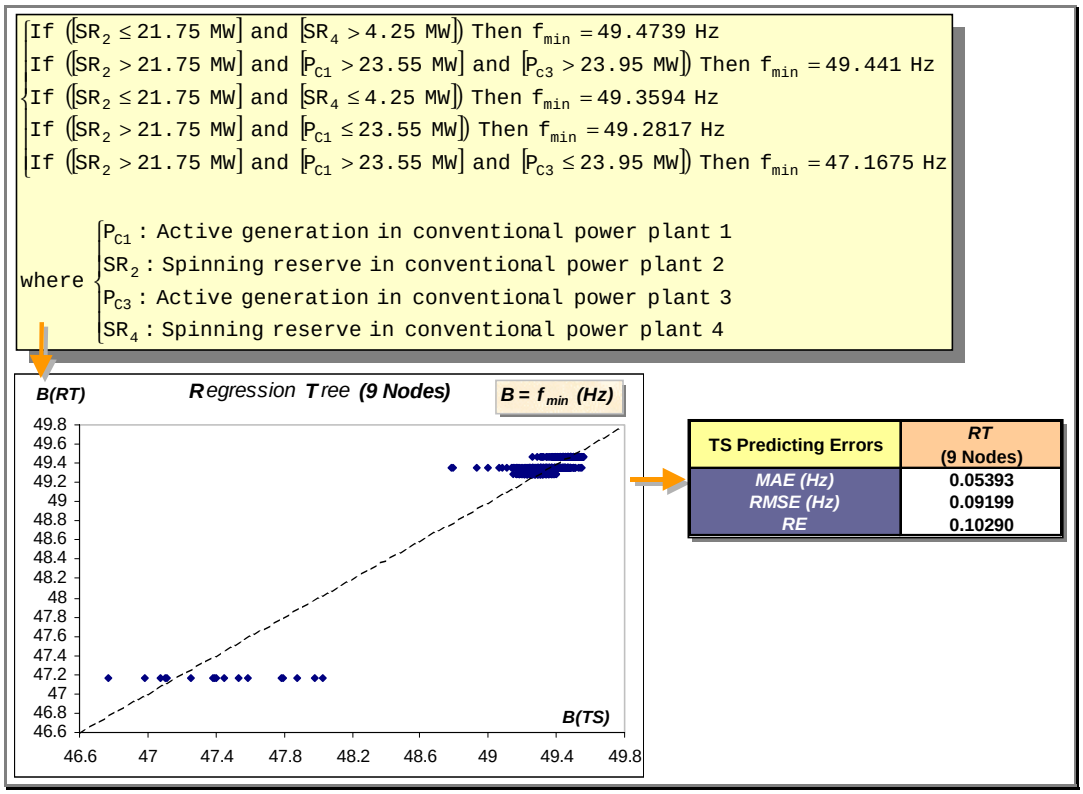


Figure 6.28 – Regression rules and TS regression errors for the obtained *RT* with 9 nodes ($B_{1,Crete}$)

Regarding the use of the extracted RT and KRT structures to classify the B_1 security index as "secure/insecure", after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. As it can be seen in Figure 6.29, all the set of KRT and RT structures achieve the same classification errors. Therefore, among the extracted structures, the simplest RT (which has 5 nodes) is the most suitable structure to produce fast security classification and to be translated into interpretable classification rules. Its tree structure, equivalent classification rules and TS performance evaluation results are presented in Figure 6.30.

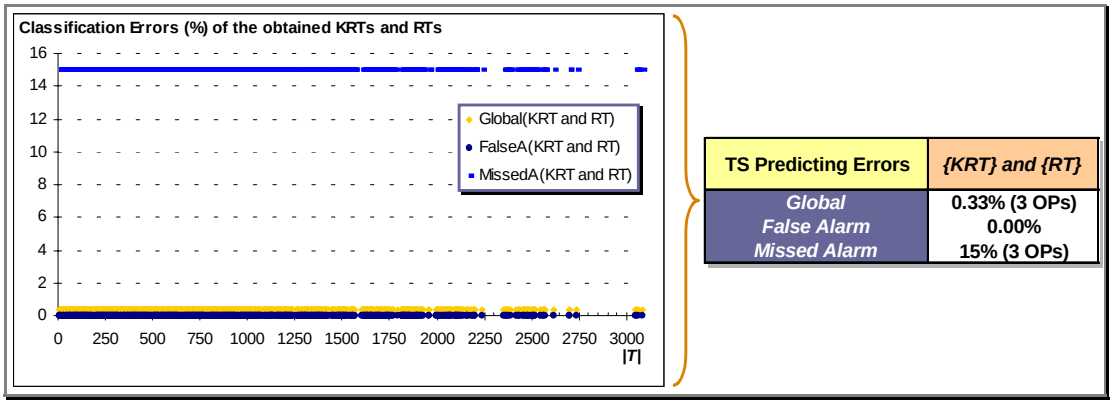


Figure 6.29 – TS classification errors for the obtained {*KRT*} and {*RT*} ($B_{1,Crete}$)

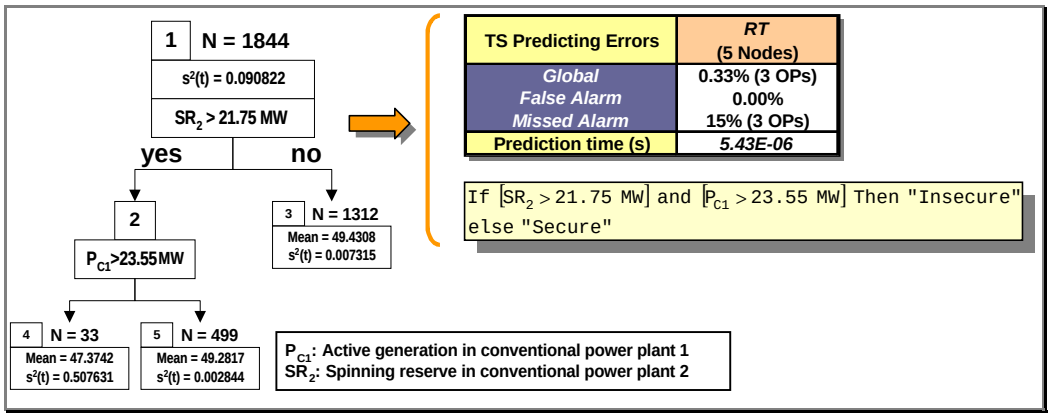


Figure 6.30 – Tree structure, classification rules and performance evaluation results for the *RT* with 5 nodes *($B_{1,Crete}$)*

Summarizing:

According to the required function of the hybrid regression tree, among the generated set of pruned trees $\{T\}$, the structures presented in Table 6.8 were selected.

Table 6.8 – Selected hybrid regression trees *($B_{1,Crete}$)*

Machine Loss, f_{min}								
Function	**Selected structure**	**MAE error**	**RMSE error**	**RE error**	**Global error**	**False Alarm error**	**Missed Alarm error**	**Prediction time (s)**
on-line evaluation of security degree	*KRT* $_{MMT}$ (39 nodes; K=3)	0.0155	0.0395	0.0190	-	-	-	*2.72E-02*
extract interpretable regression rules	*RT* with 9 nodes	0.0539	0.0920	0.1029	-	-	-	-
fast security classification and extract interpretable classification rules	*RT* with 5 nodes	-	-	-	0.33% (3 OPs)	0.00%	15% (3 OPs)	*5.43E-06*
Nº of Insecure OPs in the TS		20						
Nº of Secure OPs in the TS		901						

6.3.1.2 Results Obtained with the ANN Method (B_1 of Crete)

In Figure 6.31, the TS regression and classification errors obtained for the trained ANN are presented.

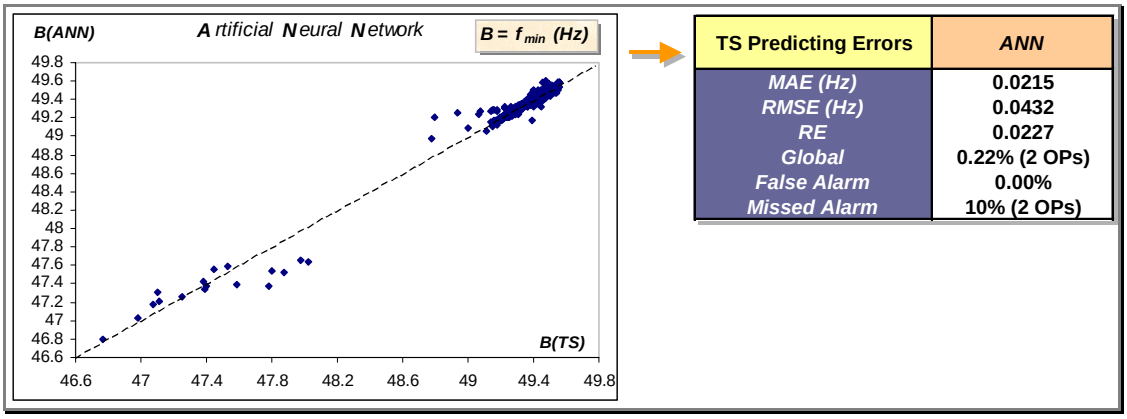


Figure 6.31 – TS errors for the trained *ANN ($B_{1,Crete}$)*

6.3.2 B_2 of Crete Case

6.3.2.1 Results Obtained with the HRT Method (B_2 of Crete)

In this case, by applying the pruning algorithm described in Chapter 5, a sequence $\{T\}$ of 1122 pruned trees was generated, where $T_1 \succ T_2 \succ \ldots \succ T_{1121} \succ root$, having T_1 3403 nodes.

Regarding the use of the extracted RT and KRT structures to produce emulation of the B_2 security index, after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. For this case, KRT_{MMT} – the most suitable structure to produce on-line evaluation of security degree – has 25 nodes (see Figure 6.32). The results of the TS performance evaluation for this structure are presented in Figure 6.33.

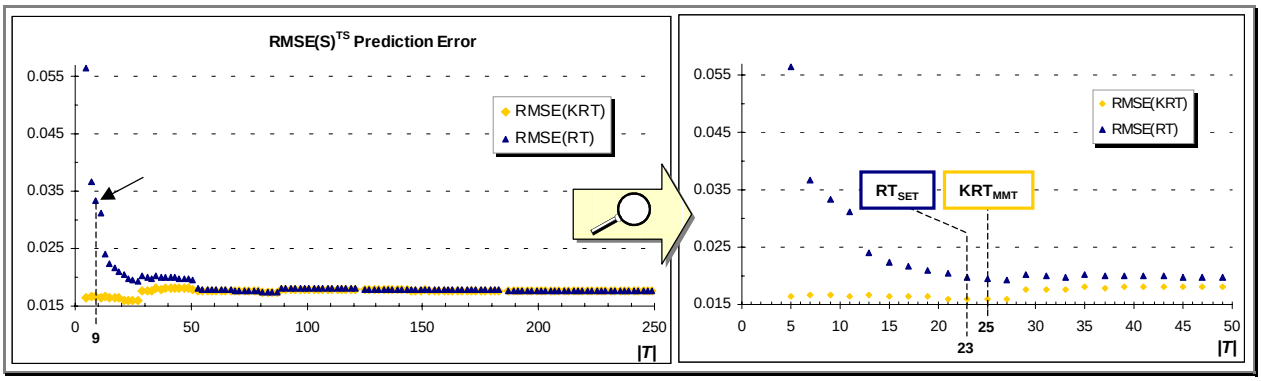


Figure 6.32 - Comparing $RMSE^{TS}$ error between the obtained {*KRT*} and {*RT*} ($B_{2,Crete}$)

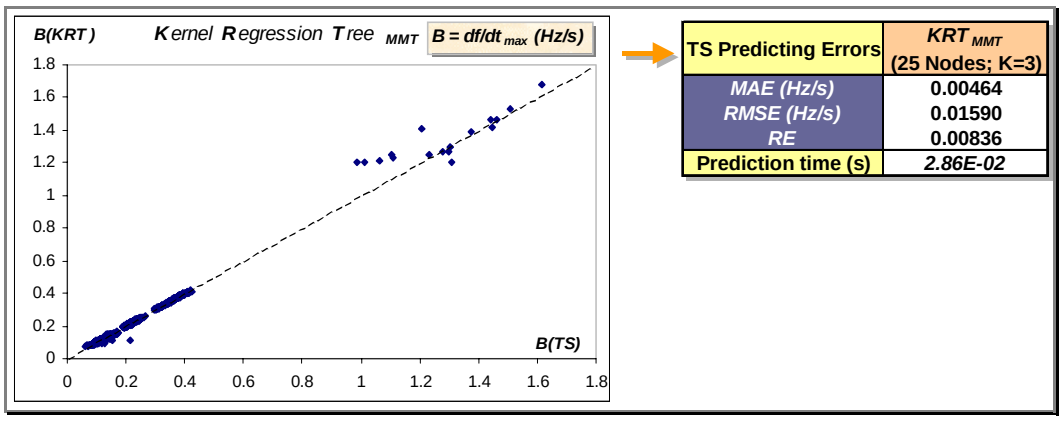


Figure 6.33 – TS performance evaluation results for the obtained KRT_{MMT} ($B_{2,Crete}$)

2. As it can be also seen in Figure 6.32, for this example RT_{SET} has 23 nodes (12 leafs), being thus too complex to be translated into comprehensible regression rules. The *RT* with 9 nodes (5 leafs) is considered much more suitable to extract simple regression rules. This structure is the one that verifies the *8 SE rule*. Its equivalent regression rules and TS regression errors are presented in Figure 6.34.

Regarding the use of the extracted RT and KRT structures to classify the B_2 security index as "secure/insecure", after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. As it can be seen in Figure 6.35, from 143 to 999 nodes and from 2411 to 3403 nodes, the set of KRT and RT structures achieve a 0% *Global Classification Error*. Therefore, among the extracted structures, the *RT* with 143 nodes is the most suitable structure to produce fast security classification. The TS performance evaluation results for this classification structure are presented in Table 6.9.

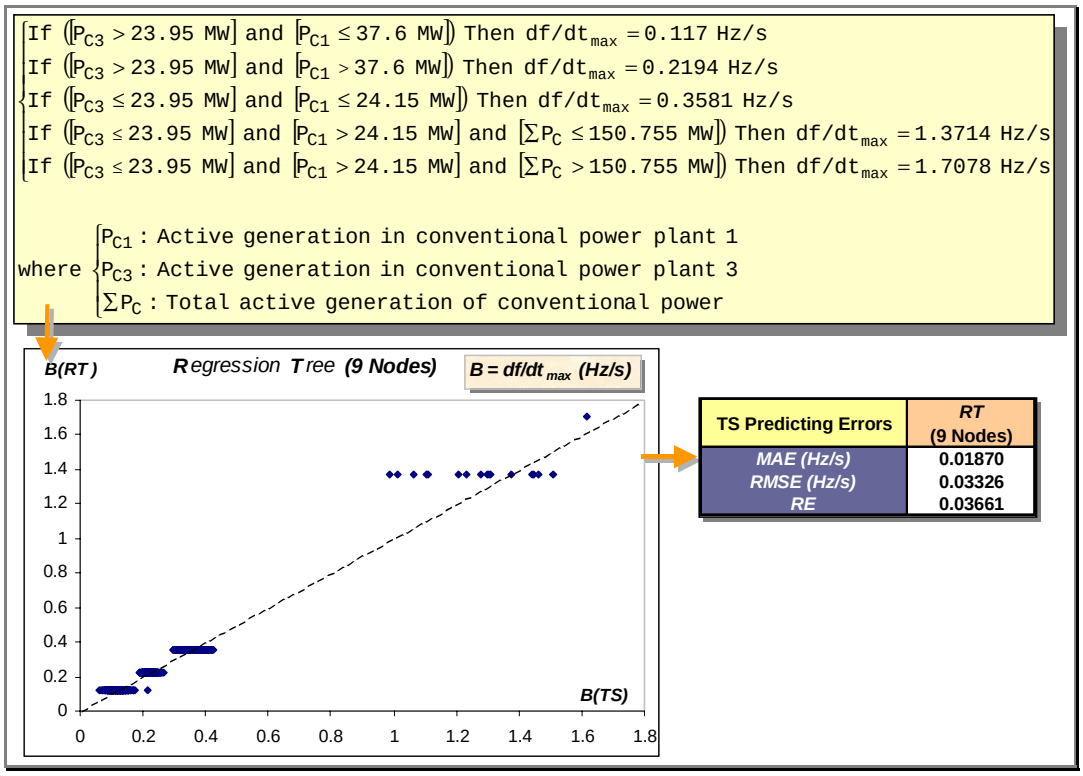


Figure 6.34 – Regression rules and TS regression errors for the obtained *RT* with 9 nodes *($B_{2,Crete}$)*

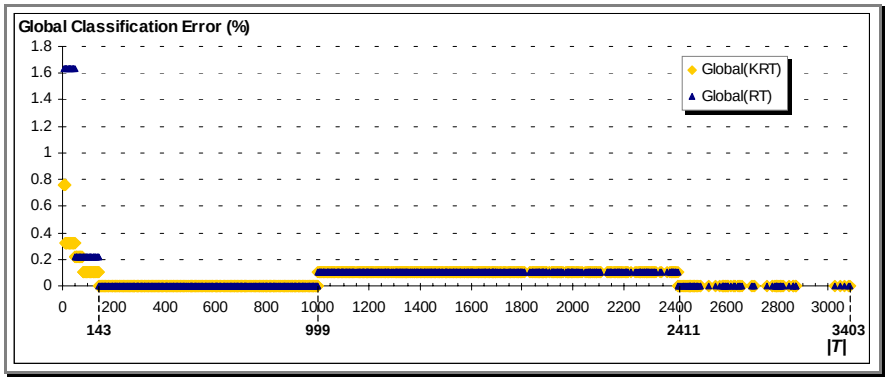


Figure 6.35 – Comparing *Global Classification Error* between the obtained {*KRT*} and {*RT*} *($B_{2,Crete}$)*

2. As it can be seen in Figure 6.36, among {*RT*}, the *RT* with 49 nodes achieves a good compromise between classification error and complexity. Moreover, its classification structure provides 2 *If rules*. Therefore, the *RT* with 49 nodes is considered suitable to extract classification rules. Its equivalent classification rules and TS classification errors are presented in Figure 6.37.

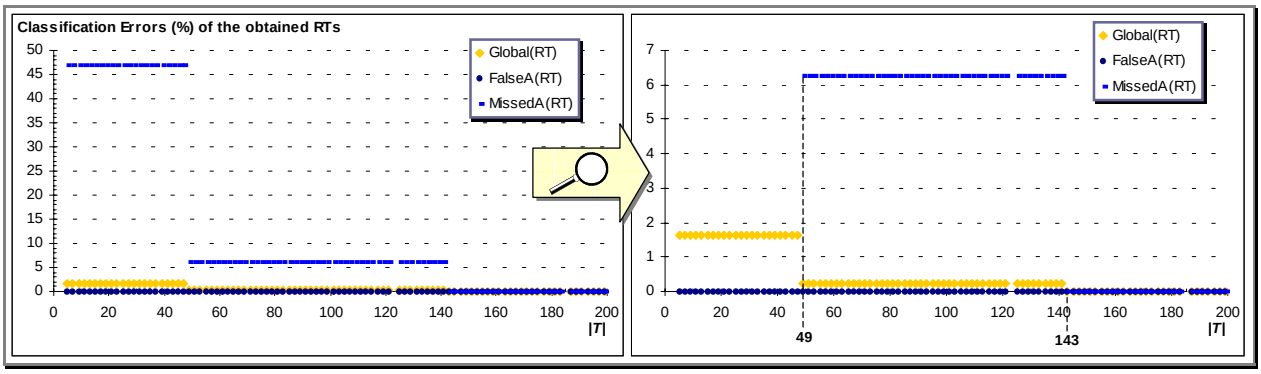


Figure 6.36 – TS classification errors for the obtained {*RT*} ($B_{2,Crete}$)

TS Predicting Errors	*RT* (49 Nodes)
Global	0.22% (2 OPs)
False Alarm	0.00%
Missed Alarm	6.25% (2 OPs)

If ($P_{C3} \leq 23.95$ MW and $P_{C1} > 24.15$ MW)
or If ($P_{C3} \leq 23.95$ MW and $P_{C1} \leq 24.15$ MW and $\Sigma P_W > 54.35$ MW) Then "Insecure"
else "Secure"

where
ΣP_W : Total active generation of wind power
P_{C1} : Active generation in conventional power plant 1
P_{C3} : Active generation in conventional power plant 3

Figure 6.37 – Classification rules and TS classification errors for the *RT* with 49 nodes ($B_{2,Crete}$)

Summarizing:

According to the required function of the hybrid regression tree, among the generated set of pruned trees $\{T\}$, the structures presented in Table 6.9 were selected.

Table 6.9 – Selected hybrid regression trees ($B_{2,Crete}$)

Machine Loss, df/dt_{max}								
Function	**Selected structure**	**MAE error**	**RMSE error**	**RE error**	**Global error**	**False Alarm error**	**Missed Alarm error**	**Prediction time (s)**
on-line evaluation of security degree	*KRT* $_{MMT}$ (25 nodes; K=3)	0.0046	0.0159	0.0084	-	-	-	*2.86E-02*
extract interpretable regression rules	*RT* with 9 nodes	0.0187	0.0333	0.0366	-	-	-	-
fast security classification	*RT* with 143 nodes	-	-	-	0.00%	0.00%	0.00%	*1.63E-05*
extract interpretable classification rules	*RT* with 49 nodes	-	-	-	0.22% (2 OPs)	0.00%	6.25% (2 OPs)	-
Nº of Insecure OPs in the TS		32						
Nº of Secure OPs in the TS		889						

6.3.2.2 Results Obtained with the ANN Method (B_2 of Crete)

In Figure 6.38, the TS regression and classification errors obtained for the trained ANN are presented.

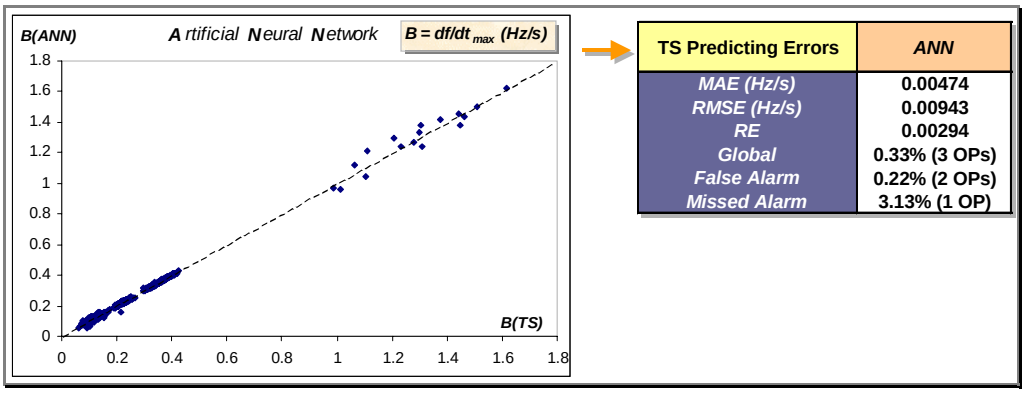


Figure 6.38 – TS errors for the trained *ANN* ($B_{2,Crete}$)

6.3.3 B_3 of Crete Case

6.3.3.1 Results Obtained with the HRT Method (B_3 of Crete)

In this case, by applying the pruning algorithm described in Chapter 5, a sequence $\{T\}$ of 1032 pruned trees was generated, where $T_1 \succ T_2 \succ \ldots \succ T_{1031} \succ root$, having T_1 3341 nodes.

Regarding the use of the extracted RT and KRT structures to produce emulation of the B_3 security index, after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. For this case, KRT_{MMT} – the most suitable structure to produce on-line evaluation of security degree – has 205 nodes (see Figure 6.39). The results of the TS performance evaluation for this structure are presented in Figure 6.40.

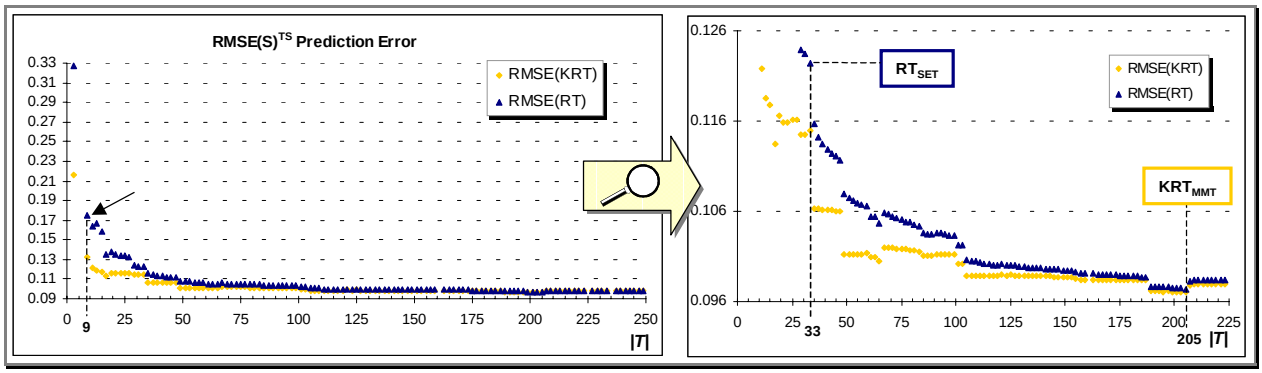


Figure 6.39 - Comparing $RMSE^{TS}$ error between the obtained {KRT} and {RT} ($B_{3,Crete}$)

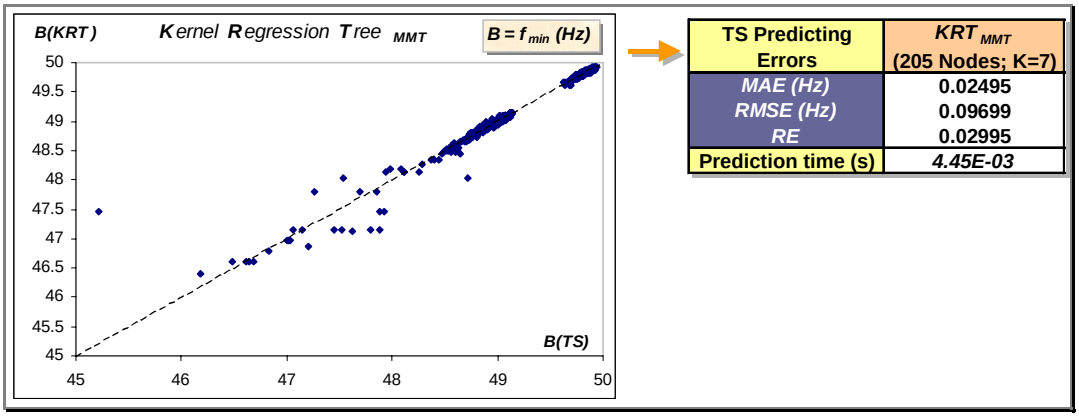


Figure 6.40 – TS performance evaluation results for the obtained KRT_{MMT} ($B_{3,Crete}$)

2. As it can be also seen in Figure 6.39, for this example RT_{SET} has 33 nodes (17 leafs), being thus too complex to be translated into comprehensible regression rules. The RT with 9 nodes (5 leafs) is considered much more suitable to extract simple regression rules. This structure is the one that verifies the *4 SE rule*. Its equivalent regression rules and TS regression errors are presented in Figure 6.41.

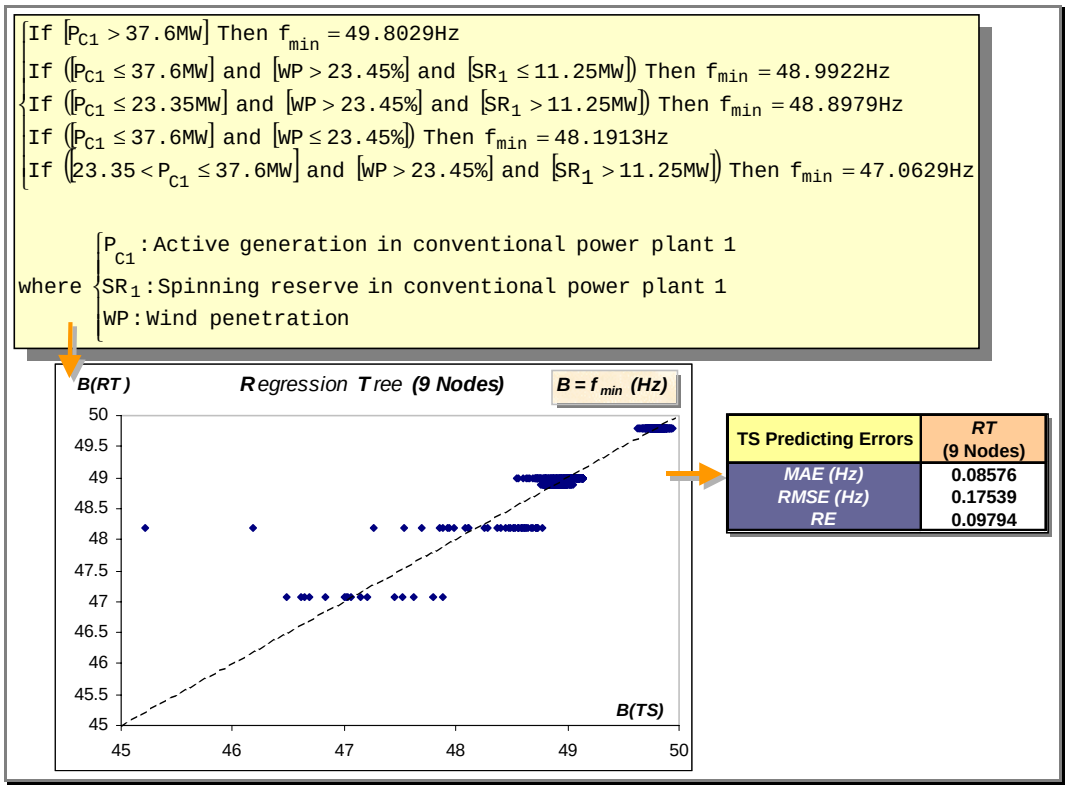


Figure 6.41 – Regression rules and TS regression errors for the obtained *RT* with 9 nodes ($B_{3,Crete}$)

Regarding the use of the extracted RT and KRT structures to classify the B_3 security index as "secure/insecure", after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. As it can be seen in Figure 6.42, from the pruned tree structure with 129 nodes to the one with 141 nodes, the resulting KRT structures achieve minimum *Global Classification Error*. Regarding this and the false and missed alarms presented in Figure 6.43, among the extracted structures, the *KRT* with 141 nodes is considered suitable to produce fast security classification. The results of the TS performance evaluation for this structure are presented in Figure 6.43.

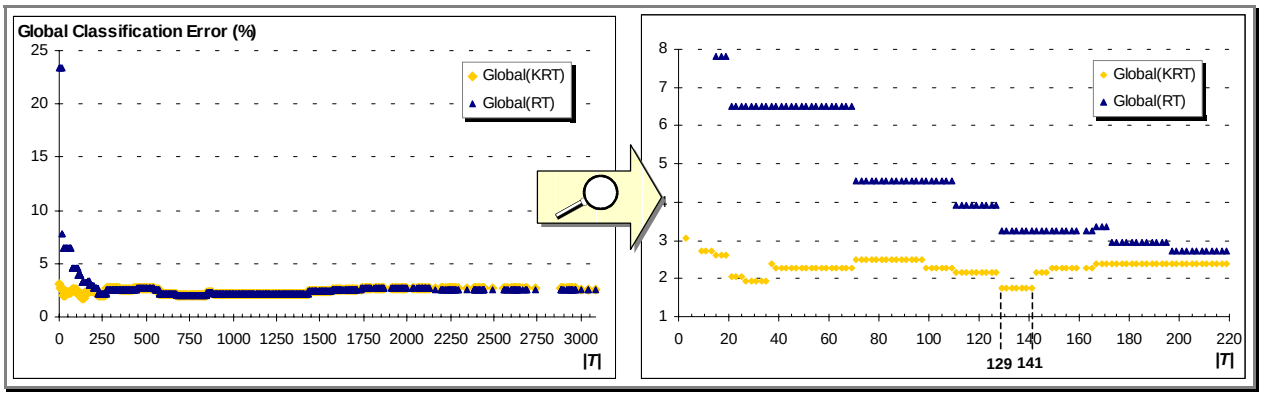


Figure 6.42 – Comparing *Global Classification Error* between the obtained {*KRT*} and {*RT*} ($B_{3,Crete}$)

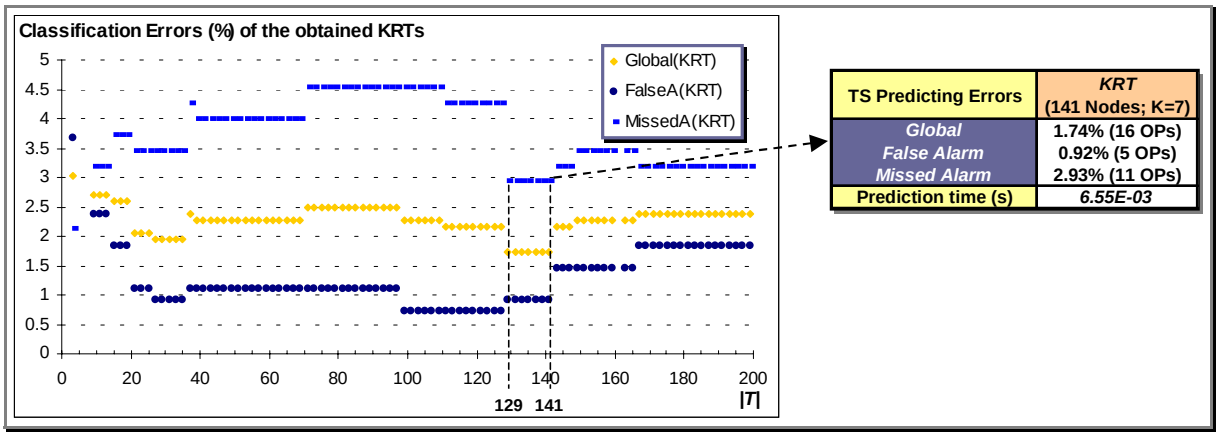


Figure 6.43 – TS classification errors for the obtained {*KRT*} ($B_{3,Crete}$)

2. As it can be seen in Figure 6.45, among {*RT*}, the *RT* with 71 nodes achieves a good compromise between classification error and complexity. Moreover, its classification structure provides 4 *If rules*. Therefore, the *RT* with 71 nodes is considered suitable to extract classification rules. Its equivalent classification rules are presented in Figure 6.44, and the TS classification errors are presented in Figure 6.45.

$$
\begin{cases}
\text{If } [P_{C1} > 37.6\text{MW}] \\
\text{or If } \left(\begin{array}{l} [23.35 < P_{C1} \le 37.6\text{MW}] \text{ and } [WP > 23.45\%] \text{ and } [SR_1 > 11.25\text{MW}] \\ \text{and } [P_{C2} > 41.6\text{MW}] \end{array} \right) \\
\text{or If } \left(\begin{array}{l} [P_{C1} \le 37.6\text{MW}] \text{ and } [WP > 23.45\%] \text{ and } [SR_1 \le 11.25\text{MW}] \\ \text{and } [\sum P_L \le 145.745\text{MW}] \end{array} \right) \\
\text{or If } \left(\begin{array}{l} [P_{C1} \le 37.6\text{MW}] \text{ and } [WP > 25.05\%] \text{ and } [SR_1 \le 11.25\text{MW}] \\ \text{and } [145.745 < \sum P_L \le 148.395\text{MW}] \end{array} \right)
\end{cases}
\text{Then "Secure"}
$$

$$
\text{else "Insecure"}
$$

$$
\text{where } \begin{cases}
P_{C1} : \text{Active generation in conventional power plant 1} \\
SR_1 : \text{Spinning reserve in conventional power plant 1} \\
P_{C2} : \text{Active generation in conventional power plant 2} \\
WP : \text{Wind penetration} \\
\sum P_L : \text{Total active load}
\end{cases}
$$

Figure 6.44 – Classification rules extracted by the obtained *RT* with 71 nodes ($B_{3,Crete}$)

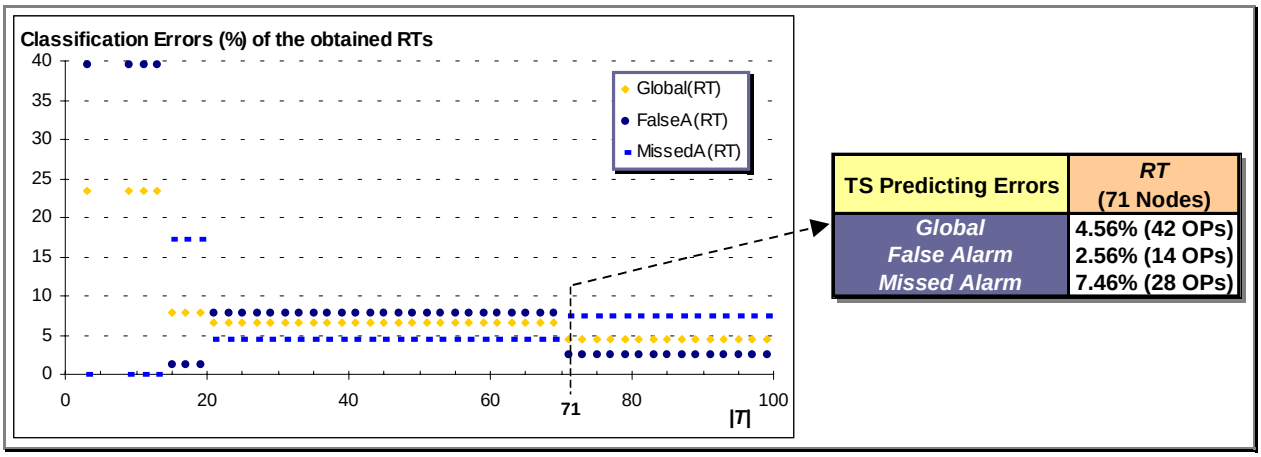


Figure 6.45 – TS classification errors for the obtained $\{RT\}$ $(B_{3,Crete})$

Summarizing:

According to the required function of the hybrid regression tree, among the generated set of pruned trees $\{T\}$, the structures presented in Table 6.10 were selected.

Table 6.10 – Selected hybrid regression trees $(B_{3,Crete})$

Short-Circuit, f_{min}								
Function	**Selected structure**	**MAE error**	**RMSE error**	**RE error**	**Global error**	**False Alarm error**	**Missed Alarm error**	**Prediction time (s)**
on-line evaluation of security degree	*KRT* $_{MMT}$ (205 Nodes; K=7)	0.0249	0.0970	0.0299	-	-	-	*4.45E-03*
extract interpretable regression rules	*RT* with 9 nodes	0.0858	0.1754	0.0979	-	-	-	-
fast security classification	*KRT* (141 Nodes; K=7)	-	-	-	1.74% (16 OPs)	0.92% (5 OPs)	2.93% (11 OPs)	*6.55E-03*
extract interpretable classification rules	*RT* with 71 Nodes	-	-	-	4.56% (42 OPs)	2.56% (14 OPs)	7.46% (28 OPs)	-
Nº of Insecure OPs in the TS		375						
Nº of Secure OPs in the TS		546						

6.3.3.2 Results Obtained with the ANN Method (B_3 of Crete)

In Figure 6.46, the TS regression and classification errors obtained for the trained ANN are presented.

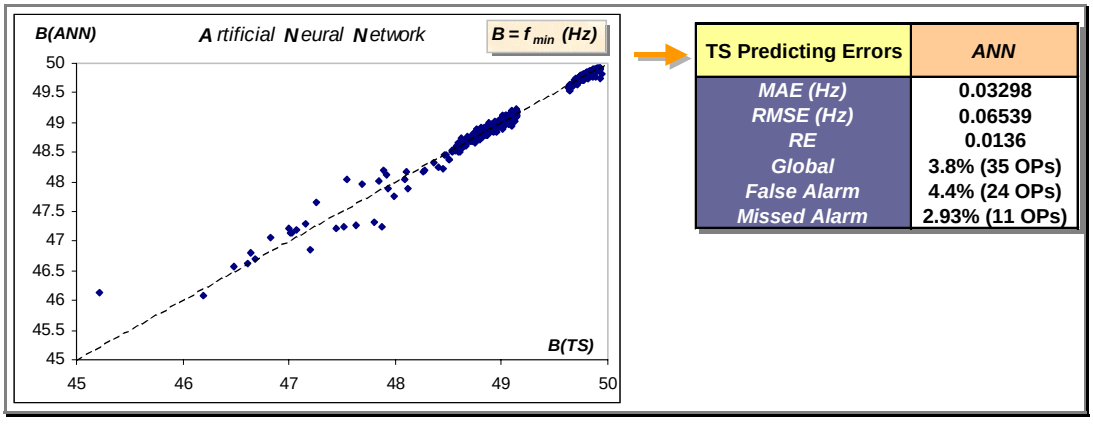


Figure 6.46 – TS errors for the trained *ANN* $(B_{3,Crete})$

6.3.3.3 Results Obtained with the Provided DT (B_3 of Crete)

The tree structure and TS classification errors of the DT provided by the NTUA researchers are presented in Figure 6.48. From this DT, which has 23 nodes, the following classification rules can be extracted:

$$\begin{cases} \text{If } \left(\left[P_{C1} \geq 37.2\text{MW}\right] \text{ and } \left[\text{WP} < 37.85\%\right]\right) \\ \text{or If } \left(\begin{matrix}\left[P_{C1} < 37.2\text{MW}\right] \text{ and } \left[25.89 \leq \text{WP} < 37.85\%\right] \text{ and } \left[8.6 \leq SR_1 < 11.2\text{MW}\right] \\ \text{and } \left[\sum P_L < 148.8\text{MW}\right]\end{matrix}\right) \\ \text{or If } \left(\begin{matrix}\left[P_{C1} \leq 37.2\text{MW}\right] \text{ and } \left[\text{WP} < 25.89\%\right] \text{ and } \left[P_{C3} \geq 24.1\text{MW}\right] \\ \text{and } \left[\sum P_C < 148\text{MW}\right]\end{matrix}\right) \end{cases} \text{Then "Secure"}$$

$$\text{else "Insecure"}$$

$$\text{where } \begin{cases} P_{C1} : \text{Active generation in conventional power plant 1} \\ SR_1 : \text{Spinning reserve in conventional power plant 1} \\ P_{C3} : \text{Active generation in conventional power plant 3} \\ \text{WP} : \text{Wind penetration} \\ \sum P_C : \text{Total active generation of conventional power} \\ \sum P_L : \text{Total active load} \end{cases}$$

Figure 6.47 – Classification rules extracted by the *DT* provided by NTUA *($B_{3,Crete}$)*

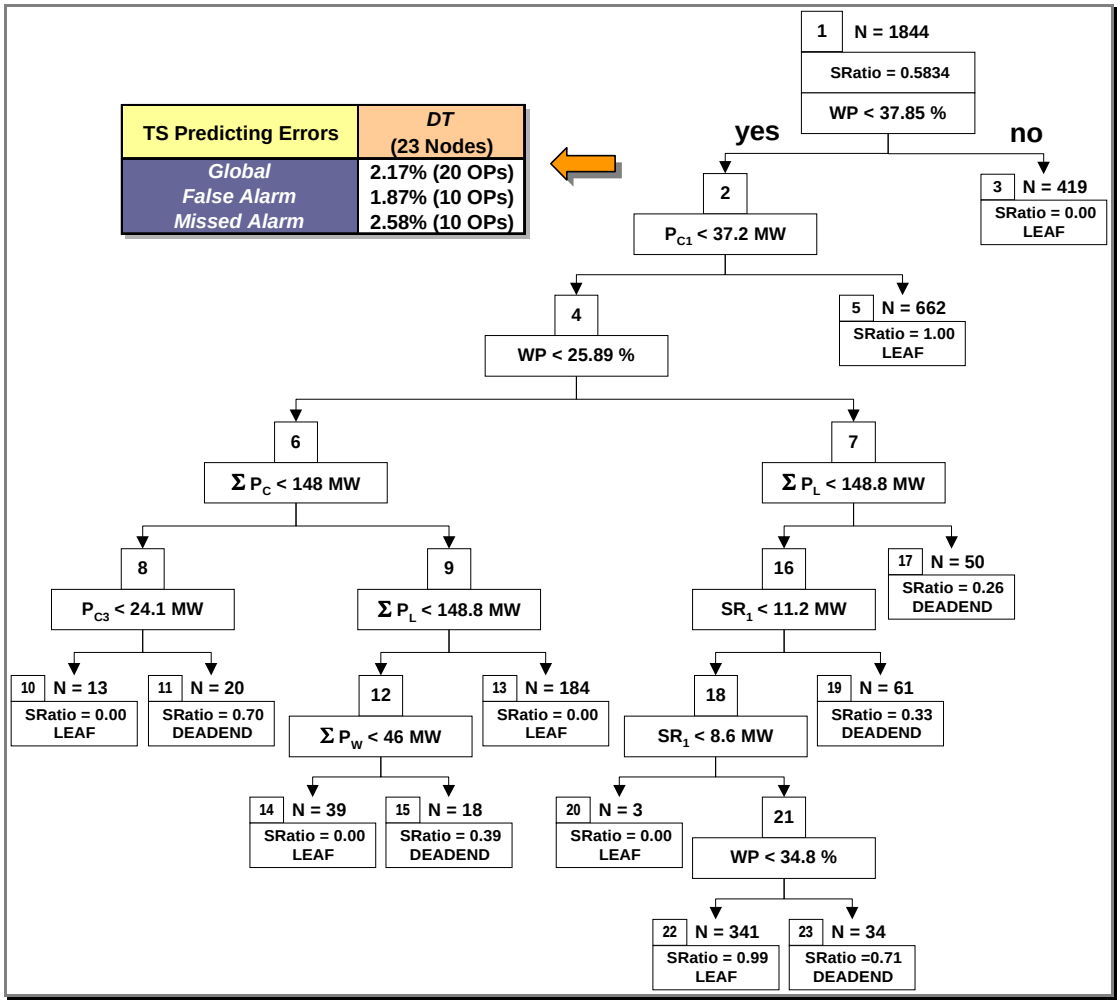


Figure 6.48 – Tree structure and TS classification errors for the *DT* provided by NTUA *($B_{3,Crete}$)*

6.3.4 B_4 of Crete Case

6.3.4.1 Results Obtained with the HRT Method (B_4 of Crete)

In this case, by applying the pruning algorithm described in Chapter 5, a sequence $\{T\}$ of 1299 pruned trees was generated, where $T_1 \succ T_2 \succ \ldots \succ T_{1298} \succ root$, having T_1 3579 nodes.

Regarding the use of the extracted RT and KRT structures to produce emulation of the B_4 security index, after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. For this case, KRT_{MMT} – the most suitable structure to produce on-line evaluation of security degree – has 43 nodes (see Figure 6.49). The results of the TS performance evaluation for this structure are presented in Figure 6.50.

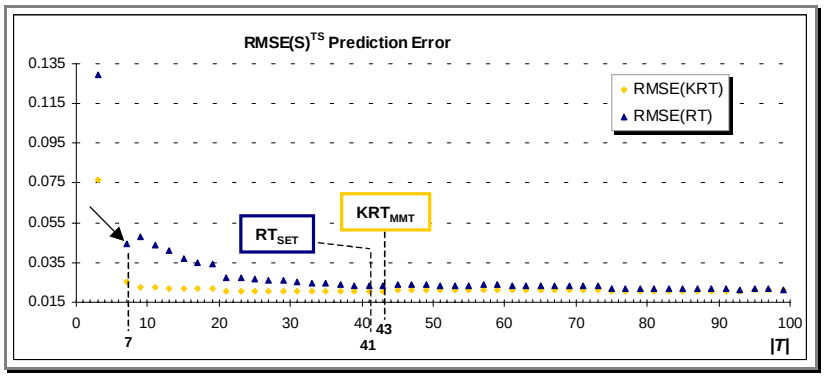


Figure 6.49 - Comparing $RMSE^{TS}$ error between the obtained {*KRT*} and {*RT*} ($B_{4,Crete}$)

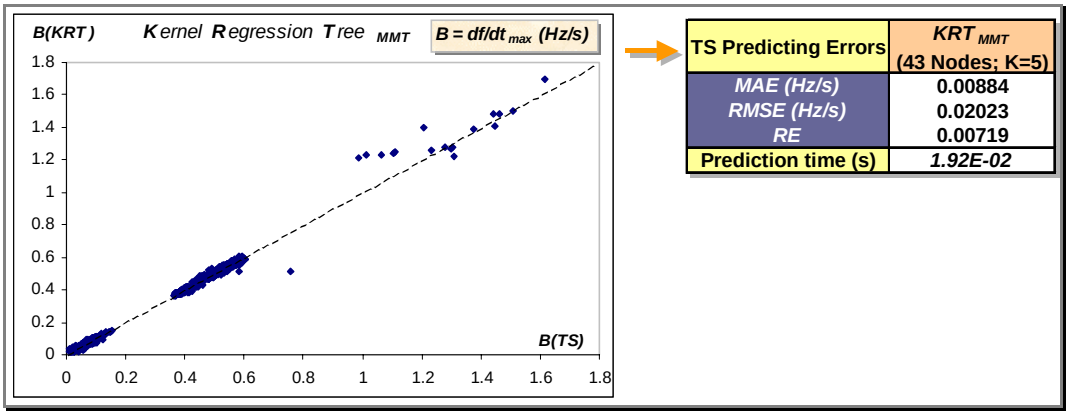


Figure 6.50 – TS performance evaluation results for the obtained KRT_{MMT} ($B_{4,Crete}$)

2. As it can be also seen in Figure 6.49, for this example, RT_{SET} has 41 nodes (21 leafs), being thus too complex to be translated into comprehensible regression rules. The *RT* with 7 nodes (4 leafs) is considered much more suitable to extract simple regression rules. This structure is the one that verifies the *13 SE rule*. Its equivalent regression rules and TS regression errors are presented in Figure 6.51.

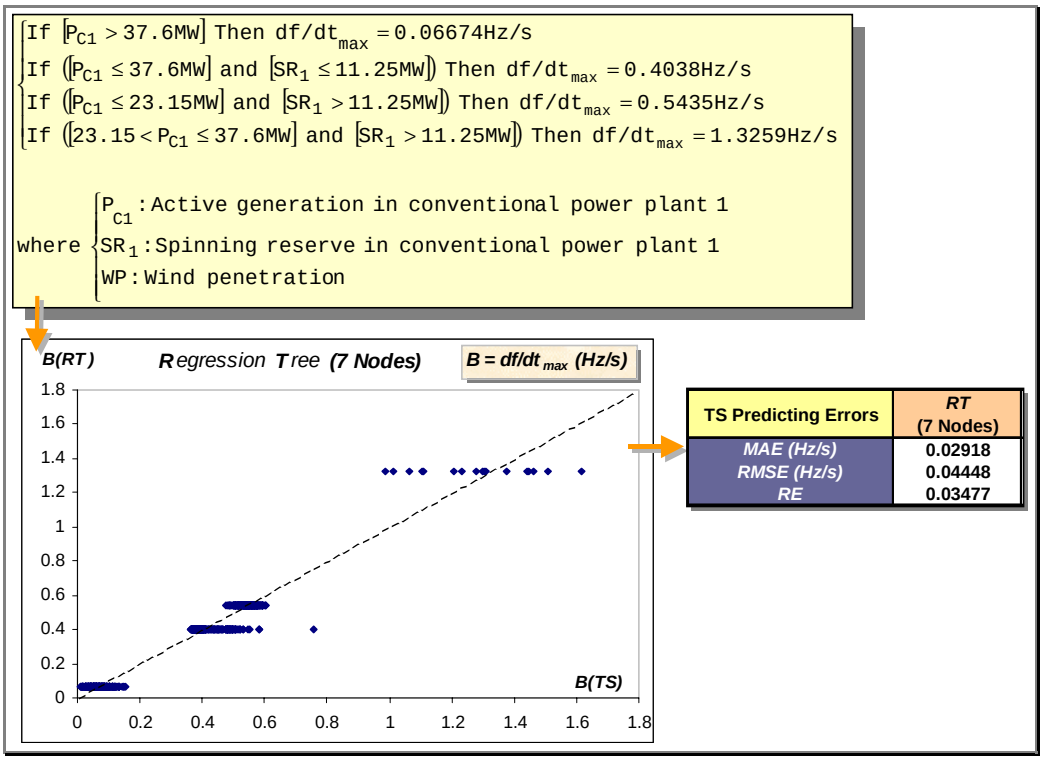


Figure 6.51 – Regression rules and TS regression errors for the obtained *RT* with 7 nodes ($B_{4,Crete}$)

Regarding the use of the extracted RT and KRT to classify the B_4 security index as "secure/insecure", after analyzing the obtained set of pruned trees one can derive the following main conclusions:

1. As it can be seen in Figure 6.52, from 1381 to 1851 nodes, the set of KRT structures achieve a minimum *Global Classification Error*. Regarding this and also the false alarms and missed alarms presented in Figure 6.53 and Figure 6.54, among the extracted structures, the *KRT* with 1851 nodes is considered suitable to produce fast security classification. The TS performance evaluation results for this structure are presented in Table 6.11.

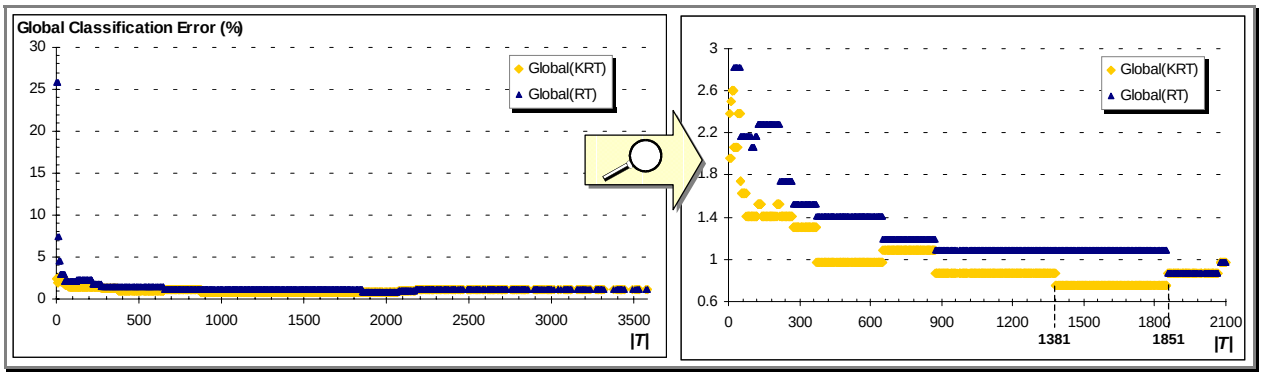


Figure 6.52 – Comparing *Global Classification Error* between the obtained {*KRT*} and {*RT*} ($B_{4,Crete}$)

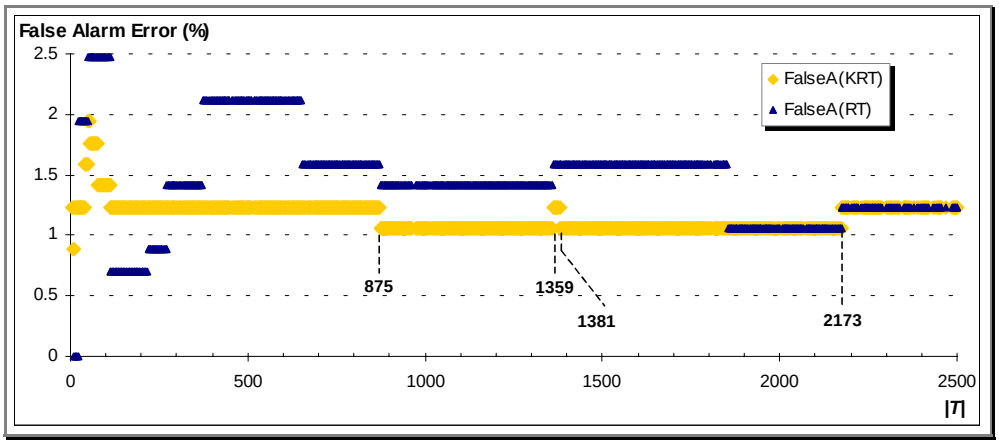


Figure 6.53 – Comparing *False Alarm Error* between the obtained {*KRT*} and {*RT*} ($B_{4,Crete}$)

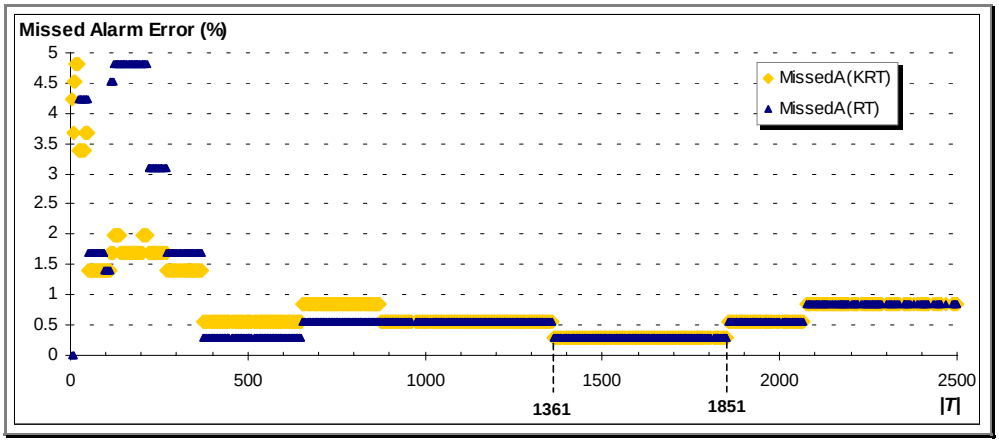


Figure 6.54 – Comparing *Missed Alarm Error* between the obtained {*KRT*} and {*RT*} ($B_{4,Crete}$)

2. As we can see in Figure 6.55, among {*RT*}, the *RT* with 51 nodes achieves a good error/complexity compromise. Moreover, its classification structure provides 3 *If rules*. Therefore, the *RT* with 51 nodes is considered suitable to extract classification rules. Its equivalent classification rules and TS classification errors are presented in Figure 6.56.

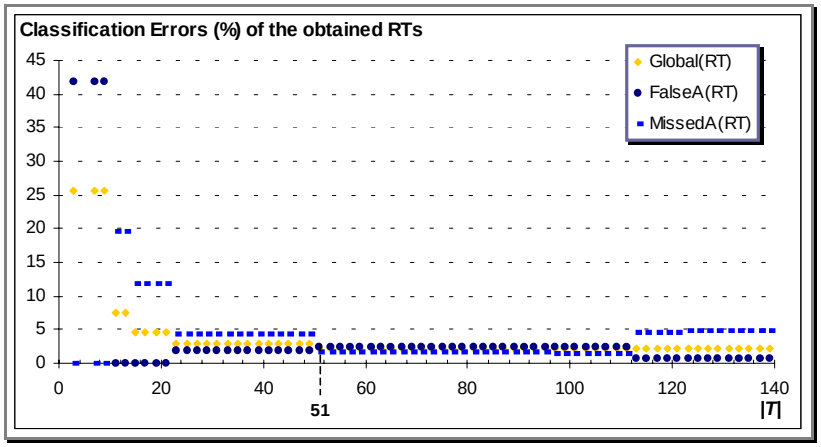


Figure 6.55 – TS classification errors for the obtained {*RT*} ($B_{4,Crete}$)

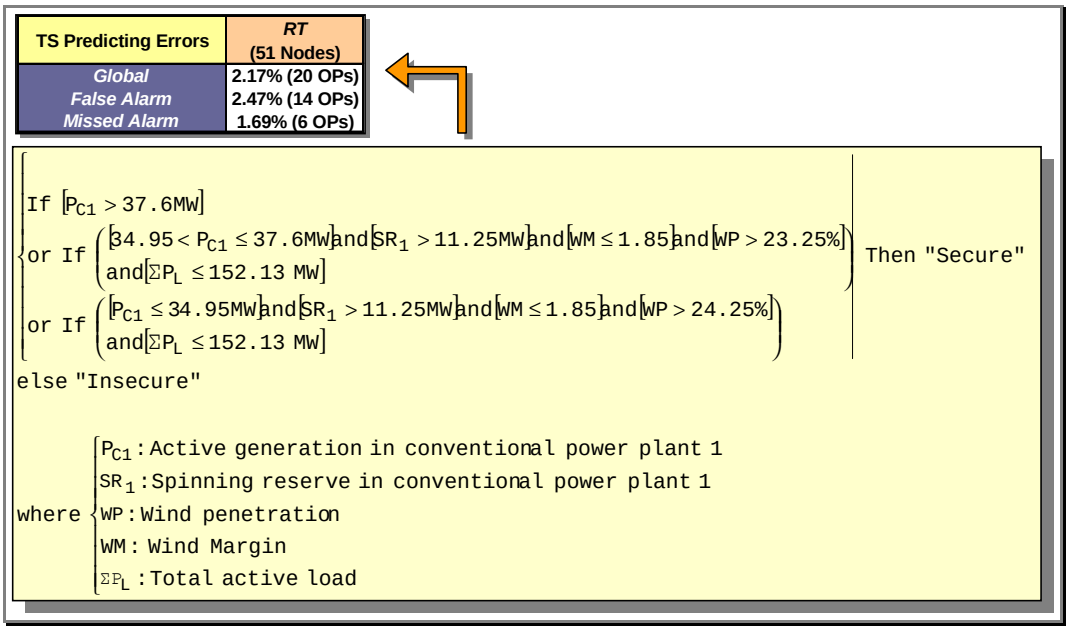


Figure 6.56 – Classification rules and TS classification errors for the *RT* with 51 nodes *($B_{4,Crete}$)*

Summarizing:

According to the required function of the hybrid regression tree, among the generated set of pruned trees $\{T\}$, the structures presented in Table 6.11 were selected.

Table 6.11 – Selected hybrid regression trees *($B_{4,Crete}$)*

Machine Loss, df/dt_{max}								
Function	**Selected structure**	**MAE error**	**RMSE error**	**RE error**	**Global error**	**False Alarm error**	**Missed Alarm error**	**Prediction time (s)**
on-line evaluation of security degree	*KRT* $_{MMT}$ (43 nodes; K=5)	0.0088	0.0202	0.0072	-	-	-	*1.92E-02*
extract interpretable regression rules	*RT* with 7 nodes	0.0292	0.0445	0.0348	-	-	-	-
fast security classification	*KRT* with 1851 Nodes and K=5	-	-	-	0.76% (7 OPs)	1.06% (6 OPs)	0.28% (1 OP)	*9.01E-04*
extract interpretable classification rules	*RT* with 51 nodes	-	-	-	2.17% (20 OPs)	2.47% (14 OPs)	1.69% (6 OPs)	-
Nº of Insecure OPs in the TS		354						
Nº of Secure OPs in the TS		567						

6.3.4.2 Results Obtained with the ANN Method (B_4 of Crete)

In Figure 6.57, the TS regression and classification errors obtained for the trained ANN are presented.

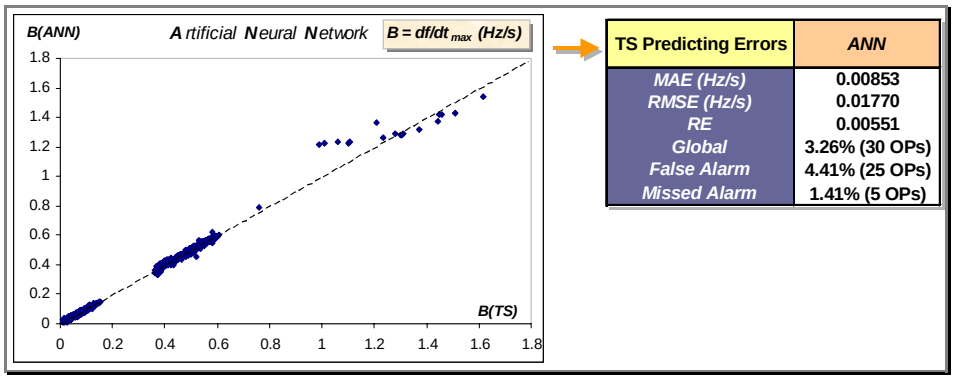


Figure 6.57 – TS errors for the trained *ANN ($B_{4,Crete}$)*

6.4 Conclusions

In this Section the conclusions, achieved by analyzing the results obtained for the Terceira and Crete power systems, are presented. First, a comparative assessment is made between the results obtained with the three applied automatic learning methods – HRT (Hybrid Regression Tree), ANN (Artificial Neural Network) and DT (Decision Tree). Then, a few last conclusions are presented regarding the functions provided by these three methods.

6.4.1 Comparative Assessment

6.4.1.1 B_3 of Terceira Case

Regarding the application of the AL structures to produce fast security classification of the $B_{3,Terceira}$ security index, the TS predicting errors obtained from the HRT and DT methods are presented in Figure 6.58. From these results, the following can be observed:

1. The selected *HRT* reaches lower classification errors than the provided *DT*.

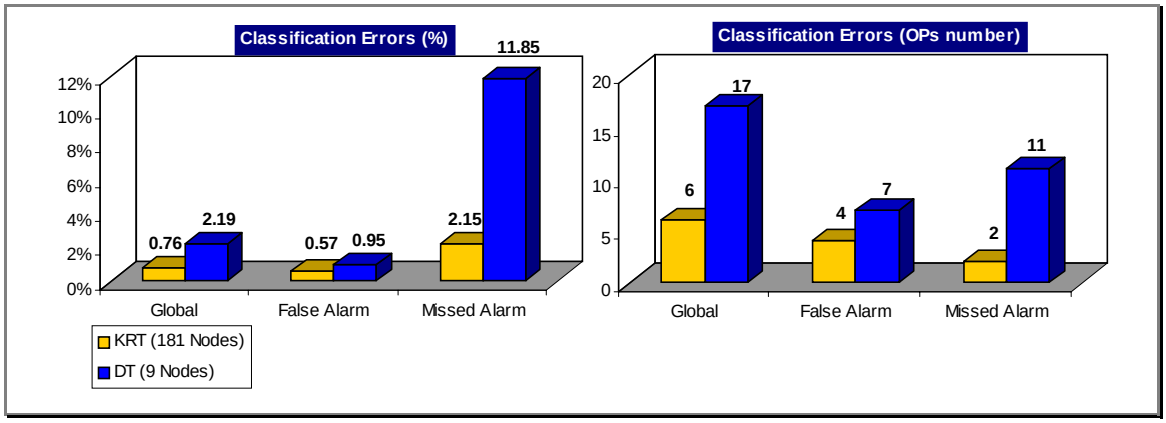


Figure 6.58 – Comparative assessment regarding fast security classification of $B_{3,Terceira}$

Regarding the extraction of classification rules to the $B_{3,Terceira}$ security index, the TS predicting errors obtained from the HRT and DT approaches are presented in Figure 6.59. From the obtained results, the following can be observed:

1. The selected *HRT* reaches lower classification errors than the provided *DT*.
2. The *DT* has a simpler classification structure. In fact, the *HRT* provides 3 *If rules*, whereas the *DT* provides 1 *If rule* (see Figure 6.23 and Figure 6.25).
3. Both techniques were capable of selecting approximately the same attributes as being the most important ones.

   Namely, the classification rules provided by the *HRT* structure include operating conditions provided by the following parameters:

> $\sum$ *active losses - total active losses;*
> SR_{GIV} *- spinning reserve in Diesel unit GIV;*
> $P_{D,GVI}$ *- active generation in Diesel unit GVI.*

The classification rules provided by the *DT* structure include operating conditions provided by the following parameters:

> $\sum$ *active losses - total active losses;*
> SR_{GIV} *- spinning reserve in Diesel unit GIV;*
> SR_{GVI} *- spinning reserve in Diesel unit GVI.*

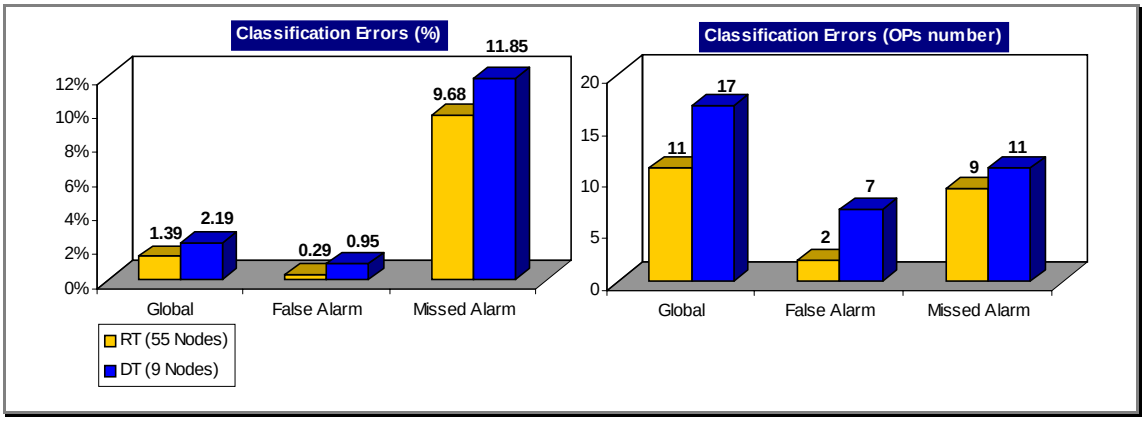


Figure 6.59 – Comparative assessment regarding the extraction of classification rules to $B_{3,Terceira}$

6.4.1.2 B_1 of Crete Case

Regarding the application of the AL structures to produce the emulation of the $B_{1,Crete}$ security index, the TS predicting errors obtained from the HRT and ANN approaches are presented in Figure 6.60. As it can be seen from these results:

1. The selected *HRT* reaches lower *MAE* and *RMSE* errors than the provided *ANN*.

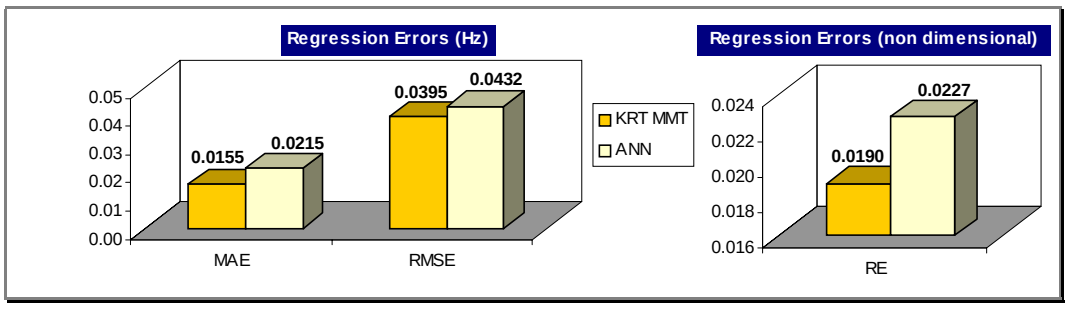


Figure 6.60 – Comparative assessment regarding the emulation of $B_{1,Crete}$

Regarding the application of the AL structures to produce fast security classification of the $B_{1,Crete}$ security index, the TS predicting errors obtained from the HRT and ANN approaches are presented in Figure 6.61. From these results, the following can be observed:

1. Both approaches have no false alarms.
2. The selected *HRT* reaches a higher missed alarm error than the trained *ANN*.

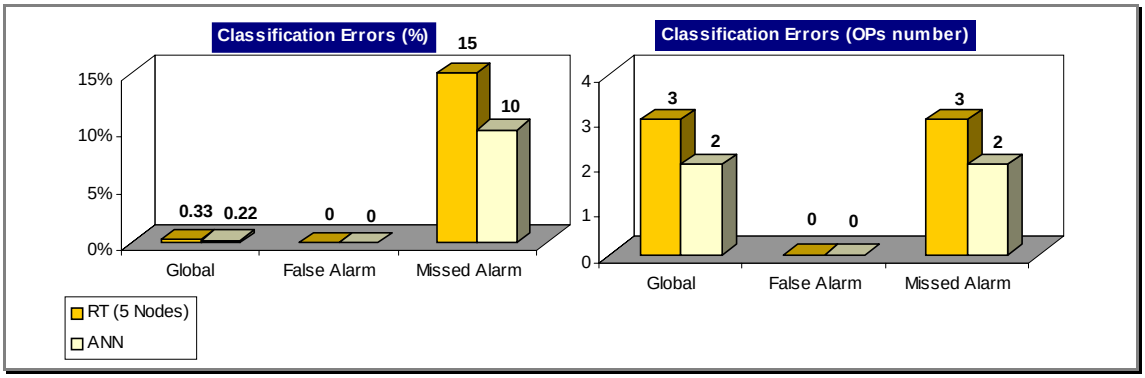


Figure 6.61 – Comparative assessment regarding fast security classification of $B_{1,Crete}$

6.4.1.3 B_2 of Crete Case

Regarding the application of the AL structures to produce the emulation of the $B_{2,Crete}$ security index, the TS predicting errors obtained from the HRT and ANN approaches are presented in Figure 6.62. As it can be seen from these results:

1. The selected *HRT* reaches the lowest *MAE* error, whereas the provided *ANN* reaches the lowest *RMSE* error.

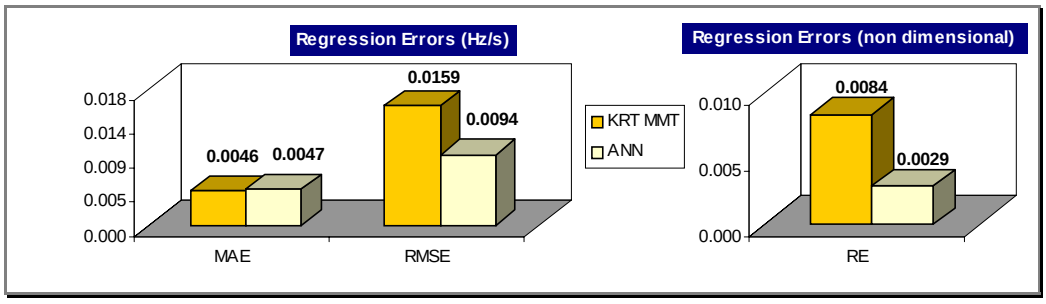


Figure 6.62 – Comparative assessment regarding the emulation of $B_{2,Crete}$

Regarding the application of the AL structures to produce fast security classification of the $B_{2,Crete}$ security index, the TS predicting errors obtained from the HRT and ANN approaches are presented in Figure 6.63. From these results, the following can be observed:

1. The extracted *HRT* reaches no classification errors, whereas the provided *ANN* has 2 false alarms and 3 missed alarms.

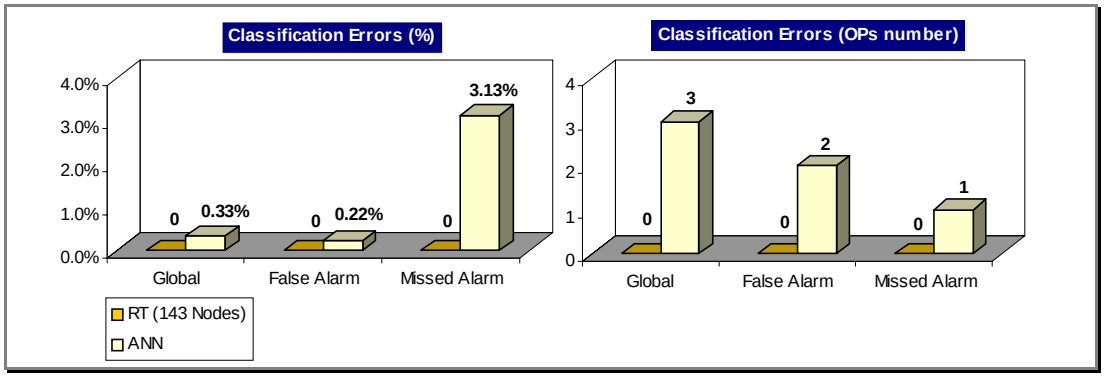


Figure 6.63 – Comparative assessment regarding fast security classification of $B_{2,Crete}$

6.4.1.4 B_3 of Crete Case

Regarding the application of the AL structures to produce the emulation of the $B_{3,Crete}$ security index, the TS predicting errors obtained from the HRT and ANN approaches are presented in Figure 6.64. As it can be seen from these results:

1. The selected *HRT* reaches the lowest *MAE* error, whereas the provided *ANN* reaches the lowest *RMSE* error.

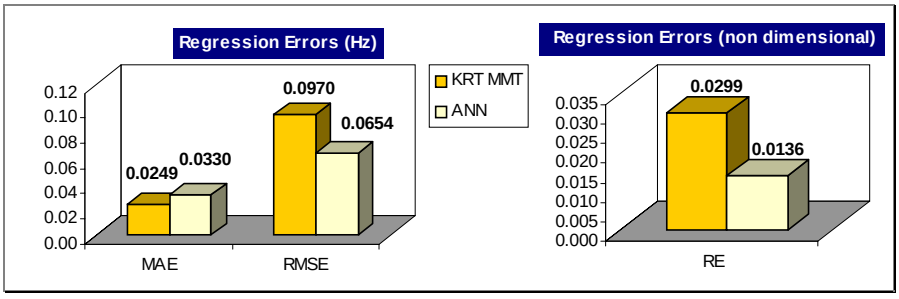


Figure 6.64 – Comparative assessment regarding the emulation of $B_{3,Crete}$

Regarding the application of the AL structures to produce fast security classification of the $B_{3,Crete}$ security index, the TS predicting errors obtained from the HRT, DT and ANN approaches are presented in Figure 6.65. From these results the following can be observed:

1. The *HRT* and *DT* reach lower classification errors than the *ANN*.
2. The *HRT* provides the lowest global and false alarms, whereas the *DT* reaches a slightly better missed alarm error.

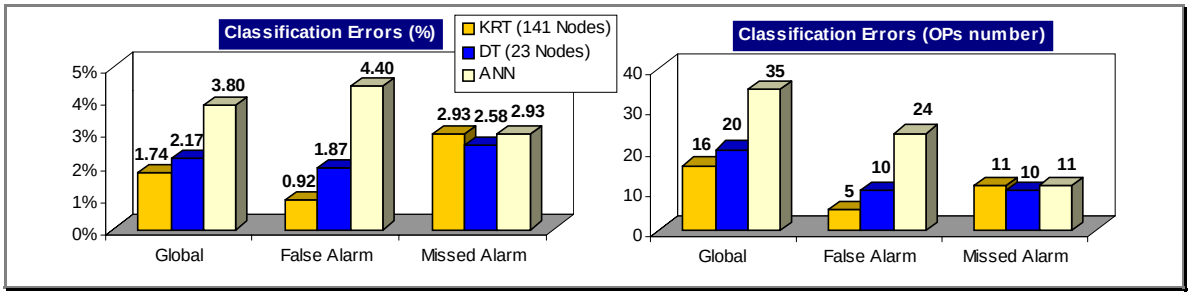


Figure 6.65 – Comparative assessment regarding fast security classification of $B_{3,Crete}$

Regarding the extraction of classification rules to the $B_{3,Crete}$ security index, the TS predicting errors obtained from the HRT and DT approaches are presented in Figure 6.66. From the obtained results, the following can be observed:

1. The selected *HRT* reaches higher classification errors than the provided *DT*.
2. The *DT* has a simpler classification structure. In fact, the *HRT* provides 4 *If rules*, whereas the *DT* provides 3 *If rules* (see Figure 6.45 and Figure 6.47).
3. Both techniques were capable of selecting approximately the same attributes as being the most important ones.

   Namely, the classification rules provided by the *HRT* structure include operating conditions provided by the following parameters:

   > P_{C1} *and* SR_1 *- active generation and spinning reserve in conventional power plant 1;*
   > P_{C2} *- active generation in conventional power plant 2;*
   > *WP - wind penetration;*
   > $\sum P_L$ *- total active load.*

   The classification rules provided by the *DT* structure include operating conditions provided by the following parameters:

   > P_{C1} *and* SR_1 *- active generation and spinning reserve in conventional power plant 1;*
   > P_{C3} *- active generation in conventional power plant 3;*
   > *WP - wind penetration;*
   > $\sum P_L$ *- total active load;*
   > $\sum P_C$ *- total active generation of conventional power.*

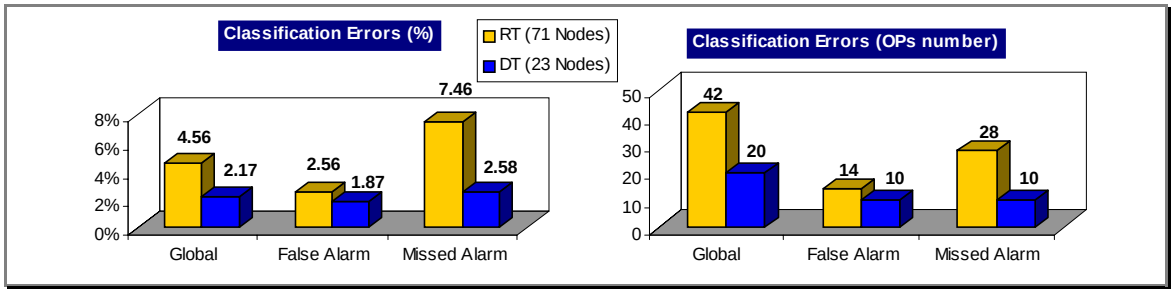


Figure 6.66 – Comparative assessment regarding the extraction of classification rules to $B_{3,Crete}$

6.4.1.5 B_4 of Crete Case

Regarding the application of the AL structures to produce the emulation of the $B_{4,Crete}$ security index, the TS predicting errors obtained from the hybrid RT and ANN approaches are presented in Figure 6.67. As it can be seen from these results:

1. The provided *ANN* reaches lower *MAE* and *RMSE* errors than the selected *HRT*.

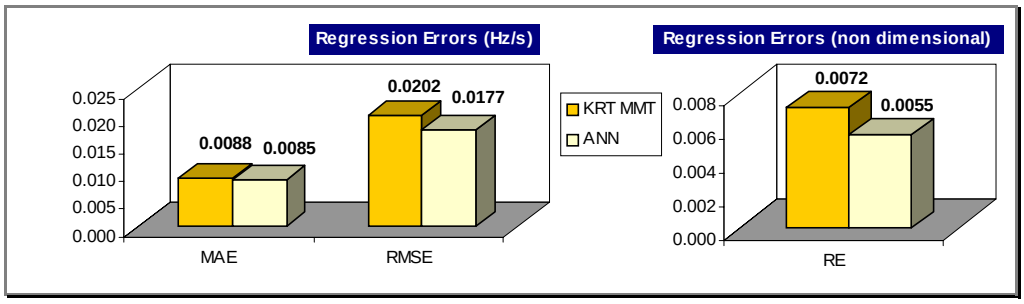


Figure 6.67 – Comparative assessment regarding the emulation of $B_{4,Crete}$

Regarding the application of the AL structures to produce fast security classification of the $B_{4,Crete}$ security index, the TS predicting errors obtained from the HRT and ANN approaches are presented in Figure 6.68. As it can be seen from these results:

1. The extracted *HRT* achieves lower classification errors than the provided *ANN*.

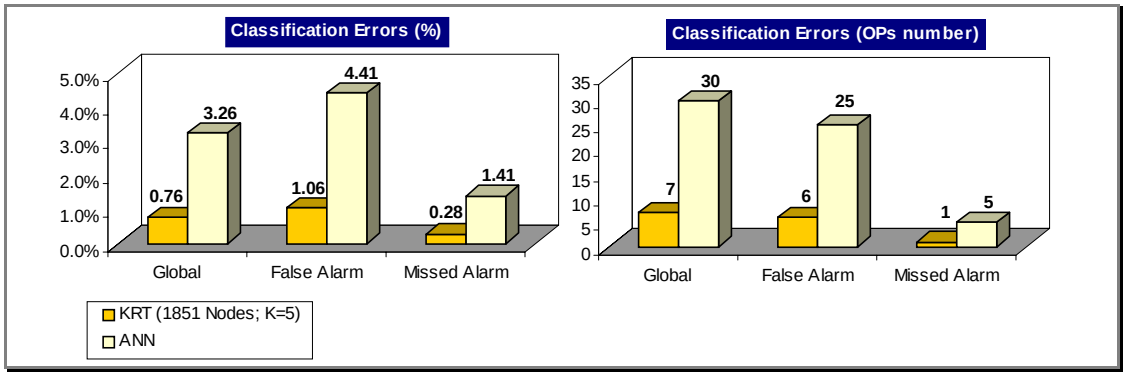


Figure 6.68 – Comparative assessment regarding fast security classification of $B_{4,Crete}$

6.4.2 Functions Provided by the HRT, ANN and DT Approaches

1. The HRT and ANN methods have the advantage of producing simultaneously a classification structure and giving the degree of robustness of the system, whereas the DT method can only perform security classification.

2. The HRT and DT methods can provide interpretable rules of the system security class (i.e., classification rules), whereas ANNs always provide quite opaque models of the data. The HRT method can still provide interpretable rules of the system security degree (i.e., regression rules).

3. For the two cases where a comparative assessment was made between the HRT and DT methods – for the B_3 security index of the Terceira and Crete cases -, the following was observed:

   - The DT always provided a simpler classification structure.

   - For the B_3 security index of the Terceira case, the HRT showed to be the most accurate approach to perform fast security classification and to provide classification rules.

   - Regarding the fast security classification for the B_3 security index of the Crete case, it was not possible to identify one of the approaches as being the most accurate one. In fact, the HRT provides the lowest global and false alarms, whereas the DT reaches a slightly better missed alarm error.

   - Since the provided rules are simpler and achieve smaller errors, the DT showed to be more suitable to extract classification rules to the B_3 security index of the Crete case.

4. From the results obtained for the Crete case, where a comparative assessment was made between the HRT and the ANN methods for the 4 security indices, the following was observed:

   - Regarding the evaluation of the system security degree, the HRT approach showed to be more accurate for the B_1 security index, whereas the ANN showed to be more accurate for the B_4 security index. For the emulation of B_2 and B_3, as the HRT reaches the lowest *MAE* error and the ANN reaches the lowest *RMSE* error, it was not possible to identify one of the approaches as being the most accurate one.

   - Regarding fast security classification, the HRT approach showed to achieve smaller classification errors for the B_2, B_3 and B_4 security indices, whereas the ANN showed to be more accurate for the B_1 security index. However, it is important to mention that the

provided ANNs were trained to provide emulating of the security indices, and therefore, they weren't trained for classification purposes.

It should be remarked that each of the provided ANN is able to emulate, at the same time, both f_{min} and df/dt_{max} values that succeed from a disturbance. On the other hand, each of the provided HRT structures is able to emulate only one security index. Thus, to obtain with the HRT method a prediction for both f_{min} and df/dt_{max} values that succeed from a disturbance, two different HRT structures are necessary to be used.

5. All the three approaches showed suitable performances to be integrated in an advanced control system as the one that was developed within the European CARE project.

7 General Conclusions

In this last Chapter, the general conclusions obtained from the work reported in this Master thesis are presented. First, the achievements of this research are provided. Then, some comments related to the implemented Hybrid Regression Tree (HRT) method and with its perspectives of development are presented.

7.1 Achievements of this Research

From the research work reported in this document, a new hybrid automatic learning technique, named as Hybrid Regression Tree, was implemented to make, for the first time, dynamic security assessment of power system in the field of frequency stability problems.

Within the framework of the European R&D project JOULE/THERMIE, the implemented HRT approach was integrated within the advanced control system that is being installed, during the present year, on the energy management center of Crete island, to perform dynamic security assessment functions. This advanced control system is a prototype of the CARE system – an advanced control system that aims to achieve optimal utilization of renewable energy sources, in a wide variety of medium and large size isolated systems with diverse structures and operating conditions [8].

In the Crete control system, the HRT technique is being applied for the first time to provide dynamic security assessment of power systems. In this control center, the results obtained with the HRT module have shown to be very promising. Namely, recently (in 23 of June 1999) a machine loss incident occurred in the Crete island, having the dynamic security assessment module predicted a minimum frequency deviation similar to the one that actually occurred in the power system [57].

Besides being tested on the Crete power system, the implemented HRT approach was also tested on a future foreseen scenario for the electrical power systems of the Terceira island.

For these two study cases, a performance evaluation of the obtained results and a comparative assessment with Decision Tree and Artificial Neural Network approaches was performed. From this analysis, the HRT showed to provide good predicting structure whose performance stands up to the performance of the two other existent methods.

7.2 Hybrid Regression Tree Implemented Method

7.2.1 Main Conclusions

1. The HRT implemented method enables a coherent evaluation of the robustness of the system by providing:

   a) fast evaluation of the degree of security by emulating, through the regression functions, the continuous security indices that quantify/classify the power system dynamic behavior to a pre-defined disturbance;

   b) a way of clarifying security by presenting a set of regression and classification rules to be interpreted and used for explanatory purposes.

2. Regarding the design of a HRT structure by using only stop-splitting rules, the pruning process has the main advantage of allowing the choice of the best tree structure according to the desired performances.

3. Among the RT and KRT variants of the HRT method, the last one was capable to provide the smallest MSE^{TS} error, for all the cases studied of the Terceira and Crete power systems. Consequently, in the sense of accuracy, the KRT approach was invariably most suitable to produce on-line evaluation of the system security degree, for all the considered security indices. Thus, to perform this security assessment function, the selected structure among the obtained {*RT*} and {*KRT*} was always KRT_{MMT} (i.e., the KRT that minimizes the MSE^{TS} error).

   For the obtained KRT_{MMT} structures, the estimated values of their response time to predict a security index for one OP is quite small (in the order of milliseconds). Therefore these structures are suitable for on-line implementation.

4. Regarding the application of a HRT structure to make fast security classification of the Terceira and Crete power systems, to select a proper structure among the obtained {*RT*} and {*KRT*}, it was necessary to make a comparative analysis between the TS performances provided by all the obtained structures. Specifically, it was necessary to analyze their classification errors and prediction time. For the B_1 and B_2 security indices of the Crete case, it was possible to identify the structure with the best accuracy performances. However, for the remaining analyzed cases of Terceira and Crete, a selection among a set of possible solutions was necessary to be performed. A way to make automatically this selection could be by solving a 4-criterion problem, where these criterions would be to minimize the global error, false alarm error, missed alarm errors, and the prediction time.

For the HRT structures selected to make fast security classification of the Terceira and Crete cases, the estimated values of their response time to predict a security index for one OP are quite small (in the order of milliseconds). Therefore, like the selected KRT_{MMT} structures, these structures are also suitable for on-line implementation.

5. Regarding the extraction of interpretable security rules, among the KRT and RT variants, only the last one could be used. In fact, the use of a KRT structure is not feasible to perform this function, since it does not assign a constant value for the prediction in the tree leafs.

6. Regarding the extraction of interpretable regression rules for the Terceira and Crete cases, the selection of a RT structure obtained through the application of the *1 SE rule*[13], showed to provide too complex regression structures. In fact, by applying this rule, RTs with a number of nodes from 23 to 195 were selected, and therefore providing a number of *If regression rules* from 12 to 98.

To select a proper RT structure, it was necessary to make a comparative analysis between the TS performances provided by the obtained RT structures. Specifically, to obtain a good accuracy/interpretability trade-off, it was necessary to analyze their regression error MSE^{TS} and the complexity of their regression structure. Note that, the complexity of the regression structure of a RT depends on its number of leafs (since it gives the number of *If rules*) and from their depth to the root node (since it gives the maximum number of attributes that can be included in each *If rule*).

By following this procedure, the actually selected RTs resulted in having a number of nodes within 7 and 11 (providing a number of *If regression rules* from 4 to 6). Instead of verifying the *1 SE rule*, the selected structures verify from the *4* to the *42 rule*. This means that these are the smallest structures which MSE^{TS} is within:

$$MSE(RT_{MMT})^{TS} + n \times \text{standard error estimation of } MSE(RT_{MMT})^{TS}$$

where for the best case $n = 4$ and for the worst one $n = 42$. Obviously, this resulted in the loss of some accuracy relatively to the structure RT_{MMT} that minimizes the regression error.

7. Regarding the extraction of classification rules for the Terceira and Crete cases, to select a proper RT structure, it was also necessary to make a comparative analysis between the TS performances provided by all the obtained RT structures. Specifically, to obtain a good accuracy/interpretability trade-off, it was necessary to analyze their classification errors and

[13] The application of this rule allows choosing the simplest tree whose accuracy is comparable to RT_{MMT} (i.e., the RT among $\{RT\}$ that minimizes the regression error $MSE(RT)^{TS}$).

the complexity of their classification structure. Note that the complexity of the classification structure of a RT doesn't necessarily depend on its number of leafs. In fact, although being large, the selected trees for the Terceira and Crete analyzed cases provide a number of *If classification rules* that goes from 1 to 3.

A way to perform automatically this selection could be by solving a 4-criterion problem, where these criterions would be to minimize the global error, false alarm error, missed alarm errors, and the complexity of the classification rules.

8. For all the studied cases, the RT approach showed to provide simple and efficient regression and classification rules. As previously explained, these extracted security rules can be easily understood, discussed, and eventually adopted by the operators to define new operating strategies. Therefore, by using this machine learning technique, much of the manual task to extract the operating guidelines can be performed automatically by a systematic methodology.

7.2.2 Perspectives of Development

In the field of machine learning techniques and kernel regression models, many different approaches are proposed in the literature. Some of them, which are considered interesting by the author of this document to be included in the implemented HRT method, are presented in this Section. Note that in order to select the approach that achieves the best performances in the field of dynamic security assessment, experimental comparative studies would be necessary to be performed, by applying those approaches to different power systems.

7.2.2.1 Use a More Accurate Model to Grow the Tree Structure of a KRT

In the growing algorithm of the tree structure presented in Section 5.2.1 (by using stop-splitting rules) and in Section 5.2.3 (by using a pruning algorithm) the function $f_t(OP)$ used to make prediction in the tree leafs was considered to be the mean value of B. As the KRT resulting structure uses a more accurate model $f_t(OP)$ to make prediction in the tree leafs (i.e., a kernel regression model), an obvious question is why not using that same prediction model during tree growing when the goal is to extract a KRT structure.

This would perhaps bring some gains in accuracy to the designed KRT structure. However, one difficulty of this approach certainly would be an increased computational effort during the learning step. However, since this step is performed off-line, this difficulty is not considered a problem in the context of dynamic security assessment.

7.2.2.2 Making Kernel Regression Prediction with Feature Weighting

Besides considering a kernel function to estimate the weight of each neighbor to *Q* (as a function to its distance to *Q*), to make prediction Luís Torgo in [3] also assigns a weight, w_i, to each attribute. These weights give an estimation of the influence that each attribute (also named as features) has on the regression problem to solve, being included in the calculation of the distance between OPs. By applying this approach, the considered distance function presented in Equation 5.11 changes into:

$$D(Q,OP) = \sqrt{\sum_{i=1}^{Na} w_i \times d_i(Q,OP)^2} \qquad (7.1)$$

In this proposed approach, an algorithm is applied to the data set that calculates a vector $W = [w_1,...,w_{Na}]$ with the estimation of the quality of each attribute. This algorithm implements a method called *RReliefF*, presented by Robnik-Sikonja and Kononenko [58], which estimates attributes quality in the context of regression. This method does not assume independence between attributes and, therefore, as claimed in [58], can correctly estimate the quality of attributes in problems with strong dependence between attributes.

Another, also considered interesting experience to perform, would be to use the set of weights $(w_1,...,w_{Na})$ extracted from the data set to make feature extraction before the learning step.

7.2.2.3 Using "Oblique" Splitting Tests

The splitting test used to grow the tree structure presented in Equation 5.1 (Section 5.2.1) is applied on a single attribute, being thus perpendicular to the applied attribute axes and therefore named as orthogonal splits. A proposed variant is presented in CART [4], which consists on the following:

> *If a linear structure is suspected to exist between attributes, to the set of allowable orthogonal splits "*$\{a_k(sample) > threshold\}$*? ", there should be included all the linear combination splits of the form:*

$$\left\{\sum_{i=1}^{Na} w_i a_i > threshold\right\}? \qquad (7.2)$$

where:

the set of weights $w = (w_1, ...,w_{Na})$ must verify $\|w\|^2 = \sum_{i=1}^{Na} w_i^2 = 1$

the threshold value must range over all possible values

By using these oblique splits, besides finding the optimal *threshold value*, the splitting procedure at each test node must also search for the optimal set of weight values. An efficient searching algorithm is proposed in CART.

This approach is based in the assumption that, if a linear structure between candidate attributes really exists, then oblique splits can provide a more efficient separation between the learning samples. Note that two obviously difficulties of this approach are the increased computational effort introduced during the learning and predicting step, and also the possible loss of some interpretability of the design RT structure.

REFERENCES

[1] Hatziargyriou, N., Peças Lopes, J. A., Karapidakis, E., Vasconcelos, M. H., *"On-Line Dynamic Security Assessment of Power Systems in Large Islands with High Wind Power Penetration"*, in Proceedings of PSCC´99 - 13th Power Systems Computation Conference, vol. 1, pages 331-337, Trondheim - Norway, June 1999.

[2] Vasconcelos, M. H., Peças Lopes, J. A., *"Pruning Kernel Regression Trees for Security Assessment of the Crete Network"*, in Proceedings of ACAI´99 Workshop on Applications of Machine Learning, Chania – Greece, July 1999.

[3] Torgo, L., *"Kernel Regression Trees"*, poster papers of the European Conference, on Machine Learning (ECML-97), Internal Report of the Faculty of Informatics and Statistics, University of Economics, Prague, 1997.

[4] Breiman, L., Friedman, H. F., Olshen, R. A., Stone, C. J., *"Classification and Regression Trees"*, Wadsworth International, 1984.

[5] Watson, G. S., *"Smooth Regression Analysis"*, Sankhya: The Indian Journal of Statistics, 1964.

[6] Nadaraya, E., *"On estimating regression"*, Theory of Probability and its Applications, 1964.

[7] ARMINES, NTUA, INESC, RAL, PPC, *"Development and implementation of an advanced control system for the optimal operation and management of medium-sized power systems with a large penetration from renewable power sources"*, Final report of EU-DG XII JOULE II project JOU2-CT92-0053. Edited by the Office for Official Publications of the European Communities, Luxembourg 1996.

[8] *"CARE: Advanced Control Advice for power systems with large-scale integration of Renewable Energy sources"*, contract JOR3-CT96-0119, 4th Project Report (Specifications of the CARE System Software), September 1998.

[9] Peças Lopes, J. A., *"Operation of Isolated Systems with a Large Penetration of Wind Power Generation"*, V SEPOPE, Recife, May 1996.

[10] Jansénio Delgado, *"Energia Eólica em Cabo Verde: Experiências e Perspectivas" (Wind Energy in Cape Verde: Experiences and Perspectives)*, A Corrente - Revista da Empresa Pública da Electricidade e Água de Cabo Verde (ELECTRA), pages 18-22, April 1996.

[11] *"CARE: Advanced Control Advice for power systems with large-scale integration of Renewable Energy sources"*, contract JOR3-CT96-0119, 2nd Project Report (Steady State and Dynamic Analysis of Study Case Islands), July 1997.

[12] Peças Lopes, J. A., Matos, M. A., Pereira da Silva, J. L., Van Acker, V., *"Estudo do impacto da ligação de parques eólicos na rede eléctrica da ilha da Madeira" (Impact study of wind parks connection in the electrical network of Madeira island)*, Final report of a consultancy project developed by INESC-Porto for AREAM, May 1995.

[13] Peças Lopes, J. A., Vasconcelos, M. H., Monteiro, C., *"Estudo do impacto da ligação de novos parques eólicos na rede eléctrica da ilha da Madeira" (Impact study of the connection of new wind parks in the electrical network of Madeira island)*, Final report of a consultancy project developed by INESC-Porto for EEM, October 1998.

[14] Peças Lopes, J. A., Vasconcelos, M. H., Monteiro, C., *"Integration of Wind Generation in the Terceira Electrical Network – Preliminary results from the steady state and dynamic analysis"*, in the 2nd Project Report of the ENN project Joule, contract JOR3-CT96-0119, July 1997.

[15] Peças Lopes J. A., Vasconcelos, M. H., Pereira da Silva, J.L., *"Relatório Preliminar dos Estudos de Análise da Rede Eléctrica da Ilha de S. Vicente da República de Cabo Verde" (Preliminary Report of the Analyze Studies for the Electrical Network of S.Vicente Island of the Republic of Cape Verde)*, Report of a consultancy project developed by INESC-Porto for ELECTRA, March 1997.

[16] Vasconcelos, M. H., Peças Lopes, J. A., Monteiro, C., *"Estudos de Análise da Rede Eléctrica da Ilha de Santiago da República de Cabo Verde" (Analyze Studies for the Electrical Network of Santiago Island of the Republic of Cape Verde)*, Final report of a consultancy project developed by INESC-Porto for ELECTRA, February 1998.

[17] Vasconcelos, M. H., Peças Lopes, J. A., *"Estudos de Análise da Rede Eléctrica da Ilha do Sal da República de Cabo Verde" (Analyze Studies for the Electrical Network of Sal Island of the Republic of Cape Verde)*, Final report of a consultancy project developed by INESC-Porto for ELECTRA, December 1998.

[18] Kundur, P., *"Power Systems Stability and Control"*, McGraw – Hill, 1993 ISBN 0-07-035958-x.

[19] Matos, M. A., Vlachos, A., Bakirtzis, T., Androutsos, A., Gigantidou, A., *"A unit commitment/dispatch scheme for isolated systems with a large penetration of renewables"*, submitted to PSCC´99 - 13th Power Systems Computation Conference, 1998.

[20] Fink, L. H., Carlsen, K., *"Operating under stress and strain"*, IEEE spectrum, March 1978.

[21] Laplace, P.S., *"Mémoire sur les approximations des formules qui sont des fonctions de très grands nombres et sur leur application aux probablités"*, Mémoires de l'Académie des Sciences de Paris, 1810.

[22] Gauss, K. F., *"Theoria combinationis observatorionum errroribus minimis obnoxiae"*, Dietrich, Göttingen, 1826.

[23] McCulloch, W. S., Pitts, W., *"A logical calculus of ideas immanent in nervous activity"*, Bulletin of Mathematical Biophysics, 1943.

[24] Hunt, E. B., Marin, J., Stone, P.J., *"Experiments in Induction"*, Wiley, 1966.

[25] Draper, N. R., Smith, H., *"Applied Regression Analysis"*, John Wiley, 2nd edition, 1981.

[26] Fix, E., Hodges, J. L., *"Discriminatory analysis, nonparametetric discrimination consistency properties"*, Technical Report, AT&T Bell Laboratories, 1951.

[27] Stone, C. J., *"Consistent nonparametric regression"*, Ann. Statist., 1977.

[28] Cleveland, W., *"Robust locally weighted regression and smoothing scatterplots"*, J. Amer. Statist., Assoc., 1979.

[29] Michie, D., Spigelhalter, D. J., Taylor, C. C., *"Machine Learning, Neural and Statistical Classification"*, Ellis Horwood Series in Artificial Intelligence, 1994.

[30] Broadley, C. E., *"Recursive Automatic Bias Selection for Classifier Construction"*, Machine Learning, Kluwer Academic Publishers, 1995.

[31] Dy Liacco, T. E., *"Control of Power Systems via the Multi-Level Concept"*, PhD thesis, Sys. Res. Center, Case Western Reserve Univ., 1968.

[32] Pang, C. K., Prabhakara, F. K., El-Abiad, A. H., Koivo A. J., *"Security evaluation in power systems using pattern recognition"*, IEEE Trans. on Power Apparatus and Systems, vol. PAS-93, pages 969-976, May/June 1974.

[33] Gupta, C. L., El-Abiad, A. H., *"Transient stability analysis of power systems by pattern recognition"*, in Proceeding of Midwest Power Symposium, University of Akron, Ohio, October 1975.

[34] Pao, Y. H., Dy Liacco, T. E., Bozma, I., *"Acquiring a qualitative understanding of system behavior through AI inductive inference"*, in Proceedings of IFAC Symposium on Electric Energy Systems, pages 35-41, 1985.

[35] Sobajic, D. J., Pao, Y.H., *"Artificial neural-net based dynamic security assessment for electric power systems"*, IEEE Trans. on PWRS, vol. 4, nr. 1, February 1989.

[36] El-Sharkawi, M. A., Marks II, R. J., Aggoune, M. E., Park, D. C., Damborg, M. J., Atlas, L., *"Dynamic security assessment of power systems using back error propagation artificial neural networks"*, 2nd Symposium on Expert Systems Application to power systems, pages 366-370, Seatle, July 1989.

[37] Wehenkel, L., Pavella, M., *"Decision tree approach to power system security assessment"*, Int. J. Electrical Power and Energy Systems, vol. 15, No. 1, February 1993.

[38] Hatziargyriou, N. D., Contaxis, G. C., Sideris, N. C., *"A decision tree method applied to on-line steady-state security assessment"*, IEEE PES Summer Meeting, 1993.

[39] Rovnyak, S., Kretsinger, S., Thorp, J., Brown, D., *"Decision trees for real-time transient stability prediction"*, IEEE/PES Summer meeting, 93 SM 530-6 PWRS, Vancouver, B. C. Canada, 1993.

[40] Peças Lopes, J. A., Fidalgo, J. N., Miranda, V., Hatziargyriou, N., *"Neural networks used for on-line dynamic security assessment of isolated power systems with a large penetration from wind production- A real case study"*, in Proceedings of Rough Sets and Soft Computing Conference'94, San José, USA, November 1994.

[41] Hatziargyriou, N., Papathanassiou, S., Papadopoulos, M., *"Decision Trees for Fast Security Assessment of Autonomous Power Systems with large Penetration from Renewables"*, IEEE Transactions on Energy Conversion, vol. 10, Nr. 2, June 1995.

[42] Wehenkel, L., *"Contingency severity assessment for voltage security using non-parametric regression techniques"*, IEEE Transactions on Power Systems, vol. 11, No. 1, February 1996.

[43] Peças Lopes, J. A., Fernandes, F., *"Fast Evaluation of Voltage Collapse Risk Using Machine Learning Techniques"*, in Proceedings of VI SEPOPE, S. Salvador da Baia, Brazil, May 1998.

[44] Peças Lopes, J. A., Hatziargyriou, N. D., *"Application of learning from examples methods for on-line dynamic security assessment of electric power systems – state of the art"*, Invited paper presented to the SEPOPE, Foz do Iguaçu, May 1994.

[45] Wehenkel, L., *"Tutorial Course on Automatic Learning Methods Application to Dynamic Security Assessment"*, CPSPP'97, July 1997.

[46] Rumelheart, MacClelland, *"Parallel Distributed Processing"*, University of California, MIT Press, Cambridge, London, England, 1988.

[47] Lipmand, *"An Introduction to computing with neural networks"*, IEEE ASSSP Magazine, April 1987.

[48] A Karalic, *"Employing Linear Regression in Regression Tree Leaves"*, in Proceedings of ECAI-92, Wiley & Sons, 1992.

[49] Quinlan, J. R., *"Learning with Continuos Classes"*, in Proceedings of the 5th Australian Joint Conference on Artificial Intelligence, World Scientific, 1992.

[50] Torgo, L., *"Error Estimates for Pruning Regression Trees"*, in Proceedings of the 10th European Conference on Machine Learning (ECML-98), Nedellec,C. and Rouveirol,C. (eds.). Lecture Notes in Artificial Intelligence 1398, Springer Verlag, 1998.

[51] McCalley et al., *"On-Line Visualization of transmission system operating constraints using intelligent information processing"*, Final report of contract No. Z-19-2-384-94 between Pacific Gas and Electric Company and Iowa State University, May 1997.

[52] Van Acker, V., *"Steady State and Dynamic Behaviour Analysis of Isolated Power Systems with Wind Power Production"*, Master Thesis, July 1995.

[53] Torgo, L., *"Functional Models for Regression Tree Leaves"*, in Proceedings of the International Conference on Machine Learning (ICML-97), Fisher, D.(ed.), Morgan Kaufmann Publishers, 1997.

[54] Torgo, L., *"RT programs - an users manual"*, LIACC, Machine Learning Group, Technical Report, 1997.

[55] Cleveland, W. S., Loader, C. R., *"Smoothing by Local Regression"*, Principles and Methods In computational Statistics, 1995.

[56] Atkeson, C. G., Moore, A. W., Schaal, S., (in press), *"Locally Weighted Learning"*, Special issue on lazy learning, D. Aha (Ed.), Artificial Intelligence Review, 1996.

[57] Gigantidou, A., "CARE Project Evaluation", in Proceeding of CARE Workshop, Crete –Greece, July 1999.

[58] Robnik-Sikonja, M., Kononenko, I., *"Context-sensitive attribute estimation in regression"*, in Proceedings of the ICML-96 Workshop on Learning in Context-Sensitive Domains, 1996.

[59] Van Hecke, J., *"Sequential probabilistic method for power system operation and planning"*, Électra No. 179, AUGUST 1998.

[60] Charfield, C., *"Statistics for technology"*, 3rd edition, Chapman and Hall, Ldt., 1983.

[61] Wettschereck, D., Aha, D. W., Mohri, T., (in press), *"A review and empirical evaluation of feature weighting methods for a class of lazy learning algorithms"*, Special issue on lazy learning, D. Aha (Ed.), Artificial Intelligence Review, 1996.

[62] Delicado, P., Manuel del Rio, *"Weighted Kernel Regression"*, downloaded from: http://ideas.uqam.ca/ideas/data/Papers/upfupfgen164.html