FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO




FEUP

# Manutenção de Sistemas de Geração de Energia Renovável Eólica através de Redes IP

**Inácio de Sousa Adelino da Fonseca**

Doutoramento em Engenharia Electrotécnica e de Computadores

Orientador: Fernando Maciel Barbosa
(Prof. Catedrático da Faculdade de Engenharia da Universidade do Porto)

Co-orientador: José Manuel Torres Farinha
(Prof. Coordenador do Instituto Superior de Engenharia de Coimbra)

Janeiro de 2010

© Inácio Fonseca, 2010

# Resumo

Esta dissertação desenvolve-se no âmbito da Manutenção Planeada Condicionada de Sistemas de Geração de Energias Renováveis utilizando redes IP, com ênfase na Geração Eólica. É proposto um modelo de *software*/*hardware* e uma arquitectura que permitem a implementação de soluções de planeamento de manutenção de condição, através da medição remota de várias variáveis de controlo. A abordagem utilizada para a manutenção planeada na vertente preditiva emprega preferencialmente Séries Temporais para seguir a evolução das variáveis de condição como sejam a temperatura, pressão, viscosidade e o módulo do espectro de frequências. O conhecimento destas variáveis permite acompanhar o estado de funcionamento dos equipamentos e antecipar o estado seguinte. Neste âmbito, é dada uma contribuição através de uma modificação do algoritmo do alisamento exponencial, de forma a torná-lo adaptativo. O seu desempenho é acompanhado através da Série de *Mackey-Glass* e de uma Série especificamente desenvolvida de acordo com a expectativa da evolução dos sinais a predizer. Para a determinação, a nível macroscópico, do estado da condição de funcionamento de um aero-gerador, foi utilizado o classificador SVM. Este classificador, após a fase de treino, determina, em função das medidas das diversas variáveis, o instante mais provável do estado de avaria. Ainda neste contexto foi efectuada uma breve análise das vibrações em motores de indução, na perspectiva da sua analogia com os geradores eléctricos utilizados na geração eólica.

Na vertente da optimização, propõe-se uma metodologia de apoio à decisão relativa à sequência das visitas a efectuar pelos técnicos de manutenção aos diferentes aero-geradores. Esta metodologia tem em consideração os planos de manutenção condicionada previamente definidos. A metodologia de optimização proposta utiliza algoritmos genéticos e uma solução específica para o problema da sequência das visitas. A organização da gestão da manutenção dos parques eólicos é estruturada de forma a integrar *hardware* adaptado/desenvolvido, assim como um conjunto de actualizações e de novos módulos implementados para o *software* SMIT. Esta estrutura permite o funcionamento nas vertentes da manutenção não-planeada, planeada sistemática e condicionada. Em termos de *software*, foi desenvolvido um sistema de gestão da manutenção baseado em comunicações cliente/servidor, utilizando, sempre que possível, *software open-source*. O sistema desenvolvido incluí o sistema operativo *Linux*, o motor de base de dados *PostgreSQL* e as ferramentas de desenvolvimento *Octave*, *R*, *Apache* e *PHP*. O cliente SMIT é programado em *Delphi* e dialoga com o utilizador através da plataforma *Windows*.

A nível do *hardware*, a metodologia utiliza componentes e dispositivos de baixo custo, para criar um sistema de aquisição de dados sobre redes IP. A ideia base consiste em distribuir um relógio mestre pelos diferentes equipamentos de campo, para garantir a aquisição síncrona nos vários pontos de recolha de dados. A partir dos protocolos SNTP e PTP, foram implementadas diversas técnicas de controlo para sincronização do relógio. A estrutura base do sistema utiliza nós de recolha de dados interligados através de uma rede *CAN*. Um dos nós, com conectividade *CAN* e *Ethernet*, concentra e envia a informação recolhida para o servidor SMIT. Em simultâneo, este nó controla a sequência de aquisição de dados, assim como a sincronização do relógio com o servidor. A integração global dos componentes de *hardware* e de *software* desenvolvidos pressupõe um fluxo de dados dos nós de aquisição para o servidor que, por sua vez, envia referências temporais ao nó mestre, incluindo um sinal de relógio padrão. O servidor SMIT, através de algoritmos baseados em Séries Temporais, analisa os dados recebidos utilizando a plataforma *R* ou *Octave* para predizer possíveis avarias e estados disfuncionais. Baseado nessa predição, o servidor gera antecipadamente os respectivos alertas com a emissão das correspondentes Ordens de Trabalho.

**Palavras Chave**: geradores eólicos, manutenção, redes IP, séries temporais, sistemas embebidos

# Abstract

This dissertation presents work developed in the area of Conditional Planned Maintenance of Systems for the Generation of Renewable Energies using IP networks, with emphasis in Wind Generation. A software/hardware model and an architecture are proposed, that allow the implementation of conditional planning maintenance solutions, using the remote measurement of a set of control variables. The chosen approach to planned maintenance, uses Time Series to monitor the evolution of condition variables such as temperature, pressure, viscosity and modulus of the frequency spectrum. The knowledge of these variables allows to follow the operating state of equipments and anticipate the following state. In this context, a contribution is proposed through a modified exponential smoothing algorithm, in order to make it adaptive. Its performance has been monitored by the Mackey-Glass Series and an especially developed Series, which is in accordance with the expectation for the evolution of the signals to predict. To determine, at a macroscopic level, the state of the operating condition of a wind generator, a SVM classifier was used. This classifier, after the training phase, determines the malfunction state according to the values of several measured variables. In this context, a brief analysis on the vibrations in induction electric motors was conducted, with the perspective of establishing an analogy with the electric generators of wind turbines.

In terms of optimization, a methodology to assist the decision for the sequence of visits to be made by maintenance technicians on the various wind generators is proposed. This methodology takes into account the on-condition maintenance plan previously defined. The proposed optimization method uses genetic algorithms and a specific solution to solve the sequence of visits problem. The organization of the maintenance management of wind farms is structured, integrating a set of developed hardware and software modules, as well as a set of updates and new modules, developed for the SMIT software. The proposed structure including the new modules, allows implementations based on corrective maintenance, planned maintenance and on-condition maintenance. A client/server maintenance management system was developed, using open-source software whenever possible. It includes the Linux operating system, the PostgreSQL database engine, and the development tools Octave, R, Apache and PHP. The SMIT client was programmed using Delphi and interacts with the user through the Windows platform.

In terms of hardware, the followed methodology relies on the use of low cost components and devices, to create a data acquisition system over IP networks. The basic idea consists on distributing a master clock to the different field equipments, to ensure the synchronous acquisition at the different data collection points. The SNTP and PTP protocols were used to implement a set of control techniques in order to achieve clock synchronization. The basic structure of the system uses data collecting devices connected through a CAN network. One of the devices, which has CAN and Ethernet connectivity, coveys the acquired information and relays it to the SMIT server. Simultaneously, this master node controls the data acquisition sequence, as well as the clock synchronization with the SMIT server. The integration of the developed hardware and software modules implies the flow of data from the acquisition nodes to the server, which sends time references to the master device, including the reference clock signal. The SMIT server, using algorithms based on Time Series, analyzes the acquired data using the Octave or R platforms, to predict possible failures or dysfunctional states. Based on these predictions, the server can anticipate the generation of the respective alerts, with the emission of the corresponding Working Orders.

**Keywords**: embedded systems, IP networks, maintenance, time series, wind generators

# Résumé

Cette thèse est développée sous la Condition basée Maintenance des Systèmes pour la Production de l'Énergie Renouvelable utilisant les réseaux IP, spécifiquement dans la Production Éolienne. Sont proposé un modèle de *software/hardware* et une architecture qui permettent la mise en oeuvre de solutions de planification de maintenance de condition, par la mesure à distance de plusieurs variables de contrôle. L'approche utilisée pour l'entretien prévu, dans la source prédictive, emploie de préférence des Séries Temporelles pour suivre l'évolution des variables de condition, comme soient la température, la pression, la viscosité et le module du spectre de fréquences. La connaissance de ces variables permet d'accompagner l'état de fonctionnement des équipements et de prévoir l'état suivant. Dans ce contexte, une contribution est accordée par un changement de l'algorithme de lissage exponentiel, afin de la rendre adaptative. Sa performance est accompagnée par la Série de *Mackey-Glass* et d'une Série spécifiquement développée conformément à l'attente de l'évolution des signes à prévoir. Pour déterminer, à niveau macroscopique, l'état de la condition de fonctionnement d'un aérogénérateur, a été utilisé le classificateur SVM. Ce classificateur, après la phase de formation, détermine, selon les mesures des plusieurs variables, l'instant plus probable de l'état de panne. Encore dans ce contexte a fait une brève analyse des vibrations dans les moteurs à induction, en raison de son analogie avec les générateurs électriques des aérogénérateurs.

En termes d'optimisation, se propose une méthodologie d'appui à la décision relatif à la séquence des visites à effectuer par les techniciens de manutention aux différents aérogénérateurs. Cette méthodologie tient compte des plans de maintenance selon les conditions définies à l'avance. La méthodologie d'optimisation proposée, utilise des algorithmes génétiques et une solution spécifique pour le problème. L'organisation de la gestion de la manutention des parcs éoliens est structurée de manière à intégrer *hardware* approprié/développé, ainsi qu'un ensemble de mises à jour et de nouveaux modules mis en oeuvre pour le *software* SMIT. Cette structure permet le fonctionnement dans les sources de la manutention non planifiée, planée systématique et conditionnelle. En ce que concernent le *software*, a été développé un système de gestion de la maintenance basée sur les communications client/serveur, en utilisant, si possible, *software open-source*. Le système développé comprend le système d'exploitation *Linux*, le moteur de base de données *PostgreSQL* et les outils de développement *Octave*, *R*, *Apache* et *PHP*. Le client SMIT est programmé en *Delphi* et il dialogue avec l'utilisateur à travers la plate-forme *Windows*.

Au niveau du *hardware*, la méthodologie utilise des composants et dispositifs de j'abaisse coût, pour créer un système d'acquisition de données sur réseaux IP. L'idée de base consiste à distribuer une horloge maître par les différents équipements de champ, pour garantir l'acquisition synchrone aux plusieurs points de collecte de données. À partir des protocoles SNTP et PTP, ont été mis en oeuvre plusieurs techniques de contrôle pour la synchronisation d'horloge. La structure de base du système utilise des noeuds de recueil des données inter reliés par un réseau *CAN*. L'un des noeuds, avec la connectivité *CAN* et *Ethernet*, concentre et envoie des informations rassemblées pour le serveur SMIT. En simultané, ce noeud contrôle la séquence d'acquisition de données, ainsi que la synchronisation de l'horloge avec le serveur. L'intégration globale du *hardware* et du *software* développé, suppose un flux de données de noeuds d'acquisition au serveur, qui envoie à son tour des références temporelles au noeud maître, y compris un signe d'horloge. Le serveur SMIT, grâce à des algorithmes basés sur les séries temporelles, analyse les données reçues en utilisant la plate-forme *R* ou *Octave* de prédire les défaillances possibles et les états dysfonctionnels. Basé sur cette prédiction, le serveur produit d'avance les respectives alertes avec l'émission des Ordres de Travail correspondant.

**Mots-clés**: aérogénérateurs, maintenance, réseaux IP, séries temporelles, systèmes embarqués

Este trabalho foi financiado pela FCT- Fundação para a Ciência e Tecnologia, através do contrato SFRH/BD/39225/2007 com participação de fundos do POPH/ESF (www.poph.qren.pt) e da EU (europa.eu).






# Agradecimentos

Desejo dar os meus agradecimentos:

- À FCT- Fundação para a Ciência e Tecnologia, que financiou este trabalho de doutoramento através do contrato SFRH/BD/39225/2007 com participação de fundos do POPH/ESF (www.poph.qren.pt) e da EU (europa.eu), e financiou ainda a impressão desta dissertação;




- À FEUP por ter dado apoio logístico;
- Ao ISEC e aos ex-Presidentes do DEE, Engº António Campos e Doutor Fernando Lopes, por terem acedido à implementação dos protótipos de *hardware*, disponibilizando os meios afectos ao GTE;
- Aos Professores Doutores Fernando Maciel Barbosa e José Torres Farinha por terem orientado este trabalho, sem os quais não teria sido realizado;
- Ao Engº José Carlos Matos do INEGI por ter sido interlocutor na obtenção dos dados referentes a um parque eólico, para a validação de resultados experimentais;
- Aos colegas que contribuíram com sugestões, em particular ao Fernando Lopes, Victor Santos, Horácio Fachada; ao colega Carlos Coelho pela ajuda valiosa na utilização do *software* PSCAD, ao colega Marco Silva por ter ajudado na soldadura de SMDs, ao colega Pedro Trovão por ter facultado o acesso ao material do Lab. de Accionamentos para o ensaio de vibrações com vários motores existentes e pela prestação de alguns esclarecimentos;
- À minha esposa e família por terem tido toda a paciência durante os longos instantes de tempo em que não me foi possível dar-lhes a devida atenção;
- Aos meus pais e ao meu irmão.

*"Quando o trabalho é prazer, a vida é uma grande alegria.*
*Quando o trabalho é dever, a vida é escravidão."*

Máximo Gorki, escritor russo (1868-1936)

*"O segredo da felicidade está na liberdade;*
*o segredo da liberdade está na coragem."*

Péricles (495 a.C.-429 a.C.)

*"O futuro constrói-se no presente."*

Autor desconhecido

# Conteúdo


**1 Introdução** **1**
- 1.1 Enquadramento . . . 2
- 1.2 A energia eólica . . . 3
- 1.3 Estrutura da Dissertação . . . 7

**2 Manutenção de geradores eólicos - Estado da arte** **9**
- 2.1 Estado da Arte . . . 9
  - 2.1.1 Manutenção de sistemas genéricos . . . 9
  - 2.1.2 Manutenção de sistemas eólicos . . . 11
  - 2.1.3 *Software* de gestão da manutenção . . . 14
- 2.2 Acções de Manutenção de parques eólicos . . . 16
- 2.3 Conclusões . . . 20

**3 As energias renováveis e os geradores eólicos** **21**
- 3.1 Motivação para o uso de energias renováveis . . . 21
- 3.2 Outros tipos de energias renováveis . . . 24
  - 3.2.1 Solar fotovoltaico . . . 24
  - 3.2.2 Solar térmico, ou solar térmico-eléctrico . . . 26
  - 3.2.3 Solar-Hidrogénio . . . 27
- 3.3 A Energia Eólica . . . 28
  - 3.3.1 Breve história da energia eólica . . . 29
  - 3.3.2 O vento e o seu comportamento . . . 31
  - 3.3.3 As várias tecnologias eólicas . . . 31
  - 3.3.4 Locais de Instalação . . . 37
  - 3.3.5 Componentes do Sistema - turbina de eixo horizontal mais comum . . . 39
  - 3.3.6 Aplicações e aproveitamento do sistema eólico . . . 43
  - 3.3.7 Estudo teórico do aero-gerador . . . 44
- 3.4 Turbinas submersas . . . 47
- 3.5 Potência eólica instalada no mundo e em Portugal . . . 48
- 3.6 Conclusões . . . 50

**4 Organização e gestão da manutenção de parques eólicos** **51**
- 4.1 Manutenção . . . 51
- 4.2 Sistema de manutenção integrado . . . 54
  - 4.2.1 O *software* de gestão da Manutenção - SMIT . . . 57
- 4.3 Codificação dos objectos de Manutenção . . . 60
- 4.4 Estratégia de planeamento das Ordens de Trabalho . . . 61
- 4.5 Conclusões . . . 64

**5 Manutenção preditiva de aero-geradores** **65**
5.1 A motivação . . . 65
5.2 Protocolos IP para sincronização temporal de relógios . . . 66
5.2.1 SNTP . . . 67
5.2.2 IEEE 1588 PTP . . . 69
5.2.3 Sincronismo em sistemas reais . . . 71
5.2.4 PLL na sincronização absoluta de relógios . . . 72
5.2.5 Contributo para o sincronismo dos relógios . . . 74
5.3 Sistema para manutenção preditiva . . . 76
5.3.1 Gestão do módulo *gateway Ethernet-CAN* . . . 77
5.3.2 Gestão do módulo *CAN* de aquisição . . . 80
5.4 Configuração do sistema na base de dados do SMIT . . . 80
5.4.1 Planos de Manutenção de condição . . . 81
5.5 Aquisição de dados . . . 83
5.6 Conclusões . . . 85

**6 Modelos de predição** **87**
6.1 Fiabilidade de parques eólicos . . . 87
6.2 Modelos para predição de avarias . . . 90
6.2.1 Classificação do estado da curva de potência com base em SVM . . . 91
6.2.2 Detecção de avarias nos geradores das turbinas eólicas . . . 94
6.2.3 Predição com séries temporais . . . 97
6.3 Métodos de predição baseados em séries temporais aplicados à manutenção . . . 100
6.4 Conclusões . . . 104

**7 Modelos de planeamento da manutenção** **107**
7.1 Algoritmo para planeamento da manutenção . . . 107
7.2 Optimização dos percursos entre torres eólicas . . . 109
7.2.1 Formulação da função de custo . . . 111
7.2.2 Inspiração no algoritmo das formigas . . . 114
7.2.3 Algoritmo de pesquisa baseado no comportamento das formigas . . . 116
7.2.4 Algoritmo de pesquisa baseado em algoritmos genéticos . . . 117
7.2.5 Algoritmo de pesquisa baseado em algoritmo específico . . . 120
7.2.6 Determinação de distâncias entre dois pontos terrestres . . . 121
7.3 Conclusões . . . 122

**8 Integração e validação dos modelos** **123**
8.1 Resultados práticos . . . 123
8.1.1 Sistema de aquisição . . . 123
8.1.2 Sistema de sincronização de relógios via IP . . . 125
8.1.3 Classificação . . . 131
8.1.4 Análise de Séries Temporais . . . 132
8.1.5 Análise da optimização de trajectos . . . 137
8.2 Conclusões . . . 141

**9 Conclusões e Trabalho Futuro** **143**
9.1 Conclusões . . . 143
9.2 Contributos . . . 144
9.3 Desenvolvimentos Futuros . . . 145

**Anexos** **147**

**A Sistema Integrado de Gestão de Manutenção** **147**
A.1 Introdução . . . 148
A.2 Diferentes níveis de acesso . . . 149
A.3 Módulo de Objectos de Manutenção . . . 150
A.4 Módulo de Planos de Manutenção . . . 153
A.5 Módulo de Ordens de Trabalho . . . 156
A.6 O Servidor SMIT . . . 161
A.6.1 Motor de base de dados . . . 162
A.6.2 Ferramentas para cálculo científico . . . 169
A.7 Instalação do *Software* SMIT . . . 172
A.7.1 Servidor SMIT para *FreeBSD* . . . 172
A.7.2 Servidor SMIT para *Slackware* . . . 173
A.7.3 SMIT Cliente para *Windows* . . . 173
A.7.4 Construção dos ficheiros de instalação para *Windows* . . . 174
A.8 Conclusões . . . 176

**B Descrição do *software*/*hardware* do sistema de Aquisição de Dados** **177**
B.1 Sistemas de aquisição de dados de baixo custo . . . 177
B.1.1 O *CAN* . . . 178
B.1.2 PIC18F2685 e DSPIC13F4012 . . . 179
B.1.3 ENC28J60 e PIC18F67J60 . . . 180
B.1.4 LM3S8962 . . . 180
B.1.5 Interligação entre o sistema de aquisição de dados e o anemómetro analógico de copos . . . 181
B.1.6 *Bootloader* para as *gateway Ethernet-CAN* . . . 182
B.1.7 Listagem Parcial do protocolo SMIT-*Gateway* . . . 183
B.2 Conclusões . . . 187

**C Predição através de séries temporais** **195**
C.1 Estudo de métodos para previsão com séries temporais . . . 195
C.1.1 Séries temporais . . . 195
C.1.2 Estado da Arte . . . 196
C.1.3 Séries temporais - métodos mais relevantes . . . 199
C.1.4 Principais Métodos de Análise Estudados . . . 209
C.2 Cooperação e integração de vários métodos . . . 219
C.3 Optimização na estimação de parâmetros das séries temporais . . . 220
C.3.1 Método de optimização de *Levenberg-Marquardt* . . . 221
C.3.2 Método dos mínimos quadrados . . . 221
C.4 Conclusões . . . 221

**D Estudo de avarias em motores de indução** **223**
D.1 Introdução . . . 223
D.2 Conclusões . . . 227

**Referências** **229**

# Lista de Figuras


1.1 Sistema para aproveitamento da energia das ondas, desenvolvido pela empresa *Aquamarine Power*, da Escócia. . . . 3
1.2 Cima: Turbina eólica suspensa em pleno mar. Baixo: turbina *off-shore* da *Principle Power* que a EDP pretende utilizar [1]. . . . 4
1.3 Ventos planetários e a sua interacção [2, 3]. . . . 5
1.4 Sistema instalado na Serra da Lousã para medida de regime de ventos, Julho de 2009. . . . 6

2.1 Exemplificação de um aero-gerador. . . . 17
2.2 Exemplo de alguns passos na construção de uma turbina eólica. Fotos: KV Venti. 19

3.1 Principais indicadores energéticos e poluentes. . . . 23
3.2 Nova célula com 42,8% de eficiência à luz ambiente será comercializada a partir de 2010. . . . 25
3.3 Sistema de aproveitamento solar para produção de energia eléctrica utilizando um motor de *Stirling* [4, 5]. . . . 27
3.4 Princípios de funcionamento do aproveitamento solar para produção de energia eléctrica. . . . 28
3.5 Esquerda, o ciclo proposto para as células de combustível, utilizando hidrogénio, o que origina uma energia limpa sem emissões. Direita, fotos do reactor criado pelos investigadores do projecto *Hydrosol* [6]. . . . 29
3.6 Mapas dos recursos eólicos em terra e nas zonas costeiras da Europa [7]. . . . 31
3.7 Diferentes tipos de geradores com indicação do limite de *Betz* 59,3%. . . . 32
3.8 Diferentes tipos de *rotors Savonius* [8, 9]. . . . 34
3.9 Exemplos do rotor de *Lenz* (esquerda e centro). Direita: Rotor *H-Darrieus* [10, 11, 12]. . . . 35
3.10 Turbinas de eixo horizontal com rotores de três pás [13]. . . . 36
3.11 Parque eólico marítimo de *Vinderby* na Dinamarca. . . . 37
3.12 Tecnologias que poderão equipar as cidades de forma a aproveitar os ventos citadinos, neste caso claramente bem integradas no meio envolvente. . . . 38
3.13 Tecnologia *motorwind*, que usa rotores de baixa potência agrupados para construir um sistema de grande dimensão. . . . 38
3.14 Principais elementos de uma Turbina eólica [14]. . . . 40
3.15 Relação de altura da torre e a potência da turbina eólica [14]. . . . 41
3.16 Aero-geradores de velocidade constante e variável [14]. . . . 42
3.17 Relação entre a Velocidade do vento ao longo de diferentes secções da turbina [14]. 45
3.18 Forças actuantes numa pá [14]. . . . 46
3.19 Construção de uma turbina submersa com rotor de duas pás [15]. . . . 48

3.20 Panorama em Agosto de 2009 (Fotos DGEG, Renováveis - Estatísticas Rápidas). 49

4.1 Sistema de Manutenção Integrado proposto. . . . . 54
4.2 Interligação do SMIT com vários sistemas, incluindo os comerciais. . . . . 57
4.3 SMIT: Cliente, para ambiente *Windows*. . . . . 58
4.4 SMIT: Relações entre os diferentes objectos de Manutenção. . . . . 59
4.5 Constituição dos códigos dos equipamentos. . . . . 60
4.6 Um plano de Manutenção inserido no SMIT. . . . . 62

5.1 Mensagens SNTP enviadas entre mestre e escravo. . . . . 68
5.2 Mensagens PTP enviadas entre mestre e escravo. . . . . 70
5.3 PLL analógica para descodificação de FM. . . . . 71
5.4 PLL digital para descodificação de televisão digital. . . . . 72
5.5 Diagrama temporal considerando um desvio de frequência no nó escravo. . . . . 73
5.6 Diagrama da actualização de relógio. . . . . 74
5.7 Modificação ao PTP para cálculo do $\Delta Erro$. . . . . 75
5.8 Sistema de suporte à manutenção preditiva. . . . . 77
5.9 Esquema de mensagens para medir o tempo de propagação na rede *CAN*. . . . . 77
5.10 Algoritmo executado no nó mestre, modo configuração. . . . . 78
5.11 Algoritmo executado no nó mestre, aquisição. . . . . 79
5.12 Módulo de configuração da condição de funcionamento dos diferentes objectos de manutenção. Indicação de rotinas matemáticas de tratamento de dados. . . . . 81
5.13 Módulo de configuração da condição de funcionamento dos diferentes objectos de manutenção. Parâmetros de aquisição. . . . . 82
5.14 Módulo de configuração da condição de funcionamento dos diferentes objectos de manutenção. Configuração de um módulo de E/S *CAN*. . . . . 83
5.15 Módulo de configuração da condição de funcionamento: visualização da recepção de dados. . . . . 84
5.16 Esquerda: anemómetro de ultra-sons, direita: anemómetro analógico de copos. . 85
5.17 Diagrama do fluxo de informação no sistema global. . . . . 86

6.1 As avarias típicas e a sua distribuição pelos diferentes equipamentos de uma turbina eólica, dados referente à Tab. 6.2 [16, 17]. . . . . 88
6.2 Distribuição de falhas de um grupo de turbinas eólicas e as respectivas consequências, em termos dos custos indirectos; dados referente à Tab. 6.2 [16, 17]. . . . . 89
6.3 Percentagem de falhas em função da potência da turbina [18]. . . . . 89
6.4 Em $\Re^2$ linear, o SVM calcula a fronteira que minimiza a distância aos pontos que definem a fronteira. . . . . 92
6.5 As funções de *kernel* efectuam um mapeamento de forma a linearizar o espaço. . 93
6.6 Faixa de treino considerada como em operação adequada da curva de potência do aero-gerador. . . . . 94
6.7 Relação entre a série $z[t]$ com amostragem periódica e a série aperiódica $y[k]$. . . 99
6.8 Interpolação parabólica para aumentar a fiabilidade da estimação do tempo para avaria. . . . . 103

7.1 Exemplo gráfico de um problema da optimização de percursos. O exemplo é apenas para fins de simulação. . . . . 111
7.2 Funcionamento do algoritmo de pesquisa da Formiga (adaptado de [19]). . . . . 114
7.3 Operadores genéticos utilizados. . . . . 118

7.4 Modelo esférico do globo terrestre [20]. . . . . 121

8.1 Medida do tempo de atraso na rede *CAN* (ns). A tracejado - valor teórico; a cheio - valor medido, para mensagens de 1 a 8 *bytes*. . . . . 124
8.2 Medida do tempo para aquisição (ns), depois de ter programado o *CAN* para enviar uma mensagem com 1 *byte* de dados. . . . . 125
8.3 Experiência nº 1. Valores calculados e recolhidos no momento da recepção das mensagens *Follow UP* e *Delay Response*. . . . . 126
8.4 Experiência nº 2: Precisão de zero casas decimais no parâmetro CNT_TICK_MAX. 127
8.5 Experiência nº 3: Precisão de duas casas decimais no parâmetro CNT_TICK_MAX. Apenas envia uma mensagem *Delay Request*. . . . . 127
8.6 Experiência nº 4. Novo método com condições análogas à experiência nº1 (Fig. 8.3). . . . . 129
8.7 Experiência nº 5. Resultados com o sistema real. . . . . 130
8.8 Classificação SVM dos dados de Parque Eólico, de acordo com a Eq. 6.6 da pág. 93. Vermelho: avaria; verde: bom funcionamento. . . . . 133
8.9 Classificação SVM dos dados de Parque Eólico, de acordo com a Eq. 6.7 da pág. 93. Vermelho: avaria; verde: bom funcionamento. . . . . 134
8.10 Estudo comparativo dos vários métodos apresentados no anexo C. . . . . 135
8.11 Esquerda: ESMSE(0.5,0), direita: ESMSE(0.5,5). Variação de $\alpha$ para a previsão de $\hat{y}[k] \approx y[k+1]$ da série da Fig. 8.10 cujo desempenho se encontra na Tab. 8.1. 136
8.12 Resultados dos algoritmos de séries temporais aplicados à série *Mackey-Glass*: previsão de $\hat{y}[k] \approx y[k+1]$. . . . . 137
8.13 À esquerda: ESMSE(0.5,0), direita: ESMSE(0.5,5). Variação de $\alpha$ para a previsão de $\hat{y}[k+2] \approx y[k+3]$ referente à Tab. 8.2. . . . . 138

A.1 Planos de Manutenção e OT - Diagrama de interligação dos diferentes módulos. . 148
A.2 Edição da mesma ficha por dois utilizadores. À Esquerda: edição da ficha. À direita: apenas acesso de leitura. . . . . 149
A.3 Em cima: Lista de OM. Em baixo: caracterização de um OM. . . . . 151
A.4 Caracterização Funcional e Técnica. . . . . 152
A.5 Em cima: totais das intervenções planeadas e não planeadas; Em baixo: listagem das peças-de-reserva do OM. . . . . 152
A.6 Em cima: editor de relações entre OMs. Em baixo: árvore de relações do OM. . . 153
A.7 Selecção do plano de manutenção. As cores podem ser alteradas. . . . . 154
A.8 Ficha de um plano de manutenção e escolha do algoritmo de planeamento. . . . . 156
A.9 Recursos do plano e respectivos custos. . . . . 157
A.10 Em cima: custos previsíveis. Em baixo: indicação das acções a efectuar. . . . . 157
A.11 Inserção dos Clientes de OM para imputação dos custos das intervenções. . . . . 158
A.12 Selecção de OT. . . . . 158
A.13 Em cima: alguns campos da Ficha de OT. Ao meio: mão-de-obra da OT. Em Baixo: Peças-de-Reserva e ferramentas da OT. . . . . 160
A.14 Caracterização de uma ordem de trabalhos correctiva (não planeada), onde é necessário "catalogar" a avaria através da recolha das causas, do procedimenteo de reparação e de palavras chave, bem como da possibilidade de ocorrência. . . . . 161

B.1 Formatos da mensagem de dados. Em cima: *Standard*. Em baixo: *Extended*. . . . 178
B.2 Indicação ao nível físico do local onde o controlador *CAN* deve amostrar um *bit* numa mensagem. . . . . 179

B.3 Esquema para interligação com anemómetro analógico de copos. . . . . 181
B.4 Programa em *Windows* para "carregar" o *firmware*. . . . . 182
B.5 Imagem de alguns dos protótipos de teste. . . . . 188
B.6 Esquema da placa PIC18F2685. . . . . 189
B.7 Esquema da placa dsPIC30F4012. . . . . 190
B.8 Esquema da placa ligação *Ethernet* com ENC28J60. . . . . 191
B.9 Esquema da placa ligação *Ethernet* com PIC18F67J60. . . . . 192
B.10 Esquema da placa ligação *Ethernet* com Arm-Cortex LM3S8962. . . . . 193

C.1 À esquerda uma rede *feedforward*. À direita uma rede recorrente com realimentação de estado. . . . . 205
C.2 Exemplo de um neurónio na camada $j$, elemento de processamento básico de uma rede neuronal. . . . . 206
C.3 Modelo neuronal de uma rede *feedforward* adaptada em conjunto com o método ARMA, dando origem ao método denominado de NARMA. . . . . 207
C.4 Exemplificação do operador de *crossover* aplicado aos indivíduos apresentados na Tab. C.2, cuja resultados foram dois novos indivíduos. . . . . 208
C.5 Relação entre duas séries, uma aperiódica e outra periódica para dedução da fórmula do alisamento exponencial aperiódico. . . . . 215
C.6 Rede *feedforward* de 3 camadas usada para prever $m+1$ valores no futuro. . . . . 217
C.7 Exemplificação gráfica do método dos mínimos quadrados, aplicado neste caso a uma recta. O método minimiza a distância em X, ou Y, conforme o caso. . . . . 222

D.1 Aquisição e tratamento. . . . . 224
D.2 FFT do sinal de vibração de dois motores em função da carga aplicada. . . . . 225
D.3 Exemplo do simulador espectral. . . . . 226

# Lista de Tabelas


2.1 Exemplo de algumas aplicações da área da gestão da manutenção. . . . . 15

3.1 Classificação das turbinas eólicas em termos da potência produzida. . . . . 33
3.2 Classificação das turbinas eólicas em termos do diâmetro e área varrida pelo rotor. 33

4.1 Exemplo de codificação. . . . . 61
4.2 Exemplos de planos inseridos no SMIT, um dos quais respeitante ao apresentado na Fig. 4.6 . . . . . 63

6.1 Dados de Fiabilidade de aero-geradores, MTBF de aero-geradores instalados na Alemanha e Dinamarca [16, 17]. . . . . 90
6.2 MTTR de um grupo de aero-geradores Alemães (MTBF apresentado na Tab. 6.1). 91
6.3 Avarias típicas num motor de indução [21]. . . . . 96

7.1 Formato genético de um indivíduo. . . . . 118

8.1 Resultados da previsão para a série da Fig. 8.10. À esquerda: resultados globais, à direita: resultados do intervalo 800-1000. . . . . 136
8.2 Resultados da previsão para a série *Mackey-Glass*. À esquerda: previsão para $\hat{y}[k] \approx y[k+1]$; à direita: $\hat{y}[k+2] \approx y[k+3]$. . . . . 137
8.3 Resultados da previsão para a série *Mackey-Glass*. À esquerda: previsão para $\hat{y}[k+4] \approx y[k+5]$, à direita: $\hat{y}[k+6] \approx y[k+7]$. . . . . 138
8.4 Dados iniciais do problema de optimização. . . . . 139
8.5 Dados: distância em quilómetros entre cada Nó (Ligação directa). . . . . 139
8.6 Dados: montante pago por quilómetro entre cada Nó (Ligação directa). . . . . 139
8.7 Dados: número de Dias de viagem (Ligação directa). . . . . 140
8.8 Dados: montante pago por cada dia de viagem (Ligação directa). . . . . 140
8.9 Solução encontrada pelo algoritmo genético secundário, para o melhor caminho entre um nó origem e um nó destino. Tempos de execução em segundos para os três algoritmos. . . . . 140
8.10 Soluções encontradas para várias simulações, com diferentes parâmetros para: número de gerações, número de indivíduos e inclusão de uma solução inicial pseudo-óptima. . . . . 141

C.1 Funções de activação: função linear; função sigmóide; função tangente hiperbólica; função degrau e função ERF. . . . . 206
C.2 Exemplo de três codificações possíveis dos indivíduos num algoritmo de previsão de séries temporais através de algoritmos genéticos. . . . . 208

# Abreviaturas e Símbolos

| | |
|---|---|
| ACO | *Ant Colony Optimization Theory* |
| ADENE | Agência para a Energia |
| APMI | Associação Portuguesa de Manutenção Industrial |
| AR | *Auto Regressive* |
| ARCH | *Autoregressive Conditional Heteroskedasticity* |
| ARIMA | *Auto Regressive Integrated Moving Average* |
| ARM | *Advanced RISC Machine* |
| ARMA | *Autoregressive Moving Average* |
| ARMAX | *Autoregressive Moving Average Exogenous* |
| ARX | *Autoregressive Exogenous* |
| API | *Application Programming Interface* |
| BMC | *Best Master Clock Algorithm* |
| CAN | *Controller Area Network* |
| CMMS | *Computerized Maintenance Management System* |
| CORBA | *Common Object Request Broker Architecture* |
| CORRE | *Correlation Pearson Factor* |
| CSMA/CD | *Carrier Sense Multiple Access With Collision Detection* |
| DFT | *Discrete Fourier Transform* |
| DHCP | *Dynamic Host Configuration Protocol* |
| DGEG | Direcção Geral de Energia e Geologia |
| DPLL | *Digital Phase Locked Loop* |
| DVB-T | *Digital Video Broadcasting - Terrestrial* |
| ERF | *Extended Ryderberg Function* |
| EU | *European Union* |
| FFT | *Fast Fourier Transform* |
| FIFO | *First In First Out* |
| FM | *Frequency Modulation* |
| FPGA | *Field Programmable Gate Array* |
| GARCH | *Generalized Autoregressive Conditional Heteroskedasticity* |
| GMT | *Greenwich Mean Time* |
| GPS | *Global Positioning System* |
| GRS | *Geodetic Reference System* |
| GTE | GTE - Gabinete Técnico de Electrotecnia |
| IEPE | *Integrated Electronics Piezo Electric* |
| INEGI | Instituto de Engenharia Mecânica e Gestão Industrial |
| INETI | Instituto Nacional de Engenharia e Tecnologia Industrial |
| IP | *Internet Protocol* |
| ISO | *International Organization for Standardization* |
| JET | *Joint European Torus* |

| | |
|---|---|
| LCO | *Local Clock Offset* |
| LPG | *Linear Genetic Programming* |
| LRN | *Layer Recurrent Network* |
| MA | *Moving Average* |
| MAE | *Mean Absolute Error* |
| MAPE | *Mean Absolute Percentage Error* |
| MERGE | *Mobile Energy Resources for Grids of Electricity* |
| MLP | *Multilayer Perceptron* |
| MSE | *Mean Square Error* |
| NARIMA | *NonLinear Autoregressive Integrated Moving Average* |
| NARMA | *Nonlinear Autoregressive and Moving Average Network* |
| NARX | *Nonlinear AutoRegressive Network with exogenous inputs* |
| NTP | *Network Time Protocol* |
| Octave | *Gnu Octave - language for numerical computation* |
| PC | *Personal Computer* |
| OSI | *Open System Interconnection* |
| PCB | *Printed Circuit Board* |
| PCR | *Program Clock Reference* |
| PG | Programação Genética |
| PIC | *Micro-controladores da Família Microchip* |
| PLC | Autómatos Programáveis (*Program Logic Controller*) |
| PLL | *Phase Locked Loop* |
| PMEs | Pequenas e Médias Empresas |
| PTP | *Precision Time Protocol* |
| R | *R Project for Statistical Computation* |
| RAD | *Rapid Application Development* |
| RBF | *Radial Basis Function* |
| RMSE | *Root Mean Square Error* |
| RS232 | Uma norma para troca série de dados binários |
| RS485 | Uma norma para troca série de dados binários em modo diferencial |
| RTC | *Real Time Clock* |
| RTD | *Round-Trip Delay* |
| SARIMA | *Seasonal Autoregressive Integrated and Moving Average* |
| SCR | *System Clock Reference* |
| SDCC | *Small Device C Compiler* |
| SNMP | *Simple Network Management Protocol* |
| SNTP | *Simple Network Time Protocol* |
| SMIT | Sistema Modular Integrado de Terologia |
| SPLL | *Software Phase Locked Loop* |
| SVM | *Support Vector Machine* |
| SVR | *Support Vector Regression* |
| SynUTC | *Synchronized Universal Time Coordinated* |
| TCMA | Taxa de Crescimento Médio Anual entre 2001 e 2009 |
| TCP | *Transport Control Protocol* |
| TDFFN | *Time Delay FeedForward Network* |
| TDNN | *Time Delay Neural Network* |
| TDT | *Televisão Digital Terrestre* |
| TI | Transformador de Intensidade |
| TIC | *Theil Inequality Coefficient* |

| | |
|---|---|
| TLFN | *Time Lagged FeedForward Network* |
| TQ | *Time Quantum* |
| TTL | *Transistor-Transistor Logic* |
| UDP | *User Datagram Protocol* |
| UTC | *Universal Time (Coordinated)* |
| VCO | *Voltage Controller Oscillator* |
| WGRS | *World Geodetic Reference System* |
| WGS | *World Geodetic System* |
| WWW | *World Wide Web* |

**Símbolos e Abreviaturas do *Software* SMIT ou da área da manutenção**

| | |
|---|---|
| MTBF | *Mean Time Between Failures* |
| MTTR | *Mean time to Repair* |
| MTTS | *Mean Time To be Stopped* |
| OM | Objectos de Manutenção |
| OT | Ordens de Trabalho |
| PM | Planos de Manutenção |

# Capítulo 1

# Introdução

A presente dissertação insere-se no âmbito da manutenção condicionada de aero-geradores eólicos de grande porte enquadrados em parques eólicos.

A manutenção de uma torre eólica só por si envolve problemas que se relacionam, por um lado, com a sua localização devido a, normalmente, se encontrarem posicionadas em locais de difícil acesso e distantes das sedes dos respectivos departamentos de manutenção e, por outro lado, com o acesso aos equipamentos de controlo e ao aero-gerador propriamente dito, devido à altura das torres e às dificuldades inerentes à falta de espaço para aceder aos respectivos órgãos e componentes.

Face a este cenário, a presente dissertação versa sobre a investigação e desenvolvimentos efectuados em torno desta temática tendo levado à implementação de protótipos de *hardware*, de *software* e respectivos algoritmos que, por sua vez, interagem com um sistema de informação para manutenção que controla e planeia as respectivas ordens de trabalho.

No que concerne ao *hardware* foram desenvolvidas placas de aquisição de dados, tais como informação de vibrações e correntes, que são variáveis indispensáveis ao planeamento da manutenção de condição das torres eólicas. Para além da aquisição, estes dados terão que ser transmitidos para o sistema de informação, SMIT (Sistema Modular Integrado de Terologia), o que foi conseguido através da implementação de um *interface* IP para que pudessem ser utilizadas as redes GSM, TCP/IP comerciais. O *hardware* desenvolvido tem embebido o necessário *software* de gestão e controlo de fluxo dos respectivos dados, relativo aos sinais adquiridos e a enviar, que se caracterizam por possuir uma característica transversal a todas as ferramentas utilizadas que é a de serem *open-source*. Nesta mesma perspectiva, todo o *hardware* desenvolvido possui uma característica comum que é o seu baixo custo ($\leq 40$ €), com o objectivo de, se tal for da conveniência de eventuais interessados no futuro, poderem replicar os dispositivos para que possam ser instalados em todas as torres dos parques eólicos ao menor custo possível.

Depois de implementada a solução da aquisição e transmissão dos dados referentes às variáveis de condição, impõe-se proceder à sua recepção e tratamento, para que o SMIT possa fazer a

predição e emissão das Ordens de Trabalho (OT) planeadas. Para o efeito, foi desenvolvido um módulo de manutenção condicionada que contém embebido modelos de previsão baseados em séries temporais e algoritmos genéticos que provaram ser adequados para os dados em apreço.

As intervenções de manutenção requerem recursos humanos e materiais que são planeados nas OTs bem como requeridos nas OTs não planeadas. Ora, no caso em análise, cada deslocação a uma torre eólica implica uma logística complexa e onerosa, pelo que o planeamento e optimização das visitas de manutenção deve ser prevista de forma racional. Nesta perspectiva, foi desenvolvido um algoritmo para planeamento das intervenções de manutenção nos parques eólicos tendo como objectivo optimizar as respectivas deslocações.

A presente dissertação apresenta assim um conjunto de *peças* de um *puzzle* que constitui o modelo de manutenção condicionada de parques eólicos que, em síntese, é constituída pelos seguintes elementos:

- *Hardware* e *software* para aquisição e transmissão de dados através de redes IP;
- Módulo de predição do SMIT;
- Módulo de optimização da logística das intervenções;
- Outros componentes para interligação dos precedentes.

## 1.1 Enquadramento

De acordo com [22, 23] as alterações climáticas têm como uma das suas importantes causas a queima de combustíveis fósseis, levando a que o nosso planeta corra sérios riscos de mudanças ao nível das condições de habitabilidade. Outro factor relevante é o preço do petróleo e a sua escassez para as futuras gerações. Salienta-se, no entanto, o facto de que uma vida sustentável, ao ritmo de crescimento do último século, só será possível através da resolução da problemática da energia, através da obtenção de uma energia mais "limpa".

No domínio da investigação, algumas hipóteses apresentam-se credíveis, como por exemplo, a energia nuclear de fusão. A Inglaterra tem uma aposta clara neste domínio com um centro de investigação no seu território [24].

A perspectiva desta dissertação centra-se nas energias renováveis pelo que, outro domínio importante que justifica referir neste capítulo introdutório é o da energia solar, quer para aquecimento de águas sanitárias - solar térmico - quer para a geração da energia eléctrica - painéis fotovoltaicos. No caso dos painéis fotovoltaicos, recentes descobertas relançaram um interesse redobrado, através do incremento da eficiência para 50% [25].

Ainda na área do aproveitamento solar, os sistemas de conversão de energia térmica em energia mecânica e, posteriormente, em energia eléctrica, assumem grande relevância através da recuperação da tecnologia dos motores de *Stirling* [26].

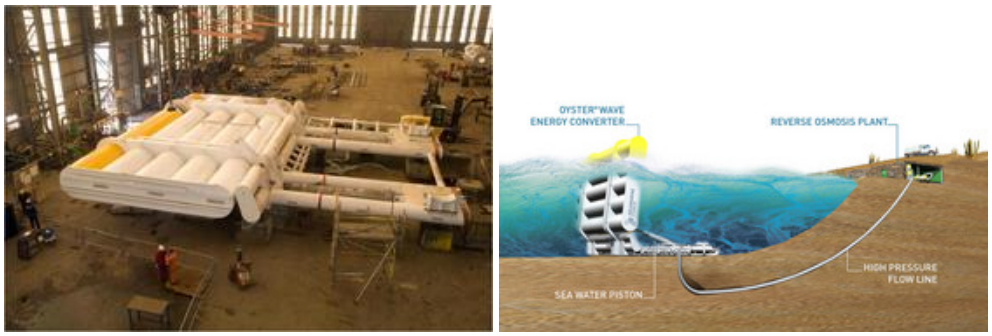



Figura 1.1: Sistema para aproveitamento da energia das ondas, desenvolvido pela empresa *Aquamarine Power*, da Escócia.

Com largo potencial de implementação em Portugal, a geração de energia através das ondas do mar é também uma das energias renováveis onde se desenvolvem enormes esforços de investigação. Algumas Instituições Portuguesas desenvolvem trabalho nesta área e, ao longo destes anos, diferentes técnicas têm sido investigadas com maior ênfase na costa dos Açores. Um dos protótipos mais recentes pode ser visualizado na Fig. 1.1 desenvolvido pela empresa *Aquamarine Power*, da Escócia [27].

A presente dissertação centra-se na energia eólica a qual tem tido uma expansão relevante em diversos países europeus assim como noutros países do mundo. A aposta clara da UE na produção de energia renovável eólica alavanca claramente o investimento nesta área. Portugal não tem deixado passar o desafio e, assume-se, à data, como um dos países já com uma larga implantação no uso dos seus recursos eólicos. Com a evolução tecnológica prevê-se ainda que esse aproveitamento se alargue a instalações marítimas dado que a tecnologia actual já o permite, através de turbinas suspensas em pleno mar. A Fig. 1.2 mostra um exemplo da primeira turbina instalada pela empresa *StatoilHydro* em parceria com a *Siemens* [28].

Com este enquadramento, face às inúmeras turbinas eólicas instaladas em Portugal, é da maior pertinência o desenvolvimento de novas técnicas e metodologias de manutenção condicionada de aero-geradores eólicos enquanto contributo muito significativo para a optimização do rendimento deste tipo de recurso.

## 1.2 A energia eólica

A energia eólica (eléctrica) advém da conversão da energia do vento em energia eléctrica [14].

O vento - massa de ar em movimento - resulta do movimento de rotação da Terra e da radiação solar que aquece de forma diferenciada a atmosfera. A movimentação de massas de ar é um processo permanente de interacção sol-planeta. Por este motivo, a relevância desta energia é elevada devido à durabilidade e ao facto de ser renovável.

Em termos práticos, existe circulação de ar entre zonas com diferentes pressões atmosféricas e, naturalmente, o ar circula das zonas de mais elevada pressão para as de mais baixa sendo, no



Figura 1.2: Cima: Turbina eólica suspensa em pleno mar. Baixo: turbina *off-shore* da *Principle Power* que a EDP pretende utilizar [1].

entanto, desviado pela força de *Coriolis* resultante do movimento de rotação do planeta.

As zonas de altas pressões formam-se quando o ar frio desce lentamente e comprime o ar que está por baixo. O ar pode ainda arrefecer em contacto com a superfície fresca da terra ou do oceano. Ao arrefecer comprime-se, libertando volume que será ocupado por outra massa de ar, aumentando assim a pressão. As baixas pressões resultam do aquecimento do ar que sobe, tornando mais rarefeito a massa de ar. Outra das razões para a existência de correntes de ar ao nível do planeta prende-se com o facto da radiação solar ser mais intensa nas zonas equatoriais relativamente às zonas polares. Tal facto advém destas zonas tropicais receberem os raios quase na perpendicular sendo, portanto, mais aquecidas que as zonas polares. O ar quente nas zonas tropicais de baixa altitude tende a subir e é substituído pelo ar mais frio que se desloca das regiões polares. Esta deslocação dá origem aos chamados ventos planetários. A Fig. 1.3 representa as grandes correntes de ar a nível do planeta devido aos aspectos referidos.

Os ventos planetários [2, 3] podem ser classificados em:

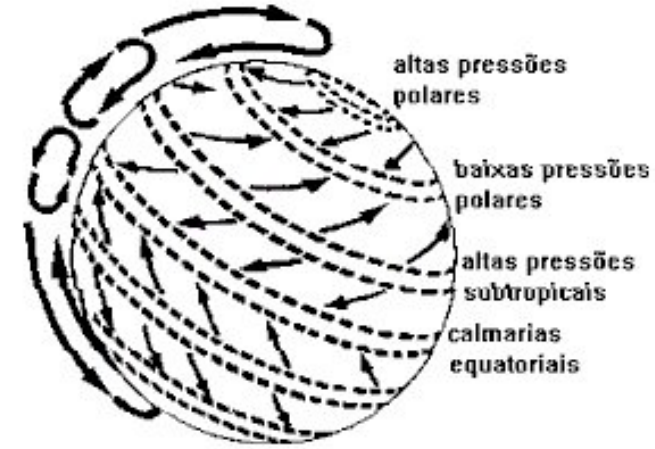



Figura 1.3: Ventos planetários e a sua interacção [2, 3].

- **Ventos alísios** - ventos de baixa altitude que circulam dos trópicos para o equador;
- **Ventos contra-alísios** - ventos de alta altitude que circulam do equador para os pólos;
- **Ventos do oeste** - ventos que circulam dos trópicos para os pólos;
- **Ventos polares** - ventos frios que circulam dos pólos para as zonas temperadas.

Sabendo que a terra está inclinada cerca de $23{,}5^{o}$ em relação ao plano da sua órbita em redor do sol, este facto traduz-se em variações sazonais na distribuição da radiação solar recebida tendo, como consequência, variações sazonais na duração e intensidade do vento. Por consequência, têm-se ventos sazonais periódicos ou continentais, designadamente moções ou brisas.

Os ventos periódicos que mudam de direcção numa base semestral são denominados de moções. Neste tipo de ventos, a cada estação do ano, o vento muda de direcção (normalmente em sentido contrário).

Em função da capacidade reflectora, absorvedora e de emissão do calor do sol, típica de cada superfície (mar, continente, etc), aparecem as brisas (ventos periódicos) que, tipicamente, circulam do mar para o continente e vice-versa. Especificamente, durante o dia, devido à terra aquecer mais rapidamente do que o mar (reflecte mais raios solares), o ar circula do mar para terra, a chamada brisa marítima. À noite, devido à terra arrefecer mais rapidamente, o ar circula na direcção oposta, a chamada brisa terrestre. Como o diferencial de temperatura nocturno entre o mar e a terra é menor, a brisa terrestre tende a ser de menor intensidade que a brisa marítima.

Em termos locais acontece um processo idêntico, sobreposto ao descrito anteriormente, os ventos locais. Normalmente, devido ao relevo da superfície terrestre, a sua forma mais expressiva acontece nos vales e montanhas: De dia, o ar quente das encostas das montanhas sobe e o ar frio desce sobre o vale para tomar o lugar do ar que subiu; De noite, a direcção do vento muda de sentido.



Figura 1.4: Sistema instalado na Serra da Lousã para medida de regime de ventos, Julho de 2009.

Genericamente, a movimentação de massas de ar processa-se em regime de turbulência. Matematicamente, a velocidade do vento é descrita por um valor médio adicionado de um desvio (flutuação). Assim, o vento é descrito por $V = v + v'$ onde $v$ é a velocidade média e $v'$ a flutuação. Outro aspecto importante é a direcção do vento que pode, e deve, ser caracterizada. Locais com ventos cuja direcção muda muito frequentemente são sinónimos de rajadas de vento. A direcção do vento predominante é fundamental para prevenir a influência entre aero-geradores num parque eólico - denominada de interferência de esteira.

Para o aproveitamento da energia do vento através dos geradores eólicos, é necessário, em muitos casos, fazer o estudo prévio, durante algum tempo, do regime de ventos, determinando assim os diferentes regimes de variação temporal do vento, nomeadamente, variações anuais, sazonais, diárias e de curta duração. Assim, para a instalação de um parque eólico é necessário levar em conta o estudo para os diferentes regimes de vento e das suas variações, concretamente:

- **Variações anuais** - é necessário adquirir dados de vários anos;

- **Variações sazonais** - dado que a potência disponível, à altura do aero-gerador, depende da velocidade do vento de forma cúbica, é importante determinar o seu comportamento em termos sazonais;

- **Variações diárias** - as variações diárias são importantíssimas, pois ao comparar-se a variação da velocidade ao longo de um dia, constata-se uma variação significativa de mês para mês. Por exemplo, escolher os locais onde o vento diurno é superior ao nocturno, dado que o consumo é superior nesse período;

- **Variações de curta-duração** - não são normalmente consideradas, desde que não sejam significativas. No entanto, oriundas das rajadas podem causar danos estruturais às turbinas eólicas e causar fadiga nos materiais. Devem ser estudadas com algum cuidado se apresentarem variações de velocidade significativas, seguidas de variações da direcção do vento.

Em resumo, é extremamente importante fazer uma análise concreta do regime de ventos nos locais onde se pretende instalar um parque eólico. Normalmente, no local onde se pretende colocar uma turbina é instalado temporariamente um sistema de medição do regime de ventos (Fig. 1.4), no mínimo durante um ano. Outro aspecto importante, é o facto da superfície terrestre influenciar a velocidade e direcção do vento, baixando essa influência à medida que a altitude aumenta. No caso do aproveitamento marítimo (*off-shore*), dado que existem menos obstáculos, há menos perturbações na direcção do vento devido ao tipo de superfície.

Um aspecto relevante na classificação da energia eólica como amiga do ambiente, passa pela caracterização do seu efeito sobre algumas espécies animais. Relativamente às aves existem por exemplo, notícias na imprensa[1] alegando que diversos aero-geradores aniquilaram morcegos e outras espécies de aves. No caso da instalação marítima, as escavações dos alicerces podem originar poeiras que são disseminadas ao longo de vários quilómetros através das correntes marítimas, tendo também um efeito nefasto na fauna e na flora aquática. Concluí-se muito superficialmente que, dependendo da forma como se constrói o aero-gerador e de como se projecta o controlo do seu funcionamento, este poderá ou não ser considerado como amigo do ambiente. Em alguns países mais avançados neste tipo de estudos, o aero-gerador é equipado com radares para detectar bandos migratórios, ficando inactivo quando se detectam os seus movimentos para evitar a morte de aves. Este problema tem tendência a ampliar-se, dado que o diâmetro do rotor tem vindo a aumentar e, portanto, a área de colisão é hoje superior relativamente há alguns anos atrás.

## 1.3 Estrutura da Dissertação

Para além da introdução exposta neste capítulo, esta dissertação contém mais oito capítulos e quatro anexos.

No capítulo 2, é descrito o estado da arte sobre manutenção de aero-geradores e manutenção de sistemas genéricos com algum grau de relação com a manutenção de aero-geradores. São ainda apresentados alguns dos programas informáticos de gestão da manutenção disponíveis e analisadas as suas principais características. Por último, são descritas as principais acções de manutenção em parques eólicos, nomeadamente a nível de inspecção e acções periódicas.

No capítulo 3 são explanadas as principais motivações para o uso de energias renováveis, e é efectuada uma apresentação dos diferentes tipos de energia, com ênfase na energia eólica. Expõem-se as principais características desta tecnologia do ponto de vista do passado/presente e os aspectos teóricos mais relevantes. Por último, apresenta-se sucintamente o panorama Português a nível das estruturas associadas à energia eólica.

[1] http://www.wind-watch.org/news/2007/05/30/study-shows-hundreds-of-dead-birds-bats-at-wind-turbines/

No capítulo 4 apresenta-se o modelo de organização da gestão da manutenção dos parques eólicos, tendo em conta o *software* de manutenção desenvolvido - SMIT. Expõe-se ainda a proposta para um sistema de manutenção integrado e, de seguida, uma abordagem de como utilizar o *software* para gerir OTs (Ordens de Trabalho) e PMs (Planos de Manutenção), com exemplos específicos para cada um dos casos.

No capítulo 5 descreve-se a estrutura necessária a nível físico para efectuar manutenção preditiva, concretizando com a apresentação da arquitectura do sistema de aquisição através de redes IP. São discutidos os protocolos de sincronização de relógio pela rede IP e o seu enquadramento no sistema global. Por último, é apresentada a integração entre os módulos de aquisição e o sistema central de gestão da manutenção - SMIT.

No capítulo 6 expõe-se uma perspectiva sobre a fiabilidade de alguns exemplos de aero-geradores, oriundos da Alemanha e Dinamarca. De seguida apresentam-se os modelos de predição de ruptura com ênfase para: a classificação por SVM, a análise de vibrações e a predição utilizando séries temporais.

No capítulo 7 exemplifica-se o método de previsão associado à manutenção planeada de equipamentos. Apresenta-se também uma metodologia para optimização da sequência das visitas dos técnicos de manutenção aos diferentes aero-geradores, sabendo qual será a previsão de avaria desses mesmos equipamentos. A metodologia de optimização utiliza algoritmos genéticos e uma solução específica para resolver o problema.

No capítulo 8 são apresentados os resultados dos diferentes testes desenvolvidos ao longo do trabalho descrito nesta dissertação.

No capítulo 9 são tecidas as principais conclusões em função do trabalho desenvolvido e dos resultados apresentados no capítulo 8, bem como perspectivadas as principais direcções para futuros desenvolvimentos.

No anexo A é apresentado o *software* SMIT sob as vertentes de OM (Objectos de Manutenção), OT, PM e, questões técnicas como instalação, programação e restantes factores associados.

No anexo B é apresentado o *hardware*, através de uma breve introdução à rede *CAN*. De seguida, apresentam-se os diferentes micro-controladores utilizados, bem como os desenvolvimentos associados ao demonstrador da arquitectura proposta. Por último, são apresentados os esquemas eléctricos das placas desenvolvidas.

No anexo C é detalhada toda a análise de séries temporais utilizada ao longo da presente dissertação. Vários métodos documentados na literatura são expostos, sendo apresentadas as modificações introduzidas com obtenção de bons resultados. O anexo apresenta também os algoritmos utilizados nos diversos métodos expostos, cuja análise comparativa é descrita no capítulo 8.

No anexo D são descritas as experiências de aquisição de sinais de vibração em motores de indução trifásicos e respectiva análise espectral. É proposta uma metodologia para, em conjunto com o uso de séries temporais determinar possíveis avarias. Embora constitua um tema sobejamente investigado na literatura, este estudo deveu-se à impossibilidade de o realizar num gerador eléctrico de um aero-gerador. É, portanto, apresentado apenas como indicativo, pelas semelhanças possíveis, entre a análise de vibração em motores de indução e em geradores eléctricos.

# Capítulo 2

# Manutenção de geradores eólicos - Estado da arte

O tema central deste trabalho versa sobre a "manutenção de sistemas de geração de energia eólica através de redes IP".

Neste capítulo é apresentado o actual estado do conhecimento em termos científicos. Numa primeira secção, será feita uma introdução bibliográfica do estado da arte dos diferentes métodos e técnicas existentes aplicadas em áreas genéricas e, depois, em particular, na área das turbinas eólicas.

Para além do panorama científico, numa segunda secção são também descritos quais os principais passos das intervenções de manutenção normalmente efectuados nos aero-geradores. Não obstante ser uma área onde é extremamente difícil obter informação, pois as empresas não permitem a divulgação dos dados reais, devido a questões de concorrência, ainda assim foi conseguida alguma informação fidedigna. Obteve-se esclarecimento sobre o esquema eléctrico de ligação de um parque à rede de transporte da EDP, bem como informação diversa sobre o sistema de gestão e dos automatismos da central de controlo, métodos de manutenção e, ainda, informação sobre diversas peças-de-reserva necessárias às intervenções de manutenção das turbinas eólicas.

## 2.1 Estado da Arte

### 2.1.1 Manutenção de sistemas genéricos

Uma das vertentes da manutenção é a da detecção e correcção de falhas. Neste âmbito, existem equipamentos de aquisição de dados que permitem monitorizar determinadas variáveis características de cada equipamento. Os dados recolhidos podem ser valores instantâneos, médios, eficazes, ..., ou valores espaciais, desde imagens térmicas aos RX, observados ao longo do tempo, de forma periódica ou aperiódica. Após a recolha dos dados, são comparados com dados de funcionamento

em regime normal, determinando-se a existência ou não de uma avaria. As técnicas de comparação dos dados são variadíssimas e dependem, em concreto, do comportamento das variáveis. Existem metodologias que efectuam a detecção em termos de: valor médio; valor eficaz; valor instantâneo; análise de frequência; análise através de redes neuronais; redes *neuro-fuzzy*; lógica *fuzzy*; processamento de imagem; entre outras.

*T.W.Verbruggen* [29] efectuou um estudo bastante detalhado sobre os métodos mais usados de análise de avarias tradicionais, bem como a forma como se podem aplicar ao caso dos geradores eólicos. Segundo este trabalho, as técnicas de monitorização de condição que podem ser usadas baseiam-se em análise de vibrações, óleo, termografia, condições físicas dos materiais, acústica, efeitos eléctricos, parâmetros do processo, inspecção visual, desempenho do sistema, sinais de sensores, tensão mecânica, etc.

A análise de vibrações é a mais utilizada em equipamentos rotativos. Pode ser utilizada nos geradores, turbinas, motores, bombas, compressores, caixas de transmissão, rolamentos, eixos, etc. Normalmente, os sensores utilizados são transdutores de posição para baixas frequências de rotação, sensores de velocidade na gama média de frequência, acelerómetros em altas-frequências e sensores de "*Spectral Emitted Energy*" em muita alta-frequência de rotação. No caso dos sistemas eólicos, este processo pode ser utilizado para monitorizar as rodas, os rolamentos da caixa de engrenagens, os rolamentos do gerador e o rolamento principal. As principais diferenças, neste caso, relativamente a outras aplicações, estão na dinâmica de carga e velocidades de baixa rotação dos sistemas eólicos.

Ainda relativamente às partes rotativas lubrificadas, a qualidade dos óleos é preponderante e, normalmente, verificada com regularidade. O método de análise de óleo tem assim dois objectivos, verificar a sua qualidade e, em simultâneo, a boa saúde das partes mecânicas lubrificadas. Normalmente, este método é feito *off-line* através da recolha de amostras. Existem, no entanto, várias aplicações onde são usados sensores para determinar o grau de contaminação do óleo e, a consequente necessidade de mudança do filtro ou determinação de existências de peças com desgaste excessivo.

A nível dos equipamentos eléctricos, onde a dissipação de potência através de calor é sempre um problema a equacionar, a termografia assume papel preponderante nessa análise. Esta é utilizada para identificação e monitorização de falhas em equipamentos electrónicos e eléctricos, normalmente por técnicos qualificados. As principais falhas detectadas por este processo incluem os pontos quentes devido à degradação dos componentes ou maus contactos. Existem empresas no mercado que já dispõem de *software* para efectuar este género de análise *on-line* embora ainda se esteja numa fase inicial. Na geração de energia eólica, esta técnica pode ser utilizada para detectar avarias no gerador e nos elementos de electrónica de potência, no entanto, a sua utilização é dispendiosa devido aos elevados custos dos equipamentos.

Outro aspecto importante é a tensão mecânica, uma técnica comum mas não muito utilizada na monitorização da condição de funcionamento. A medida desta variável pode ser muito útil para determinar o tempo de vida dos componentes, nomeadamente, para as pás do rotor. Actualmente, começam a aparecer os primeiros sensores de fibra óptica para este género de medida; no entanto,

ainda são demasiado caros.

Relativamente aos aspectos ambientais do ponto de vista sonoro, a monitorização acústica é utilizada não só para controlar este aspecto, mas está também fortemente relacionada com a análise de vibrações. Contudo, as diferenças existem ao nível da fixação dos sensores. Enquanto os sensores de análise de vibração têm de ser fixados rigidamente com o objecto, os sensores acústicos podem ser colocados em qualquer lugar desde que o som seja audível. Normalmente, esta técnica é utilizada para monitorizar os rolamentos das caixas de engrenagens.

Do ponto de vista dos componentes eléctricos e da detecção de fenómenos pouco usuais, é utilizada a análise de efeitos eléctricos, com variadíssimas técnicas disponíveis. Algumas destas são a medida de descargas, velocidade dos interruptores, medida da força de contacto de interruptores, falhas de isolamento em cabos, etc.

A monitorização de parâmetros do processo é muito usual e está directamente relacionada com os sistemas de controlo. A metodologia mais utilizada é através da comparação dos sinais com um limiar, resultando daí o alarme de falha. Para a análise de desempenho é, neste caso, aplicada a monitorização da relação entre a potência produzida, a velocidade do vento, a velocidade do rotor e o ângulo das pás do rotor. Caso existam desvios relativamente à normalidade será indicada uma falha.

### 2.1.2 Manutenção de sistemas eólicos

Nas pás do rotor pode ser utilizada a análise de tensão para estimar o tempo de vida, apesar dos métodos ainda não estarem completamente desenvolvidos e estabilizados. Outros métodos utilizados são a análise acústica e de vibração. Também pode ser utilizada a emissão acústica, à semelhança das técnicas de radares para determinar falhas nas pás do rotor.

O mecanismo que controla o ângulo das pás do rotor pode ser monitorizado através da medida das correntes/tempos e a diferença entre o ângulo desejado e o actual, dependendo da dinâmica dos motores. O sistema hidráulico para ajuste do ângulo das pás do rotor é um dos sistemas mais críticos, dado ser um sistema de segurança no funcionamento global das turbinas. As técnicas de monitorização utilizadas são idênticas às aplicadas na indústria.

As caixas de transmissão são muito utilizadas na indústria e as técnicas de monitorização de falhas já estão consolidadas. Tipicamente, nos sistemas eólicos, a monitorização é efectuada através da análise de vibrações vinda de diferentes sensores, emissão acústica e análise do óleo.

Nos geradores, os rolamentos podem ser analisados através de vibração. O estator e o rotor do gerador podem ser monitorizados por temperatura e termografia. Devido à carga não ser constante, é necessário efectuar análise de tendência baseada em técnicas de estimação de parâmetros para prever possíveis falhas antes de estas acontecerem.

Genericamente, o sistema de diagnóstico de um determinado equipamento poderá receber os sinais de entrada aplicados bem como as suas saídas; com base nestes, deverá indicar se existem falhas ou não. Um dos processos para determinar falhas baseia-se em estimadores. Um sistema de diagnóstico contém o modelo do sistema que está a monitorizar e, com base nas entradas, estima as saídas; seguidamente, compara a estimação (*residual fault analysis*) com o valor de saída. Outra

das variantes consiste num sistema de diagnóstico que, com base nas estradas/saídas, estima os parâmetros do modelo e, analisando a tendência dos parâmetros do modelo estimado, determina se existem falhas.

UPM - *Universal Process Modeling* é um algoritmo de modelização que tem a possibilidade de prever o comportamento de sistemas complexos [30, 31]. Este método baseia-se em exemplos de boa operação dos equipamentos para sintetizar equações que permitam descrever o sistema (modelização empírica). Inicialmente, tem de ser fornecido ao sistema um conjunto de dados referentes a um funcionamento normal e indicadas situações de alarme. Em modo normal, o sistema recebe entradas, faz estimativas dessas entradas e validação, fornecendo informação sobre uma possível falha do sistema ou de algum dos sensores que alimentam as entradas.

Através do programa JOULE III, financiado pela União Europeia, foi desenvolvida uma metodologia de avaliação estrutural da integridade das pás do rotor de aero-geradores [32]. O método analisa e monitoriza as emissões acústicas quando a turbina está em operação. O algoritmo baseia-se no reconhecimento de padrões dos sinais obtidos para avaliação automática da integridade física das pás. O método é actualmente comercializado e, com base nesta metodologia, prevê-se que possa vir a dar origem a uma norma europeia.

*Philip J. Morris* [33] elaborou um estudo para determinar e prever o ruído acústico produzido numa turbina eólica. O método é baseado na construção de modelos tendo em conta a física e a dinâmica dos componentes usados. Neste trabalho, é efectuado um estudo dos diferentes processos existentes na literatura (CAA, LES, PSU-WOPWOP, entre outros) para determinar o ruído em sistemas.

As técnicas de monitorização tradicionais têm sido usadas por fabricantes, com ligeiras alterações, permitindo fazer uma boa manutenção. Existe, no entanto, uma técnica utilizada pelo SWANwind (*software* comercial) [34] que monitoriza os equipamentos através do som, utilizando uma análise em alta-frequência da onda. Este *software* trabalha numa rede IP distribuída e vem integrado com o CMMS - *Condition Monitoring Maintenance System* [35]. Na prática, em vez de se efectuar uma análise de vibração, utiliza-se a análise do som para determinar a possibilidade de fricção e de impactos internos na turbina ou noutros componentes do sistema. Todos os valores recolhidos são transmitidos a uma central onde é efectuado o tratamento.

*P.Caselitz* e Outros, [36, 37, 38, 39, 40] utilizam vários sensores para determinar se existem avarias no sistema eólico. Através do processo implementado, são medidas a velocidade angular do rotor, vibração (3 sensores, 2 colocados na caixa de transmissão, um no gerador) e, por último, as correntes geradas à saída do gerador, através de pinças amperimétricas. Os autores determinam um modelo matemático aproximado do sistema e, através da análise em frequência dos sinais, correlacionam as variáveis do sistema em função da entrada (velocidade de rotação do rotor) e da saída (energia produzida), determinando assim se existe alguma anomalia [41].

Normalmente, nos motores eléctricos podem-se diagnosticar avarias medindo a corrente e tensão consumidas em vazio e em determinadas condições de carga. As técnicas são variadas, desde a análise do vector de *PARK*, à análise de vibrações, termografia, entre outras [42]. Contudo,

a análise de vibrações [43] é a mais difundida para equipamentos rotativos, permitindo detectar avarias mecânicas em rolamentos e outros elementos mecânicos [44].

A *GePower* [45] apresentou um sistema de *software* (*System1*) que permite efectuar a monitorização de sistemas eólicos, baseando-se na análise de vibrações através do espectro de frequência. Para este efeito, são utilizados acelerómetros ou transdutores de velocidade.

Viriato Marques [46] estudou um método para incorporar num sistema de informação o conhecimento de um técnico de manutenção. Esse conhecimento, muitas vezes, advém da experiência prática das pessoas. Nesse trabalho, estuda-se a forma de guardar esse conhecimento num sistema informático, sendo o *interface* com o utilizador o mais agradável possível. Esta informação, caso esteja disponível, pode ser utilizada por um sistema de monitorização de falhas, de forma a classificar o tipo de falha que pode estar a ocorrer.

*Zimin Yang* [47] efectuou o estudo da implementação da manutenção utilizando informação sobre o estado de funcionamento do sistema. Neste caso, são apresentados diferentes exemplos no domínio de sistemas mecânicos utilizando, entre outras ferramentas, as redes neuronais.

*R. Singh* e Outros [48] elaboraram um trabalho onde se utilizam redes neuronais para determinar falhas em sensores. Este é também um assunto que merece destaque, dado que a falha pode ser do sensor em vez do sistema. Os autores utilizam três redes neuronais encadeadas: A primeira identifica se existe falha no sistema; A segunda determina se a falha é dos sensores ou de componentes do sistema; A terceira rede neuronal identifica qual o sensor que falhou, caso seja a situação.

*H.N. Koivo* e Outros [49] utilizaram redes neuronais para identificar o modelo de motores de indução. É referido na literatura que uma rede neuronal com três camadas pode aproximar qualquer sistema, linear ou não linear. Desta forma, utiliza-se um conjunto de dados físicos obtidos experimentalmente para treinar a rede de forma a esta representar o modelo do motor. No caso em apreço, os autores utilizaram um modelo de motor para simular as avarias e assim validar o algoritmo proposto.

*L.B. Jack* [50], *Abhinav Saxena* e Outros [51] apresentaram uma técnica para análise de vibrações, usando uma rede neuronal em conjunto com algoritmos genéticos. Os algoritmos genéticos são utilizados para seleccionar os dados que serão apresentados à rede neuronal. Em [50], o autor reporta que, de entre 156 amostras, o algoritmo genético seleccionou 6 para treinar a rede e advoga que, desta forma, obteve uma taxa de sucesso de 100%.

Ainda na área dos algoritmos genéticos [52, 53, 54], são apresentados três pacotes de *software*, onde um é comercial [53], sendo os outros dois licenciados em *open-source*. Nestes artigos, são apresentados os avanços mais recentes da área (*Linear Genetic Programming*), aplicados à modelização de sistemas através de dados de amostra [55]. Nos exemplos descritos, os algoritmos genéticos a partir de uma solução de *software* inicial, reescrevem o *software* através de técnicas de mutação e recombinação genética, até optimizar uma função (o que se deseja encontrar) aos dados apresentados. No *software* [53], a modificação do programa é efectuada a nível das instruções *assembly* em *byte code*. Quando a solução óptima tiver sido encontrada, é feita a tradução de *assembly* para *Java* ou C. Segundo os autores, este método é mais rápido entre sessenta a duzentas

vezes relativamente às redes neuronais e tem sido utilizado com sucesso em áreas onde estas, normalmente, têm falhado.

Os algoritmos genéticos constituem um método de pesquisa num espaço de estados muito grande e, permitem encontrar a solução óptima através dos mecanismos de evolução presentes na natureza (cruzamento, os mais aptos sobrevivem, etc). A solução pode ser encontrada após muitas gerações e não existe garantia de ser a melhor solução (a solução é óptima até à geração actual).

Uma das particularidades das redes neuronais é a necessidade de serem treinadas. Para se treinar uma rede têm de estar disponíveis dados de amostra que consistem em saber para determinadas entradas quais são as saídas. De seguida e, através de várias iterações, adaptam-se as ligações neuronais até a rede responder correctamente aos dados de treino. Dependendo do algoritmo de treino e da rede, poderá existir ou não convergência. Findo o treino, a rede está pronta a ser utilizada. Na literatura existem variadíssimos modelos de redes neuronais, cada uma, dadas as suas especificidades, vocacionada para a resolução de determinados problemas. João Neto [56] desenvolveu e apresentou um compilador criado especificamente para a construção de redes neuronais, bem como a apresentação de vários exemplos de redes neuronais. *Won-Yong Lee* [57] apresentou uma rede neuronal que, segundo o autor, apenas precisa de uma iteração para fins de treino.

Com a crescente implantação da energia eólica, os investigadores têm desenvolvido modelos para simular o impacto que a produção de energia eólica tem nas redes de distribuição de energia. Para além deste estudo muitos outros têm sido feitos para analisar outras questões, nomeadamente, para diagnóstico de avarias [49]. Os modelos têm sido desenvolvidos por diversos investigadores dos quais se referem os seguintes: [58, 59, 60, 61, 62]. Alguns simuladores existentes são comerciais sendo outros disponibilizados para fins de investigação, como é o caso de [62]. Com esta *package* de *software* é possível simular uma turbina eólica, bem como diferentes distribuições para o vento, recolher variáveis intermédias simulando sensores, etc. Dada a flexibilidade deste *software*, é possível criar um modelo onde se possam injectar falhas em variáveis do sistema simulado e verificar, assim, qual o comportamento do sistema perante uma avaria. Desta forma, é possível estudar um mecanismo que possa detectar as avarias usando o modelo de simulação. Tal é importantíssimo, pois não é fácil usar um sistema real por questões de acessibilidade física.

Alguns dos algoritmos, técnicas, bem como constituição dos sistemas utilizados na monitorização de aero-geradores podem ser consultados em detalhe em [63, 64, 65].

### 2.1.3 *Software* de gestão da manutenção

"Sistema informático para gestão da manutenção é um programa organizado à volta de uma base de dados, permitindo programar e vigiar sob os três aspectos técnico, orçamental e organizativo, todas as actividades do serviço de manutenção e os meios desta actividade (serviços, máquinas, equipamentos, mão-de-obra, etc.) a partir de terminais disseminados nos gabinetes técnicos, oficinas e armazéns de *stocks*" [66].

Nesta secção pretende-se apresentar brevemente algumas aplicações de gestão da manutenção existentes no mercado sejam comerciais ou *open-source*. Evidentemente não serão indicadas todas

| ***Software*** | **Empresa** | **Licença** | ***URL* da empresa** |
|---|---|---|---|
| ManWinWin | Navaltik, Lda | comercial | www.navaltik.com |
| GTM | CTCV | comercial | www.ctcv.pt |
| ENGEMAN | ENGEMAN | *comercial* | www.engeman.com.br |
| DataStream | Infor | comercial | www.datastream.net |
| Maximo | Projetech, Inc | comercial | www.projetech.com |
| Sigma2008 | RedeIndustrial | *open-source* | www.redeindustrial.com.br |
| CWORKS | CWORKS | *open-source* | www.cworks.com.my |

Tabela 2.1: Exemplo de algumas aplicações da área da gestão da manutenção.

as possibilidades existentes e, caso se pretenda um estudo mais aprofundado, poderão consultar-se as seguintes referências para obtenção de outro *software* da área de CMMS [67].

Assim, das muitas possibilidades, destacam-se as da Tab. 2.1, sendo que, do ponto de vista de apresentação da análise das aplicações indicadas apenas serão expostos os aspectos principais de duas, uma comercial e outra *open-source*.

**ManWinWin da *Navaltik***

O ManWinWin, de origem nacional, desenvolvido pela *Navaltik Software*, empresa com implementação no mercado desde 1981, é um produto bem conseguido no que diz respeito à área da manutenção industrial, embora não cubra outras áreas também importantes, à imagem de outras aplicações concorrentes. A *Navaltik* é especializada no desenvolvimento de aplicações informáticas para organização e gestão de manutenção, área que começou a desenvolver desde 1990.

Esta empresa tem certificação ISO 9001:2008 e conta já com um grande número de clientes, sendo uma referência para as PMEs no domínio destes sistemas de gestão.

Está disponível em várias línguas além do Português (Inglês, Espanhol, Polaco, etc.), deixando a *Navaltik Software* a possibilidade de incorporação de outras línguas, desde que exista colaboração activa do cliente para tal.

Os recursos que oferece ao utilizador são:

- Registo e codificação de um parque de objectos de manutenção;
- Planeamento e gestão da manutenção industrial;
- Recursos específicos para gestão de frotas e gestão de calibrações (DMMs);
- Organização e codificação de materiais;
- Planeamento e registo de trabalhos;
- Documentação e gestão de pedidos à manutenção;
- Gestão de armazém;
- Gestão de encomendas;
- Análises à manutenção e quadros de bordo técnico, financeiros e evolutivos;
- Orçamento e controlo de gestão da manutenção.

É um produto fechado do ponto de vista de desenvolvimento de terceiros e o preço de aquisição inicia-se em 4.590,00 € em função das funcionalidades requisitadas.

***Sigma2008* da RedeIndustrial**

O *SIGMA* - Sistema Gerencial da Manutenção, é um programa *freeware* para manutenção, desenvolvido pela empresa Brasileira Rede Industrial. O *software* não limita o número de utilizadores ou de postos de trabalho (locais de instalação).

Na instalação do programa, escolhe-se qual a área de actuação que interessa, dentro das opções disponíveis (Industrial, Predial, Hospitalar, Hotelaria, *Shopping*, Assistência Técnica, Prefeituras, e Escola). Este programa funciona tanto em *Windows* como em *Linux* e foi desenvolvido com base na linguagem *DELPHI 2006* permitindo o funcionamento em rede *Ethernet*. O programa possui aplicativos *Web* sendo possível solicitar Ordens de Serviços, pedido de matérias-primas, visualizar relatórios entre outros, sendo necessário apenas um servidor *on-line*.

Oferece todos os recursos do *software ManWinWin* podendo ser personalizado perante o pagamento do serviço. O motor de base de dados utilizado é o *FireBird*, descendente do *Interbase*.

Apesar desta aplicação ser *open-source*, o seu código fonte não se encontra disponível para alterações.

**Outros programas informáticos**

Similar situação pode ser verificada para os restantes programas informáticos da Tab. 2.1 bem como outros não listados. Assim, no decorrer da pesquisa sobre esta temática, optou-se por redesenvolver, no âmbito dos trabalhos desta dissertação, um *software* de gestão da manutenção para ambiente *Windows* cuja última versão estável tinha sido desenvolvida para *VAX/VMS* [68].

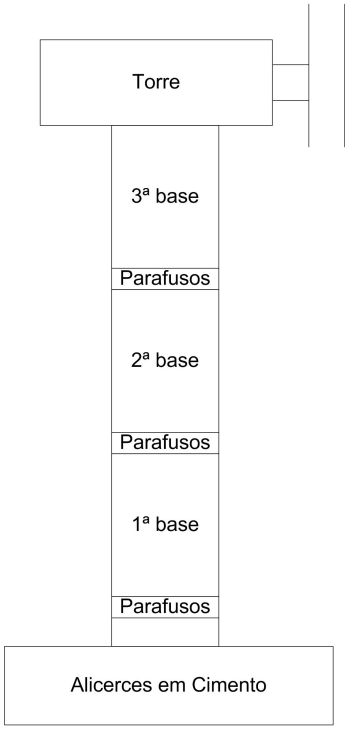
As vantagens por esta opção são evidentes: o código fonte torna-se aberto a todos os investigadores do grupo com mais-valias evidentes a nível da adição de novos módulos e possibilidade de reformulação de todo o sistema.

O *software* desenvolvido denomina-se SMIT-Sistema Modular Integrado de Terologia.

## 2.2 Acções de Manutenção de parques eólicos

Conforme é do conhecimento geral, as turbinas eólicas devido às forças elevadas a que estão sujeitas, partes mecânicas, etc, necessitam de manutenção especializada e regular. As turbinas em si são constituídas pelo alicerce (base) em cimento, no qual é fixado, por meio de parafusos, uma primeira base cilíndrica, da qual consta a porta de acesso. De seguida, são fixados mais dois andares cilíndricos. O último andar cilíndrico só pode ser fixado em simultâneo com a torre (*nacelle*) - no conjunto existem assim três andares - (por norma).

A Fig. 2.1 representa, em termos de blocos, os diferentes componentes de uma turbina eólica. Cada um dos componentes é fixo através de parafusos. Para enfatizar a necessidade de manutenção, à direita na mesma figura, apresenta-se um aero-gerador da ilha de Tenerife, onde são visíveis, a olho nu, zonas com ferrugem, na união entre os três blocos tubulares e pás do rotor. Todos os componentes representados na figura necessitam, regra geral, de manutenção e, em termos de inspecções periódicas, alguns dos aspectos mais importantes para verificar são os seguintes:




(a) Principais blocos na construção da turbina



(b) Turbina junto ao mar

Figura 2.1: Exemplificação de um aero-gerador.

- **Base da turbina** - inspecção do cimento onde está assente a turbina;
- **Portas e acessos** - verificar se funcionam normalmente;
- **Apertos estruturais** - parafusos que fixam as diferentes estruturas da turbina;
- **Apertos de parafusos** - parafusos genéricos de variadíssimos componentes, como motores, cabos, geradores, etc;
- **Linha de segurança** - linha da vida, trata-se do sistema que em caso de avaria do elevador permite aos técnicos descer em segurança com equipamento e cinto de segurança;
- **Ruídos e vibrações** - devem ser verificados especialmente a parte do rotor, gerador e caixa de engrenagens, caso exista;
- **Corrosão na pintura** - exterior e interior, normalmente documentada com fotografias;
- **Rachas, fissuras e soldas** - normalmente, no exterior, esta verificação é efectuada através de binóculos. A nível das pás do rotor, caso exista alguma fissura, são imediatamente substituídas, o que implica uma grua, tendo a duração dos trabalhos a estimação de uma semana - caso exista grua disponível;
- **Travões** - os diferentes componentes de travagem, travão hidráulico do rotor, da *nacelle*, travão de disco do sistema de *pitch* e travão de parqueamento;
- **Ventiladores** - troca dos filtros dos ventiladores;

- **Sistema eléctrico** - verificação de cablagem;
- **Engrenagens** - troca do óleo de três em três anos, embora, por questões ambientais, se tente chegar aos cinco anos. São cerca de 250l a 300l de óleo e, após a troca são enviadas amostras regulares do óleo para análise em laboratório, a fim de aferir sobre aspectos de qualidade deste, se existem partículas metálicas, indicando assim danos nos rolamentos;
- **Rolamentos** - de seis em seis meses é trocada a massa de lubrificação, sendo enviada para análise a massa substituída. Incluem-se aqui os rolamentos do sistema de *pitch*.

Alguns aspectos importantes prendem-se com os reapertos dos parafusos estruturais que são feitos com chave dinamómetrica - existe um valor de força indicado pelo fabricante em Nm. Caso seja encontrado algum parafuso partido, são substituídos todos os quatro parafusos adjacentes a este e é criado um relatório de avaria para a empresa instaladora do sistema. Em termos de curiosidade, para uma turbina de 2MW, duas pessoas levam cerca de três horas a efectuar os reapertos dos três andares. Os reapertos estruturais, em termos das pás do rotor, levam cerca de um dia por pá quando feitos por duas a três pessoas. Quando se trata da manutenção das pás do rotor, a turbina é parada e a pá colocada na posição horizontal. O técnico pode-se deslocar, sensivelmente, até meio do interior da pá, onde efectua as acções de verificação.

Em termos de motores eléctricos, o sistema de *pitch* das pás do rotor é comandado por 3 motores DC de 24 volts, havendo uma protecção de alimentação com baterias. Este sistema de protecção da alimentação é essencial em termos de falta de energia, dado que permite alterar o ângulo das pás e, assim, evitar danos elevados em situações onde simultaneamente falte a energia eléctrica e existam regimes de ventos elevadíssimos. Em termos do sistema de orientação da *nacelle* é conseguido através da utilização de um motor trifásico com o auxílio de um variador de velocidade. A *nacelle* pode rodar cerca de duas voltas de $360^o$. Os cabos que chegam à *nacelle* estão suspensos desde o topo da turbina até à base de cimento e como são flexíveis permitem torções e, portanto, não são afectados pelas duas voltas da *nacelle*.

Outro dos aspectos importante é o das vibrações. O gerador é assente em cinoblocos de borracha - um género de sistema anti-vibração - e o acloplamento entre a caixa de engrenagens e o gerador é feito por peças em borracha, de modo a não serem transferidas vibrações mecânicas. Algumas turbinas incluem um sensor de vibração do género pendular na *nacelle*, cuja função é monitorizar a vibração da estrutura. Sempre que accionado, todo o sistema é desligado necessitando de ser rearmado presencialmente para entrar novamente em funcionamento. Compreende-se como o "sistema é desligado", o *parking* da *nacelle*, através da alteração do ângulo de *pitch* das pás para a posição de repouso e o accionamento de todos os travões hidráulicos.

Todo o sistema eléctrico da turbina é monitorizado por um processador com conectividade TCP/IP (na base da turbina) que, através de sensores, monitoriza a temperatura nos rolamentos (engrenagem - tipicamente $60^o$, gerador - tipicamente $40^o$), através de sondas pt100. A monitorização da energia produzida é efectuada através de Transformadores de Corrente (TI) com uma relação de 2000A/0,5A. Nos diferentes patamares, conforme a necessidade de sensores, são colocados módulos escravos com rede industrial. Existem ainda sensores de nível hidráulico para os diferentes sistemas, nomeadamente, lubrificação e radiadores de dissipação.




Figura 2.2: Exemplo de alguns passos na construção de uma turbina eólica. Fotos: KV Venti.

Do ponto de vista eléctrico, os geradores trabalham a 690 volts, em sistema trifásico. O inversor, constituído por conversor AC-DC-AC, converte a energia do gerador para 380 volts e é também um dos componentes fundamentais do sistema. No aero-gerador existe um transformador trifásico de 380 volts para 20kV. A ligação entre a estação de transformação e cada turbina é feita por cabo subterrâneo à tensão de 20kV (incluí-se aqui também o cabo/fibra de comunicações TCP/IP). Na estação de transformação, os 20kV são transformados em 60kV para transmissão da energia. Relembra-se que os valores em vez do par 20kV/60kV podem ser também 15kV/30kV, conforme o fabricante do aero-gerador utilizado no parque eólico. Obviamente que os transformadores são bidireccionais, significa isto que o gerador eléctrico da turbina pode consumir ou produzir energia. Normalmente, consome energia na fase de arranque do rotor e na fase de não produção. Em termos do sistema eléctrico, normalmente, as avarias acontecem a nível da passagem de cabos da *nacelle* para as pás do rotor, onde se encontram os motores de accionamento do sistema de *pitch*.

Genericamente, estes são alguns dos passos que os responsáveis pela manutenção de um parque eólico devem analisar, sendo que, normalmente, sempre que existam situações anómalas, devem ser devidamente documentadas, com fotografias, imagens termográficas, gráficos e reportadas à empresa responsável pela manutenção (tipicamente os fabricantes do material instalado). Outra curiosidade interessante é a necessidade do pessoal da manutenção fazer testes de descida em altura, à semelhança dos bombeiros pois, a qualquer momento, podem ter que descer da turbina eólica pela parte exterior, ou seja, descer mais de 100m, pelo lado de fora, seja qual for o regime de ventos. Aliás, este é um teste obrigatório para as pessoas que pretendem seguir esta profissão.

A Fig. 2.2 exemplifica alguns dos passos da construção de um aero-gerador, desde as fundações em cimento com a tubagem para passagem dos cabos a 20kV e cabo de comunicações, montagem dos diferentes andares tubulares, montagem da *nacelle* até ao rotor e pás do rotor. Incluem-se ainda fotos da vista interior da zona tubular e do interior da *nacelle*.

## 2.3 Conclusões

Este capítulo apresenta o estado da arte na área de manutenção de aero-geradores e nas áreas afins cuja aplicação advenha de outras actividades humanas, relacionadas com equipamentos que sejam semelhantes aos dos aero-geradores.

A nível de aplicações informáticas de gestão de manutenção são apresentadas algumas das hipóteses presentes no mercado, quer a nível comercial, quer a nível de *open-source*. Resumidamente, mostram-se algumas das características mais importantes de duas aplicações, uma de cada uma das vertentes, bem como as principais características necessárias para a gestão da manutenção de aero-geradores.

O capítulo termina com uma apresentação das principais acções de manutenção que devem ser efectuadas pelas equipas de manutenção de aero-geradores, tendo como base a informação recolhida num exemplo prático de um aero-gerador no centro do país.

# Capítulo 3

# As energias renováveis e os geradores eólicos

Neste capítulo descrevem-se os tipos de energias renováveis com maior relevância, bem como as formas mais usuais para efectuar o seu aproveitamento. É ainda feita uma resenha histórica da energia eólica bem como da evolução das diferentes tecnologias utilizadas nos aero-geradores, as suas vantagens e desvantagens e, ainda, as tendências de evolução futura.

Parte da informação apresentada neste capítulo, é baseada em livros da especialidade, bem como em artigos da internet [2, 69] sobre energia eólica, indicando-se ao longo do texto as respectivas referências nas outras áreas das energias renováveis.

## 3.1 Motivação para o uso de energias renováveis

As energias renováveis denominam-se assim por permitirem a produção de energia limpa, sem emissões de poluentes para a atmosfera, a partir de recursos naturais que a natureza sistematicamente repõe ou, por outras palavras, não se extinguem. Esta preocupação é evidente e tem sido claramente impulsionada nos últimos anos pelos diferentes países. Assistiu-se ao desenvolvimento e investigação para geração de energia eléctrica a partir do vento, da energia solar, da energia das ondas, o aproveitamento do calor solar para aquecimento e arrefecimento de edifícios, técnicas inovadoras para iluminação de edifícios através da luz ambiente exterior, etc. Sem entrar em discussões sobre o facto dos materiais utilizados nas diferentes tecnologias serem produzidos através de energias limpas ou não, a certeza é que a preocupação pelo aquecimento global é uma realidade, a par com a suposta escassez do petróleo antecipadamente prevista para o futuro.

O mercado das energias renováveis cresceu ao longo do século XX e manterá a tendência neste século XXI. O forte investimento nestas áreas está directamente ligado a questões, tais como, a tendência crescente de escassez de petróleo e o seu elevado preço nos mercados internacionais e, ainda as mudanças ambientais produzidas pela poluição. A razão do preço elevado do petróleo

deve-se, entre outros factores, à previsão de esgotamento das reservas num prazo de 100 anos e às novas economias emergentes, que requerem elevadas quantidades deste recurso para manterem os seus níveis de crescimento elevados.

Segundo a *International Energy Agency*, a necessidade de energia em 2030 será 60% superior à actual e, destes 60%, dois terços serão da China e Índia, entre outras economias emergentes [70].

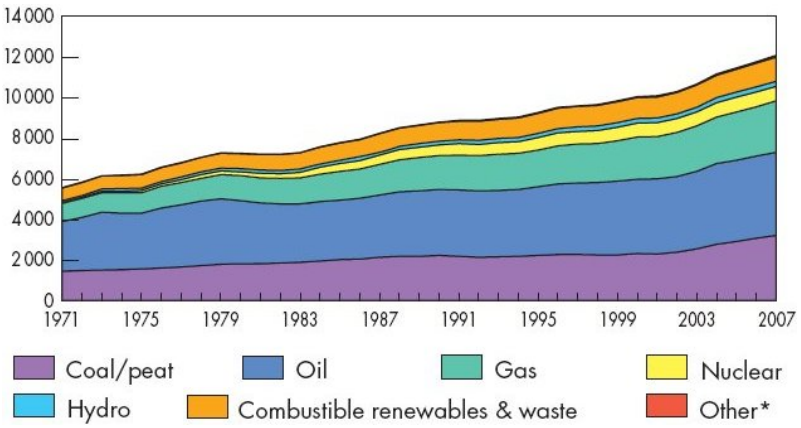
As Figs. 3.1(a), 3.1(b) e 3.1(c) indicam as tendências crescentes do consumo de energia com o consequente aumento de emissões de $CO_2$. A crise do preço do petróleo é evidente pelo incremento constante do preço nos mercados internacionais. Todas estas razões mostram a necessidade de uma mudança no paradigma energético.

Para além deste aspecto, as energias renováveis têm ainda como impulsionador outra forte alavanca, as mudanças ambientais produzidas pela poluição. Cada vez mais este problema tem ganho ênfase, com diversos encontros mundiais entre governos e instituições, que levaram à assinatura do protocolo de Quioto no passado e a reactualizações no presente. Sabe-se hoje que é imperioso efectuar mudanças a nível das políticas energéticas, levando à criação de recursos energéticos mais amigos do ambiente. Não se trata só de produzir energia amiga do ambiente, trata-se também de "educar" os consumidores para a poupança de um bem, que é um recurso escasso.

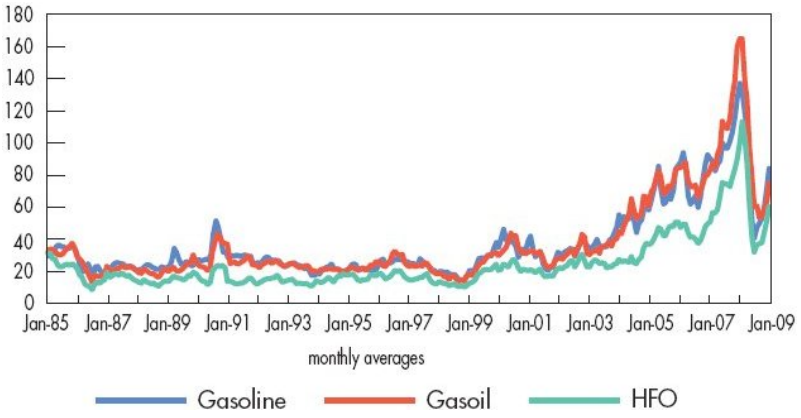
Outro mercado em "revolução" é o automobilístico, que já se encontra em transformação face aos preços elevados dos combustíveis e à necessidade de poluir menos. Em Portugal, o avanço político pioneiro é reconhecido, com a forte aposta no veículo eléctrico, concretizável nos próximos anos, ao qual se juntam projectos para estudo do impacto na rede eléctrica dum aumento de consumo devido às futuras mudanças. O INESC Porto é o líder científico de um desses projectos - MERGE (*Mobile Energy Resoures for Grids of Electricity*) - que visa preparar o sistema eléctrico europeu para a massificação da utilização de veículos automóveis eléctricos. No sector automobilístico salienta-se ainda, à data, a existência de algumas soluções mais amigas do ambiente. Os motores híbridos, veículos a ar comprimido, células de combustível, etc, são alvo de uma forte aposta de investigação. A ideia transmitida pelas autoridades portuguesas aponta o sector como demasiado poluente, sendo o principal responsável pelos níveis de poluição existentes no país (emissão de $CO_2$, ...), muito acima do sector produtivo Industrial.

Em termos de poluição, a produção de energia eléctrica em Portugal ocupa o primeiro lugar. Para colmatar este aspecto e o preço elevado de produção, há em Portugal muitos defensores da energia nuclear. Os principais aspectos atractivos, segundo os defensores deste tipo de energia, são o preço e uma menor poluição (em termos quantitativos). No entanto, a poluição da energia nuclear é extremamente perigosa e mantém-se activa por milhares de anos. Outro dos aspectos negativos é o dos recipientes onde são guardados esses resíduos, cujo tempo de vida é muito inferior aos resíduos que contêm no seu interior. A única energia nuclear sem contra indicações ambientais é a fusão nuclear, que ainda se encontra em fase de investigação, nomeadamente, em Inglaterra e nos Estados Unidos da América. Na Europa, o projecto *JET - Joint European Torus* é um dos maiores a nível mundial na área, com participação portuguesa [24].

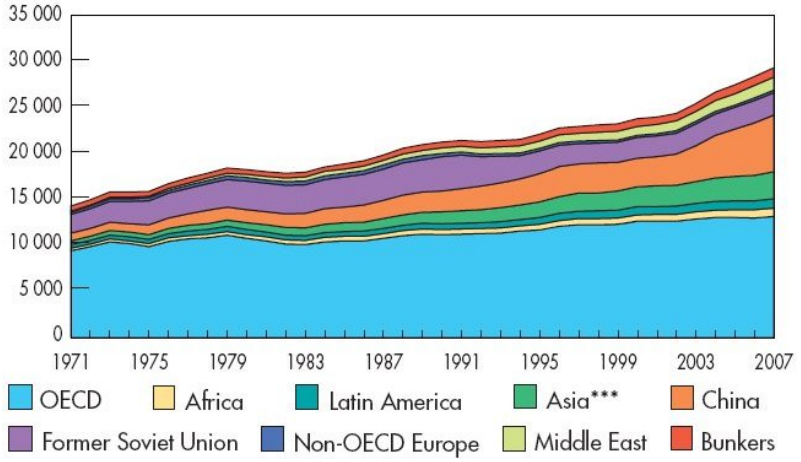
A própria sociedade precisa hoje de mais comodidades que, normalmente, implicam um aumento do consumo da energia eléctrica, entre outras. Este aumento do consumo da energia não




(a) Energia consumida até ao ano 2007 (Mtoe) [70];




(b) Evolução do preço em dólares do barril de petróleo em *Rotterdam* [70];




(c) Emissões de $CO_2$ em Mt [70];

Figura 3.1: Principais indicadores energéticos e poluentes.

pode ser efectuado através do incremento do consumo de petróleo, pois terá como consequência a violação de tratados internacionais.

Nesta sequência de acontecimentos e condicionalismos, a grande motivação e desafio das energias renováveis (não poluentes) é a de responder a estes problemas, aproveitando recursos até agora desaproveitados e potenciá-los, desempenhando assim um papel muito importante no controlo dos problemas ambientais associados ao efeito de estufa, entre outros.

A incorporação de fontes renováveis, reduzindo o peso das centrais térmicas no sistema produtor eléctrico, é uma medida de grande importância para reduzir as emissões gasosas, visando o cumprimento do estabelecido em tratados internacionais.

Na sua génese, as energias renováveis englobam diferentes sectores de produção de energia, que têm vindo a ser estudados pela União Europeia, Estados Unidos da América, Instituições, Universidades, empresas do sector e pela comunidade mundial em geral.

Dentro das energias renováveis, encontramos diversos grupos temáticos dos quais se salientam os seguintes:

- Solar fotovoltaico;
- Solar térmico;
- Biogás;
- Geotermia;
- Solar Passivo/Edifícios;
- Biocombustíveis;
- Oceanos;
- Biomassa;
- Mini-Hidrícas;
- Energia Eólica.

Em Portugal, das diversas Instituições ligadas ao sector, salienta-se a ADENE/INETI (Agência para a Energia/ Instituto Nacional de Engenharia e Tecnologia Industrial), que têm vindo a produzir trabalho neste sector, tendo em 2002 editado um relatório síntese sobre as "Energias Renováveis em Portugal" [71]. O INETI é responsável ainda pela construção de uma turbina eólica urbana denominada *Turban*.

A Directiva Comunitária [72] para a promoção da electricidade gerada a partir de fontes renováveis de energia transformou em objectivos concretos algumas das ideias que vinham desde há muito a ser debatidas, estabelecendo metas concretas para os Estados-Membros e diferenciando as contribuições esperadas das grandes hídricas e das outras fontes renováveis. Alguns dos problemas apontados à construção das grandes centrais hídricas, surgem também no caso das mini-hídricas, pelo que, apesar de se esperar que também esse sector possa conhecer novos e interessantes desenvolvimentos, é da energia eólica que se espera o maior incremento da contribuição das fontes renováveis na convergência para os objectivos.

## 3.2 Outros tipos de energias renováveis

### 3.2.1 Solar fotovoltaico

A produção de energia a partir da energia solar, utiliza como princípio o efeito fotovoltaico. Basicamente, trata-se de materiais que, ao serem expostos à luz (fotões), produzem corrente eléctrica (electrões). No caso do material ser semicondutor, esta corrente é produzida pela diferença



Figura 3.2: Nova célula com 42,8% de eficiência à luz ambiente será comercializada a partir de 2010.

de potencial criada nas faces opostas da junção do semicondutor. Os semicondutores utilizados pertencem ao grupo IV da tabela periódica, em especial, silício e germânio. As eficiências de conversão variam entre 15% e 25%, sendo que, no primeiro ano de operação, alguns painéis, devido ao seu processo de fabrico, perdem 50% da sua eficiência. O tempo de vida é de 10 a 20 anos conforme a tecnologia utilizada. Tais aspectos tornam o sistema solar fotovoltaico pouco atraente [73] e caro apesar do sucesso de instalação no mercado residencial.

No entanto, em Junho de 2007, *Christiana Honsberg* e *Allen Barnett*, investigadores da Universidade de *Delaware*, EUA [74], construíram uma nova célula com eficiência de 42,8% (Fig. 3.2) em condições normais de iluminação, sendo o seu objectivo atingir os 50%. As novas células solares de altíssima eficiência utilizam um novo sistema de concentração óptica lateral, que divide a luz solar em três diferentes níveis de energia - alto, médio e baixo - dirigindo-os para o interior de células solares feitas com materiais de diferentes sensibilidades à luz. Assim, tira-se proveito da maior parte do espectro electromagnético, evitando o desperdício de luz. As novas células têm o dobro da eficiência das suas congéneres actualmente no mercado, estando prevista a sua chegada ao mercado a partir de 2010.

A 16 de Junho de 2009, *John F. Geisz* e *Daniel Friedman*, anunciavam ter conseguido criar uma célula solar com eficiência entre 42% a 50%, cujo processo de fabrico consiste na inversão do processo de crescimento cristalino, que dá origem à célula. São utilizadas células solares multijunção, na prática metamórficas de tripla junção invertida [75].

Inúmeros avanços científicos são apresentados todos os anos, quer no processo quer na composição do fabrico das células fotovoltaicas, dos quais se salientam os seguintes:

- Pelo Instituto *Fraunhofer-Gesellschaft* [76] foram relatados aumentos de eficiência de 20% com a utilização de perfuração com raios laser para criar os filamentos condutores das células solares;

- A tese de doutoramento de *Sergio Hiroshi* [77] relata um método que utiliza átomos de ruténio na construção da célula solar, indicando uma eficiência de 80%, embora a durabilidade temporal das propriedades seja limitada;

- Alunos do MIT desenvolveram uma espécie de sistema de automação para acompanhamento da trajectória solar, sem motores. A ideia foi transposta do mecanismo existente nas plantas e permite alterar a orientação do painel de acordo com a direcção da luz solar[1] baseado apenas nas propriedades dos materiais que o suportam [78];

- Na área dos painéis solares flexíveis, a empresa *NanoSolar* colocou no mercado, em Janeiro de 2008, os primeiros produtos desta natureza, cuja característica principal é o baixo custo por Watt, cerca de um dólar americano. Estes painéis são produzidos por impressão [79] e melhoraram o desempenho em cinco vezes relativamente a painéis idênticos;

- No MIT, em Março de 2008, investigadores aumentaram a eficiência das células solares policristalinas em 27%, passando a sua eficiência a ser comparável à eficiência das monocristalinas, com um factor de conversão de 19,5%. As inovações passaram pelo aumento da captação de luz e da introdução de superfícies rugosas para os raios "baterem" várias vezes na célula e converterem mais energia. Um dos investigadores, Dr. *Sachs*, salienta que os novos revestimentos antirreflexivos devem fazer baixar o custo das células solares policristalinas relativamente ao custo de produção do carvão, por volta de 2012. Os investigadores criaram uma empresa para comercialização [80] do novo produto;

- No início de 2008, no laboratório nacional de *Idaho*, investigadores [81] criaram uma célula solar capaz de captar energia, mesmo depois do sol ter desaparecido do horizonte. A ideia passou por construir uma célula que convertesse a luz visível e a luz infravermelha, que continua a ser irradiada pela superfície terrestre para a atmosfera após o pôr-do-sol. Para além deste facto, a grande novidade é a eficiência de 80% de conversão no espectro do infravermelho.

Estes são apenas alguns exemplos relacionados com a área dos painéis fotovoltaicos, i.e., conversão da energia solar em electricidade directamente pelo material que compõe o painel.

Por fim, não poderia deixar de ser referido o facto de, actualmente, o governo Português incentivar claramente a instalação de painéis fotovoltaicos, através do financiamento do preço produzido por kW.

### 3.2.2 Solar térmico, ou solar térmico-eléctrico

Normalmente, as tecnologias na área do solar térmico aproveitam o calor para aquecimento de águas sanitárias. O conceito em si é simples: coloca-se um colector com um permutador térmico para transferir o calor do fluído do painel à água. Este tipo de produtos está amplamente divulgado, sendo os painéis de tubos de vácuo uma referência na área.

De seguida, descrevem-se algumas tecnologias, não para aquecimento mas para geração de energia a partir de calor. Uma dessas tecnologias, que se encontra em franca investigação, é a do aproveitamento da energia solar para produção de energia utilizando motores *Stirling* [4], Fig.

[1] Do Inglês *Heliotropic solar panels*



Figura 3.3: Sistema de aproveitamento solar para produção de energia eléctrica utilizando um motor de *Stirling* [4, 5].

3.3. Basicamente, esta tecnologia utiliza a reflexão solar através duma espécie de antena parabólica espelhada para concentrar a energia solar num ponto. Em termos médios, a energia do sol por $m^2$ é cerca de 1 kW. O motor de *Stirling* foi inventado por *Robert Stirling* na Escócia por volta de 1816 e, normalmente, é constituído por 2 cilindros selados com um gás no seu interior. O princípio de funcionamento do motor é baseado no facto do gás aquecido expandir e, quando comprimido, aquecer, o que permite movimentar os cilindros e, por último, accionar um gerador de energia eléctrica (Fig. 3.4). A sua eficiência é de 40%, muito superior aos motores de combustão, com a vantagem de que, nesta aplicação em concreto, as emissões serem zero, tratando-se, portanto, de uma energia renovável. Curiosamente, este processo revela-se mais eficiente que a energia fotovoltaica, sendo, actualmente, alvo de grande investigação.

Ainda nesta área, existem avanços em motores de conversão de calor para energia eléctrica, nomeadamente, investigadores da Universidade de *Utah* [82], EUA, criaram um motor termo-acústico, que converte calor em ondas sonoras e, de seguida, pelo efeito piezo-eléctrico é criada electricidade. As aplicações são inúmeras, dado que estes geradores podem ser incluídos em equipamentos eléctricos e utilizar o calor dissipado para criar energia, aproveitando assim um recurso que estava a ser desaproveitado.

### 3.2.3 Solar-Hidrogénio

As energias renováveis baseadas na energia solar-hidrogénio tem por objectivo principal produzir hidrogénio utilizando a energia solar. O hidrogénio produzido tem como destino as células de combustível sendo, à data, apontado pelos investigadores como uma das energias alternativas à substituição do petróleo no sector dos transportes - o combustível do futuro. Actualmente, praticamente todo o hidrogénio utilizado na indústria é produzido a partir da queima de combustíveis fósseis.

A União Europeia tem largamente apoiado a investigação nesta área, tendo financiado o projecto *Hydrosol* [6], que tinha como objectivo a criação de um reactor para produção de hidrogénio

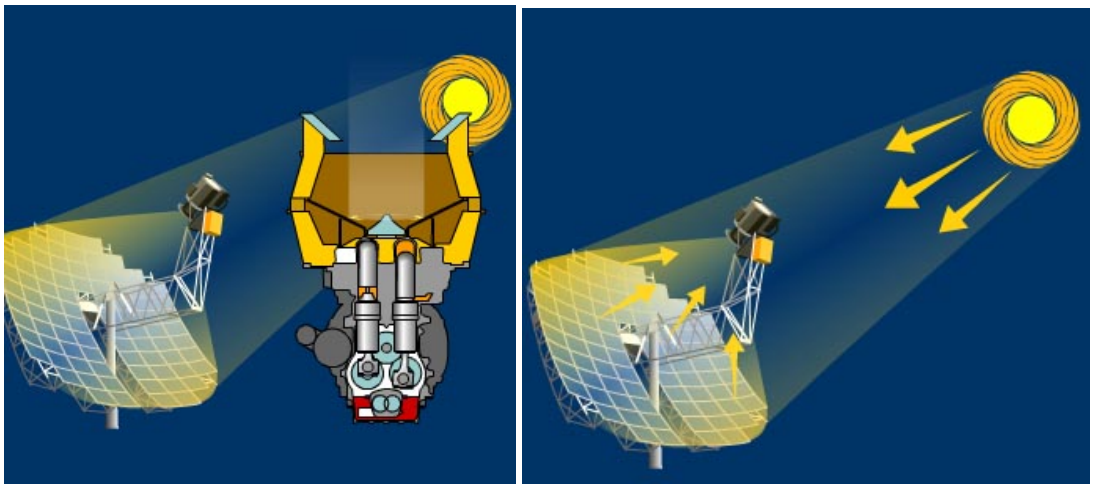


Figura 3.4: Princípios de funcionamento do aproveitamento solar para produção de energia eléctrica.

através da energia solar. Em Julho de 2007, foi tornado público pelos responsáveis por este projecto o anúncio da construção de um reactor capaz de produzir hidrogénio a partir da água, utilizando apenas a energia solar. Uma das características do reactor é a sua simplicidade técnica. O princípio é idêntico ao utilizado em 2003 por uma equipa de investigadores da Universidade de *Massachusetts* [83], com a diferença de ser mais eficiente. O processo actual atinge uma eficiência de 70% em contrapartida aos 30% do processo criado pela Universidade Americana.

O dispositivo criado pelos investigadores Americanos utiliza a energia eléctrica produzida por painéis fotovoltaicos em conjunto com o espectro de infravermelho da luz solar. Os raios infravermelhos aquecem a água até 600º C. A esta temperatura a água vaporiza e o vapor é injectado numa solução alcalina que força a divisão em hidrogénio e oxigénio através da energia eléctrica dos painéis - electrólise.

O reactor criado pela equipa do projecto *Hydrosol* - Fig. 3.5 - utiliza um corpo de cerâmica porosa cujos canais estão revestidos por um catalisador. A luz solar é concentrada através de espelhos e o calor produzido evapora a água, que, ao evaporar, é obrigada a passar pelos poros da cerâmica. O catalisador decompõe a molécula de água em hidrogénio e oxigénio. Este processo, contrariamente ao anterior, dispensa a utilização da energia eléctrica e, segundo os autores, atinge uma eficiência de conversão de 70%. Contudo, um dos problemas da utilização do hidrogénio é o seu armazenamento, dado que este fica no estado gasoso e é altamente reactivo com outras moléculas.

## 3.3 A Energia Eólica

A energia eólica (tal como apresentado na secção 1.2, pág. 3) resulta do deslocamento de massas de ar. A ideia principal é aproveitar a energia cinética presente nesses deslocamentos. A deslocação de massas de ar é criada devido às diferenças de temperatura na superfície do planeta,

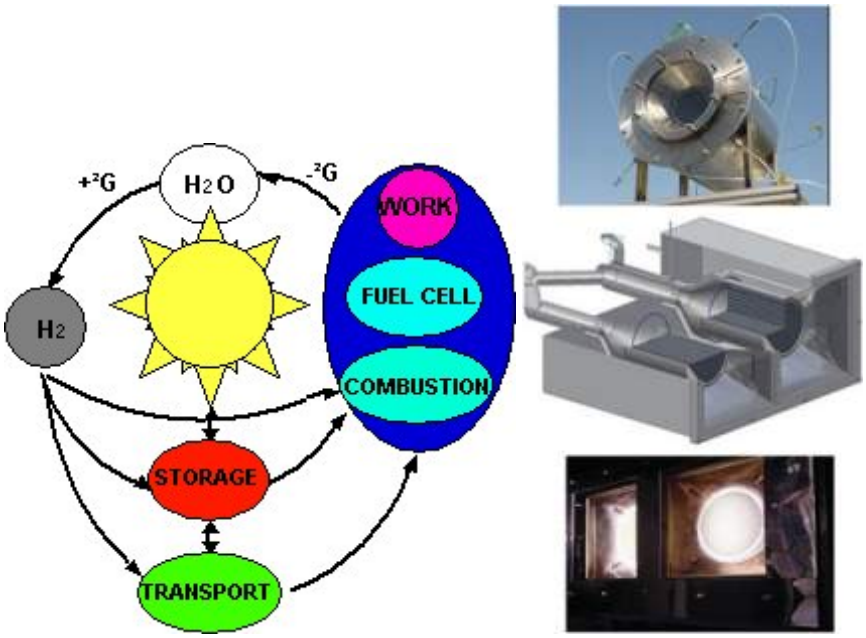



Figura 3.5: Esquerda, o ciclo proposto para as células de combustível, utilizando hidrogénio, o que origina uma energia limpa sem emissões. Direita, fotos do reactor criado pelos investigadores do projecto *Hydrosol* [6].

criadas pela incidência da energia solar no nosso planeta. Na prática, os ventos formam-se devido à radiação solar, em conjunto com o movimento de rotação do planeta e fenómenos naturais repetitivos. Devido às causas que lhe dão origem é considerada uma energia renovável. Assim, nesta secção, descreve-se o aproveitamento deste recurso na perspectiva da produção de energia eléctrica sendo, pontualmente, indicados outros géneros de aproveitamentos.

### 3.3.1 Breve história da energia eólica

A energia eólica tem sido utilizada desde os tempos mais antigos. Ao certo, quando foi a sua primeira utilização, não é conhecido, no entanto, julga-se que foram os egípcios, por volta de 5000 a.c., os primeiros a utilizarem o vento [84]. Tudo indica que no ano 4000 a.c. já se fazia a utilização da energia eólica em embarcações, testemunhado por barcos encontrados num túmulo da época. Em 2800 a.c. o vento ajudou a conduzir barcos ao longo do rio Nilo, em conjunto com os remos. No ano 1000 a.c. os fenícios foram pioneiros na construção de barcos utilizando exclusivamente a força dos ventos. Ao longo dos séculos as embarcações foram evoluindo com o destaque para as caravelas, surgidas na Europa no século XIII, que permitiram as Descobertas Marítimas, nas quais Portugal teve um papel preponderante e de liderança. Com o surgimento dos navios a vapor, a utilização do vento em embarcações teve a sua quebra por volta de 1807.

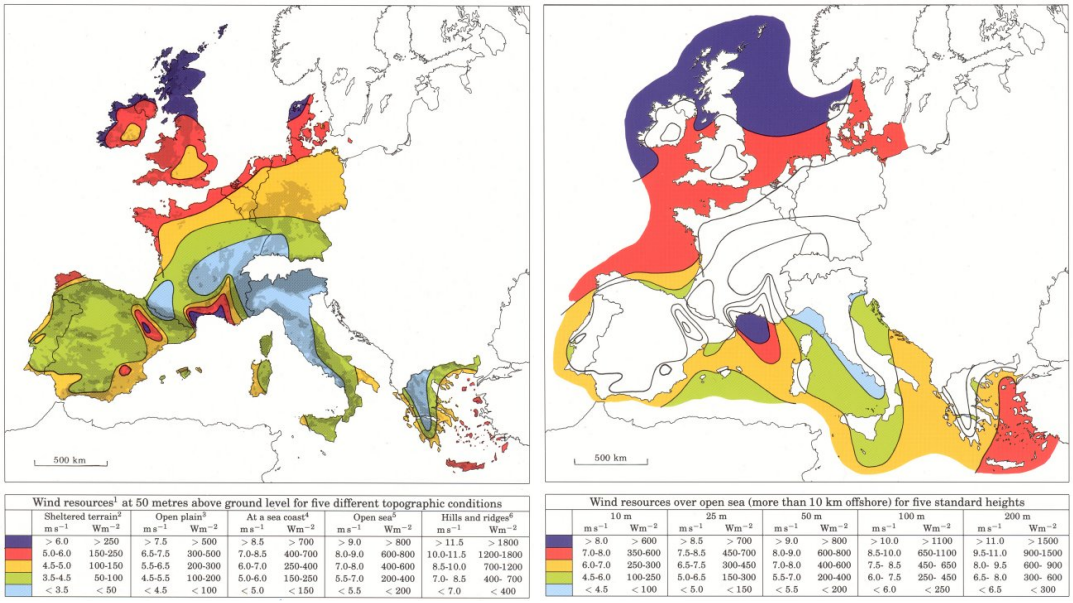
A energia do vento também foi utilizada nos veículos. O carro de *Nassau*, foi inventado no ano de 1600 pelo holandês *Maurício de Nassau*, na sequência da guerra da independência entre a Holanda e a Espanha. Este inventor idealizou um canhão que era impulsionado por velas de

forma idêntica aos barcos. O sistema de propulsão foi construído com madeira e lona, permitindo, com ventos fortes, percorrer uma distância de 80 km em duas horas, o qual veio a surpreender os espanhóis num ataque surpresa. Foi o primeiro veículo a não depender de músculos para se movimentar. Nos anos 70/80 surgiram nos EUA veículos de lazer com o sistema de vela, os quais foram muito utilizados em praias, tendo sido baptizados de *windcars*.

No que respeita à utilização de moinhos de vento (podemos chamá-los os predecessores das actuais turbinas eólicas), julga-se que os persas foram pioneiros em 700 d.c. ao construir tal engenho para a moagem de grãos. No entanto, existem registos referentes a 200 a.c. de moinhos sem o objectivo da moagem. As civilizações do médio oriente, nomeadamente, os muçulmanos, no século XI, utilizavam intensivamente os moinhos para a produção de alimentos, moagem e bombagem de água, tendo, provavelmente, a ideia chegado à Europa pelos participantes das cruzadas. Os eixos motores dos moinhos de vento eram, nesta fase, projectados com direcção fixa em função da direcção predominante do vento. Na Europa, os holandeses tiveram um papel preponderante ao adaptarem esta tecnologia, introduzindo a sua utilização na drenagem de terras e inventando os moinhos com cúpula giratória. Este facto permitiu uma maior adaptação dos moinhos aos ventos europeus que tinham como principal característica a variação de intensidade e direcção. A forma de direccionar a cúpula do moinho, segundo o vento, era manual, mas com a invenção da ventoinha em 1750, passou a ser automático.

A primeira notícia da utilização da energia eólica para a produção de energia surgiu em 1887 nos EUA, sendo também creditado aos Dinamarqueses desenvolvimentos nesta área [85]. As turbinas de vento, como eram chamadas, aparecem na Dinamarca em 1890. No entanto, é a partir de 1930 que nos EUA se utilizam em grande escala os mini-geradores eólicos para carregamento de baterias, destinando-se esta utilização ao meio rural, onde ainda não tinha chegado a energia eléctrica. Nos anos de 1950 e 1960 na Europa e, nomeadamente, em França, foram desenvolvidos geradores de 100 kW a 300 kW. A Alemanha construiu, nesta fase, geradores eólicos para fornecer energia extra para a sua linha de distribuição. Pelos anos 40, já nos EUA existia a maior turbina instalada, com uma potência de 1.25 MW. Com a revolução industrial e a introdução da máquina a vapor, a utilização dos geradores eólicos caiu em desuso e a investigação nesta área diminuiu fortemente. Outro importante factor para esta diminuição foi a electrificação das zonas rurais que oferecia energia eléctrica a preços baratos. O ressurgimento da energia eólica (energia renovável) dá-se na década de 70 devido à crise petrolífera e devido à degradação do meio ambiente, em consequência da utilização de energias poluentes. Este foi o principal acontecimento para o relançamento da energia eólica que, actualmente, é vista como uma das mais promissoras fontes de energia renovável.

A revolução industrial evidenciou a principal desvantagem da energia eólica em relação a outras energias - a sua produção inconstante - que acontece devido à ocorrência irregular e intensidade variável do vento. Foi no século XV que o famoso inventor, *Leonardo da Vinci*, desenhou ou inventou o primeiro sistema de controlo de potência de que há conhecimento, o travão de cintas aplicado à roda accionadora e a inclinação das pás em relação ao horizonte, que foram largamente utilizados nos moinhos de vento até ao século XVII.




Wind resources[1] at 50 metres above ground level for five different topographic conditions

| | Sheltered terrain[2] | | Open plain[3] | | At a sea coast[4] | | Open sea[5] | | Hills and ridges[6] | |
|---|---|---|---|---|---|---|---|---|---|---|
| | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ |
| | > 6.0 | > 250 | > 7.5 | > 500 | > 8.5 | > 700 | > 9.0 | > 800 | > 11.5 | > 1800 |
| | 5.0-6.0 | 150-250 | 6.5-7.5 | 300-500 | 7.0-8.5 | 400-700 | 8.0-9.0 | 600-800 | 10.0-11.5 | 1200-1800 |
| | 4.5-5.0 | 100-150 | 5.5-6.5 | 200-300 | 6.0-7.0 | 250-400 | 7.0-8.0 | 400-600 | 8.5-10.0 | 700-1200 |
| | 3.5-4.5 | 50-100 | 4.5-5.5 | 100-200 | 5.0-6.0 | 150-250 | 5.5-7.0 | 200-400 | 7.0- 8.5 | 400- 700 |
| | < 3.5 | < 50 | < 4.5 | < 100 | < 5.0 | < 150 | < 5.5 | < 200 | < 7.0 | < 400 |

Wind resources over open sea (more than 10 km offshore) for five standard heights

| | 10 m | | 25 m | | 50 m | | 100 m | | 200 m | |
|---|---|---|---|---|---|---|---|---|---|---|
| | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ | m s$^{-1}$ | Wm$^{-2}$ |
| | > 8.0 | > 600 | > 8.5 | > 700 | > 9.0 | > 800 | > 10.0 | > 1100 | > 11.0 | > 1500 |
| | 7.0-8.0 | 350-600 | 7.5-8.5 | 450-700 | 8.0-9.0 | 600-800 | 8.5-10.0 | 650-1100 | 9.5-11.0 | 900-1500 |
| | 6.0-7.0 | 250-300 | 6.5-7.5 | 300-450 | 7.0-8.0 | 400-600 | 7.5- 8.5 | 450- 650 | 8.0- 9.5 | 600- 900 |
| | 4.5-6.0 | 100-250 | 5.0-6.5 | 150-300 | 5.5-7.0 | 200-400 | 6.0- 7.5 | 250- 450 | 6.5- 8.0 | 300- 600 |
| | < 4.5 | < 100 | < 5.0 | < 150 | < 5.5 | < 200 | < 6.0 | < 250 | < 6.5 | < 300 |

Figura 3.6: Mapas dos recursos eólicos em terra e nas zonas costeiras da Europa [7].

### 3.3.2 O vento e o seu comportamento

É extremamente importante conhecer o comportamento do vento para caracterizar o potencial de produção em função dos custos de investimento por cada aero-gerador (cerca de 1 milhão e 600 mil euros, potência 1,6 MW). Assim, é necessário qualificar as variações do vento, diárias, sazonais e anuais, assim como, as variações de curta duração, tal como previamente descrito na secção 1.2.

Todas estas variações são importantes, destacando-se as de curta duração - turbulência - pois indicam a necessidade de adequada robustez física do aero-gerador e, em muitos casos, podem inclusivé, inviabilizar a instalação.

Com o objectivo de caracterizar os ventos terrestres e marítimos, foi criado o Atlas Europeu do vento [7], cujos mapas disponíveis mais recentes estão representados na Fig. 3.6.

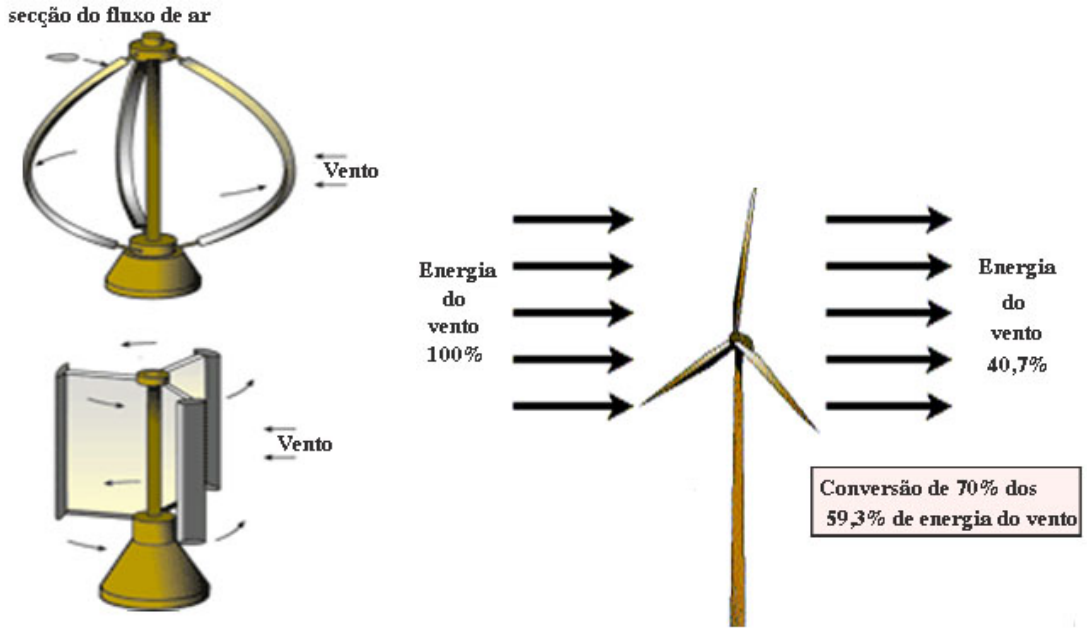
### 3.3.3 As várias tecnologias eólicas

Nesta secção são introduzidos os conceitos associados à tecnologia eólica, sendo descritos os diversos tipos de turbinas, bem como os elementos que as constituem. Um sistema eólico é constituído por vários elementos que, em conjunto, devem proporcionar o maior rendimento possível. Para efeitos de estudo da energia eólica, devem ser considerados os seguintes componentes:

- **Vento** - Disponibilidade energética do local destinado à instalação [7];
- **Rotor** - Responsável por transformar a energia cinética do vento em trabalho mecânico de rotação;

- **Transmissão e caixa multiplicadora** - Responsável por transmitir a potência fornecida pelo rotor ao gerador. Alguns sistemas não utilizam este componente, sendo o rotor ligado directamente ao gerador;
- **Gerador eléctrico** - Responsável por converter a energia mecânica em energia eléctrica;
- **Mecanismo de controlo** - Responsável pela orientação do rotor, controlo de velocidade e controlo do gerador;
- **Torre** - Responsável por posicionar e sustentar o rotor e demais elementos na altura conveniente à captação da energia do vento;
- **Transformador** - Responsável pelo acoplamento eléctrico entre o gerador e a rede eléctrica;
- **Acessórios** - São os componentes periféricos.

Existem no mercado diferentes soluções para sistemas eólicos. As turbinas eólicas costumam ser classificadas pela posição do eixo do seu rotor, que pode ser vertical (Fig. 3.7(a)) ou horizontal (Fig. 3.7(b)). Uma máquina horizontal tem as pás a girar num eixo paralelo à terra. Uma máquina vertical tem as pás a girar num eixo perpendicular à terra. Em cada um dos dois tipos de turbinas há uma variedade de arquitecturas possíveis que são fornecidas pelos diversos fabricantes (número de pás, formato das pás, etc).




(a) VAWT - *Vertical Axis Wind Turbine*

(b) HAWT - *Horizontal Axis Wind Turbine*

Figura 3.7: Diferentes tipos de geradores com indicação do limite de *Betz* 59,3%.

| Porte | Potência | Utilização |
|---|---|---|
| Pequeno | Até 80 kW | Consumo local, com eventual fornecimento à rede da energia excedente |
| Médio | De 80 kW a 500 kW | Consumo local ou só fornecimento à rede |
| Grande | Superior a 500 kW | Fornecimento de energia à rede |

Tabela 3.1: Classificação das turbinas eólicas em termos da potência produzida.

A turbina de eixo horizontal é a mais comum, pois, normalmente, é mais eficiente no aproveitamento da energia do vento. Outra das suas particularidades é poder rodar horizontalmente de forma a captar melhor a direcção do vento, sendo a captação de ventos em *upwind* ou *downwind* (conceito directamente ligado à aerodinâmica das pás - no primeiro caso, o vento sopra da frente do rotor e no segundo caso sopra de trás).

As turbinas eólicas podem ainda ser classificadas em termos da potência nominal (Tab. 3.1) [35], e/ou em termos do diâmetro/área varrida pelo rotor (Tab. 3.2).

#### 3.3.3.1 Turbinas eólicas de eixo vertical

A principal vantagem das turbinas de eixo vertical é não necessitarem de mecanismo de controlo de direcção, simplificando bastante os mecanismos de transmissão de potência; como desvantagem, apresentam o facto das pás, devido ao movimento de rotação, verem constantemente alterados os ângulos de ataque e de deslocamento em relação à direcção dos ventos, gerando forças resultantes alternadas, o que, para além de limitar o seu rendimento, causa vibrações acentuadas em toda a estrutura. As turbinas de eixo vertical utilizam, na sua maioria, o rotor *Darrieus*. Podem ainda ser equipadas com o rotor *Savonius*, que se adequa a velocidades mais baixas do vento, contudo, têm uma baixa eficiência. Assim, em termos de rotor, tem-se:

- **Rotor *Darrieus*** - Este tipo de rotor tem a vantagem de estar sempre voltado para a direcção do vento (ver Fig. 3.7(a) - cimo). As pás têm a forma de catenárias e o gerador e sistemas de controlo são instalados no solo, na base da turbina. Têm uma eficiência inferior aos geradores de eixo horizontal. Este rotor foi inventado pelo Francês *Darrieus* tendo registado a sua patente em 1931. Na *internet* encontram-se inúmeros exemplos académicos para construção de uma mini turbina deste género [11];

| Porte | Diâmetro do rotor (m) | Área varrida pelo rotor ($m^2$) |
|---|---|---|
| Pequeno | Até 16m | Até 200 |
| Médio | De 16 a 45 m | de 200 a 1600 |
| Grande | Superior a 45m | Maior de 1600 |

Tabela 3.2: Classificação das turbinas eólicas em termos do diâmetro e área varrida pelo rotor.



Figura 3.8: Diferentes tipos de *rotors Savonius* [8, 9].

- **Rotor *H-Darrieus*** - As pás são semelhantes à letra H, sendo a sua construção ainda mais simples que a precedente. Neste tipo de rotor, as pás podem ser instaladas à mesma altura das turbinas de eixo horizontal e, desta forma, podem produzir, sensivelmente, a mesma energia. O controlo de potência pode ser semelhante aos das turbinas de eixo horizontal (ver Fig. 3.9);

- **Rotor *Savonius*** - Apresenta a sua curva de rendimento em relação à velocidade próxima da do rotor de múltiplas pás de eixo horizontal, mas numa faixa mais estreita e de menor amplitude. Utiliza duas pás e, como foi referido, a sua eficiência é baixa, entre 15 e 20%. Existem várias versões, das quais algumas se encontram representadas na Fig. 3.8. Este rotor deu origem a variantes, tendo como objectivo a melhoria da eficiência. O rotor de *Lenz* é um desses exemplos (Fig. 3.9). Na *internet* encontram-se inúmeros exemplos académicos para construção de uma mini turbina deste género [86]. Na Fig. 3.8, a turbina da esquerda é fabricada pela *HelixWind* [8], que produz para o mercado doméstico aero-geradores com potências até aos 5 kW.

#### 3.3.3.2 Turbinas eólicas de eixo horizontal

As turbinas eólicas de eixo horizontal utilizam, maioritariamente, rotores de uma, duas ou três pás (sistemas utilizados tipicamente em parques eólicos de elevado aproveitamento, Fig. 3.10), sendo, actualmente, utilizado o rotor de três pás. No entanto, em sistemas para aproveitamento de ventos com baixa velocidade, é típico serem utilizados rotores de múltiplas pás. O motivo do número de pás ser pequeno está relacionado com os seguintes aspectos:

- Maior velocidade de rotação;

- Diminuição dos custos no fabrico das pás;

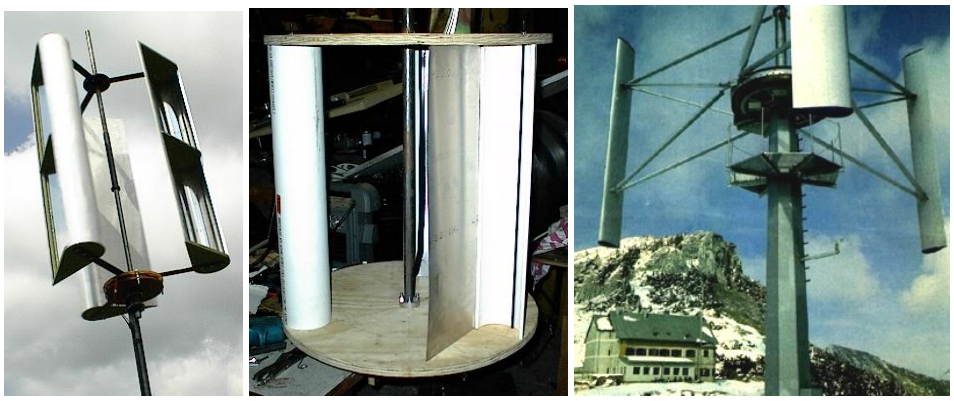


Figura 3.9: Exemplos do rotor de *Lenz* (esquerda e centro). Direita: Rotor *H-Darrieus* [10, 11, 12].

- Redução de esforços mecânicos.

Assim, em termos de rotor para as turbinas de eixo horizontal, tem-se:

- **Rotor de três pás** - Este rotor é, na generalidade, o padrão utilizado nas turbinas modernas, devendo-se ao facto da grande relação de potência extraída por área de varredura do rotor ser muito superior ao rotor de múltiplas pás (no entanto, só em velocidades de vento elevadas). Apresenta porém baixos valores de binário de partida e de rendimento para velocidade baixas, características que, apesar de aceitáveis em sistemas de geração de electricidade, incompatibilizam o seu uso em sistemas que requeiram altos momentos de força e ou carga variável. As turbinas de três pás têm velocidades de rotação inferior às de duas e de uma pá. Normalmente, o ângulo das pás é também controlado de forma a permitir alterar a incidência do vento;

- **Rotor de duas pás** - Este rotor tem uma velocidade de rotação superior ao anterior, no entanto, obriga a esforços mecânicos maiores. Os problemas aerodinâmicos também são mais complexos, sendo necessário levar em consideração os esforços de torção e flexão;

- **Rotor de uma pá** - Este rotor é o que consegue atingir a maior velocidade de rotação, permitindo reduções no custo da caixa de velocidades e do gerador. No entanto, apresenta um desequilíbrio maior e os esforços mecânicos são também maiores;

- **Rotor de quatro ou mais pás (múltiplas pás)** - Estes rotores estão limitados em velocidade, devido ao número de pás, sendo utilizados em sistemas de bombagem de água. As suas características tornam o seu uso mais próprio para baixas velocidades do vento, devido a disporem de uma boa relação entre o binário de partida e a área de varredura do rotor. O melhor rendimento é extraído em regime de ventos fracos, limitando assim a potência máxima extraída por área do rotor, que não é das melhores, o que torna este tipo de rotor pouco indicado para geração de energia eléctrica.



Figura 3.10: Turbinas de eixo horizontal com rotores de três pás [13].

As turbinas de eixo horizontal, contrariamente às de eixo vertical, não estão alinhadas com o vento, sendo necessário um mecanismo que permita o posicionamento do eixo do rotor em relação à direcção do vento, para um melhor aproveitamento global, principalmente onde existe muita mudança na direcção dos ventos.

O alinhamento do rotor pode ser obtido através de diversas técnicas que, por sua vez, dependem do porte da turbina:

- Auto direccionamento - Ocorre pelo posicionamento do centro aerodinâmico da *nacelle* (ver secção 3.3.5), que é excêntrico ao eixo da torre. Este método só é utilizado quando o rotor está em rotação, por motivos mecânicos;
- Direccionamento com auxílio de leme aéreo - Por razões mecânicas só pode ser utilizado em turbinas de pequeno porte;
- Direccionamento motorizado - Actualmente, o mais utilizado, tem duas vertentes:
  - Efectuado através de um pequeno rotor de eixo horizontal, perpendicular ao eixo do rotor principal que, através de uma caixa de redução, move a *nacelle* e o rotor até que este esteja paralelo ao sentido do vento. Este mecanismo era muito utilizado nos antigos moinhos de vento;
  - Efectuado através de um accionamento eléctrico ou hidráulico, onde o eixo do rotor é posicionado paralelamente ao sentido dos ventos. Este mecanismo é o mais utilizado em termos das turbinas de eixo horizontal de grande porte.

As turbinas de eixo horizontal podem ser classificadas de acordo com o posicionamento do rotor relativamente à direcção do vento:

- Rotor a barlavento (*upwind* rotor) - O rotor está a montante da torre, à frente, captando o vento pela parte da frente;
- Rotor a sotavento (*downwind* rotor) - O rotor está a jusante, atrás, captando o vento pela parte de trás.



Figura 3.11: Parque eólico marítimo de *Vinderby* na Dinamarca.

### 3.3.4 Locais de Instalação

As turbinas eólicas podem ser instaladas em terra ou em mar. Já existem, inclusivé, novas ideias em investigação para aproveitamento do vento de alta altitude através de turbinas eólicas aéreas, dado que a intensidade do vento aumenta com a altura. De acordo com os locais onde as turbinas são instaladas, a sua denominação é diferente. Quando existe mais do que uma turbina, normalmente, denomina-se de parque eólico (*wind farm*). O recurso principal para a instalação de turbinas é a existência de vento com potencial eólico [7].

#### 3.3.4.1 Parques eólicos terrestres *(on-shore)*

As primeiras utilizações das turbinas foram em ambiente terrestre, sendo os parques eólicos terrestres os mais comuns. Uma das vantagens é a facilidade de deslocação de equipas de manutenção por via terrestre, contrariamente aos parques marítimos.

#### 3.3.4.2 Parques eólicos marítimos (*off-shore)*

As condições normalmente existentes no mar são mais interessantes devido à falta de obstáculos. Desta forma, o vento circula de uma forma mais homogénea. A Dinamarca tem liderado este sector, com cerca de 400 MW instalados no final de 2006 (Fig. 3.11).

#### 3.3.4.3 Turbinas eólicas rurais

Foram utilizadas largamente em locais onde a energia eléctrica ainda não tinha sido instalada, dando alguma qualidade de vida às populações. Foram utilizadas nos EUA para este efeito, conforme descrito na secção 3.3.1. Actualmente, surgem como uma possibilidade para fornecer

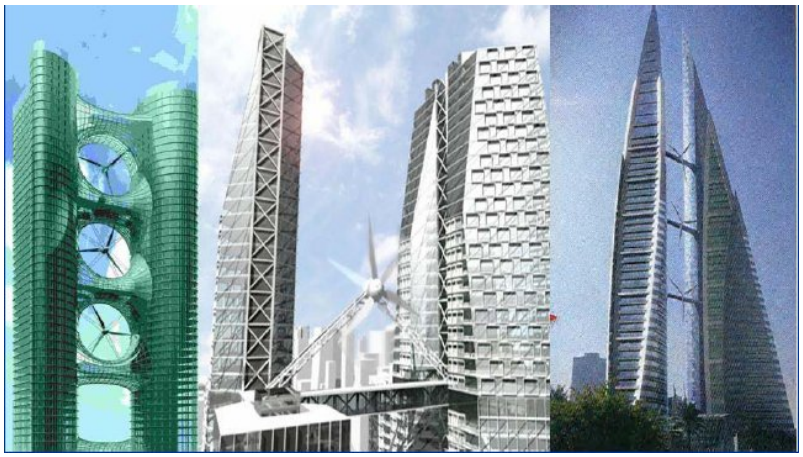


Figura 3.12: Tecnologias que poderão equipar as cidades de forma a aproveitar os ventos citadinos, neste caso claramente bem integradas no meio envolvente.

electricidade a sistemas de rega em campos e contribuir para as finanças domésticas, dado que os particulares podem vender este género de energia aos operadores eléctricos.

#### 3.3.4.4 Turbinas eólicas urbanas

O aproveitamento da energia eólica em ambiente urbano (Figs. 3.12 e 3.13), é já hoje uma preocupação real, tendo a União Europeia, através do contrato (JOR3-CT98-0270, que decorreu de 1998 a 2000) financiado um consórcio de universidades para estudar a aplicabilidade das diferentes tecnologias de turbinas nas cidades [87].

Outra tecnologia que está a ter grande sucesso nos países nipónicos, é a das micro-turbinas (*Motorwind*) eólicas para aproveitamento dos ventos citadinos (Fig. 3.13). A tecnologia foi desenvolvida por investigadores da Universidade de *Hong-Kong*, em conjunto com a empresa *Motorwave Ltd* [88]. Foram instaladas 18 unidades, cada uma contendo 22 turbinas. A potência estimada para ventos de 5m/s é de 15 W e a 10m/s de 125 W, num total de 2,25 kW a 10m/s para todo o sistema. A estimação de produção é de 10 a 30 kW diários, tendo o sistema custado cerca de 2500 USD.




Figura 3.13: Tecnologia *motorwind*, que usa rotores de baixa potência agrupados para construir um sistema de grande dimensão.

### 3.3.5 Componentes do Sistema - turbina de eixo horizontal mais comum

Na Fig. 3.14 ilustram-se os principais componentes da turbina de eixo horizontal, a mais comum, que será descrita no restante texto desta secção.

Do ponto de vista de um observador externo, uma turbina eólica é constituída por três componentes principais: rotor, cabine (*Nacelle*) e torre. No entanto, de forma mais detalhada (Fig. 3.14), salientam-se os seguintes componentes: cabine; radiador de arrefecimento; gerador; painel de controlo; carcaça principal; isolamento de ruídos; sistema hidráulico; caixa de velocidades; isolamento de ruídos; mecanismo de orientação direccional; veio do rotor; radiador de arrefecimento; orientação das pás do rotor; rotor *hub* e cone do rotor.

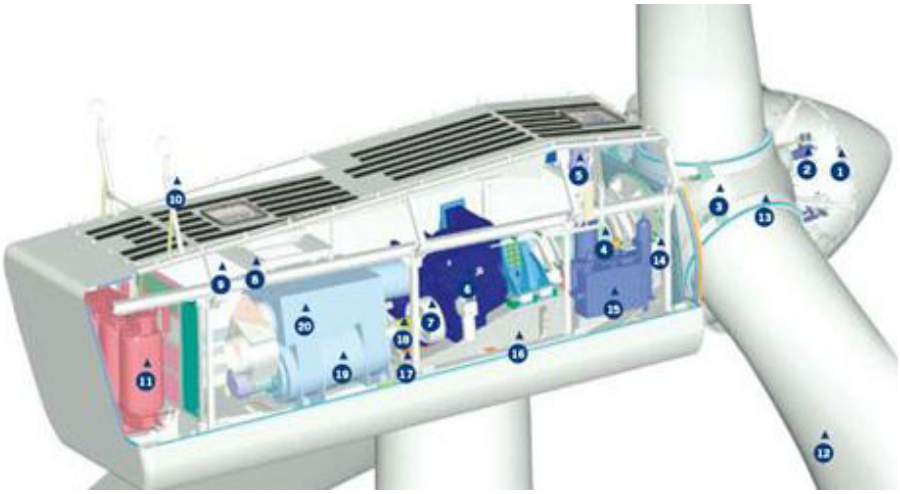
#### 3.3.5.1 O rotor

O rotor é responsável por transformar a energia cinética do vento em movimento circular do eixo do motor. Normalmente, é constituído por três pás (mais de 90% das instalações). Muitas vezes, são incorporados no rotor mecanismos de controlo de potência e de direccionamento para alinhamento das pás com a direcção do vento. Na literatura define-se *Solidity* (Solidez) como sendo a razão entre a área total das pás e a área varrida pelas mesmas. Se o diâmetro e a solidez das pás forem mantidos constantes, o rendimento aumenta com o número de pás, porque diminuem as chamadas perdas de extremidade. Bastante relevantes são os materiais utilizados para a construção do rotor, dado que este elemento é o que suporta a força do vento. Os materiais utilizados são, a madeira e compostos sintéticos, nomeadamente, plásticos reforçados com fibra de vidro e fibra de carbono. A escolha da utilização dos materiais relaciona-se com, o preço, a moldabilidade, as propriedades mecânicas e a resistência à fadiga. A tendência actual aponta para o desenvolvimento de novos materiais compósitos híbridos, de forma a tirar partido das melhores características de cada um dos componentes, designadamente, peso, robustez e resistência à fadiga.

#### 3.3.5.2 Cabine

Na cabine estão alojados os diferentes componentes, nomeadamente, veio transmissor, travão, gerador, caixa de velocidades, mecanismo de orientação direccional e controlo de paragem de emergência. Existem ainda sensores incorporados na cabine, no exterior, nomeadamente, um anemómetro medindo a velocidade e direcção do vento, servindo como entrada para o sistema de controlo que, por sua vez, permite a entrada em funcionamento a partir dos 5m/s e a paragem para velocidades superiores a 25m/s. A informação do vento é usada para o sistema de orientação direccional.

Do ponto de vista da construção da turbina eólica, quando se coloca o último elemento da torre (normalmente três secções, Fig. 2.2, pág. 19), é obrigatório instalar a cabine, caso contrário, a estrutura poderá colapsar devido à força do vento.



(a) Cabine com gerador tradicional (*Vestas*); 1-Controlador do Cubo; 2- Controlo *pitch*; 3- Fixação das pás no cubo; 4- Eixo principal; 5- Aquecedor de óleo; 6- Caixa multiplicadora; 7- Sistema de travagem; 8- Plataforma de serviços; 9- Controladores e Inversores; 10- Sensores de direcção e velocidade do vento; 11- Transformador de alta tensão; 12- Pás; 13- Rolamento das pás; 14- Sistema de travagem do rotor; 15- Sistema hidráulico;16- Plataforma da *nacelle*; 17- Motores de posicionamento da *nacelle*; 18- Luva de acoplamento; 19- Gerador; 20- Aquecimento de ar;




(b) Cabine com gerador multipólos (*Enercon*); 1- Apoio principal da *nacelle*; 2- Motores de orientação da *nacelle*; 3- Gerador em anel (multipólos); 4- Fixador das pás ao eixo; 5- Cubo do rotor; 6- Pás; 7- Sensores de direcção e velocidade do vento;

Figura 3.14: Principais elementos de uma Turbina eólica [14].

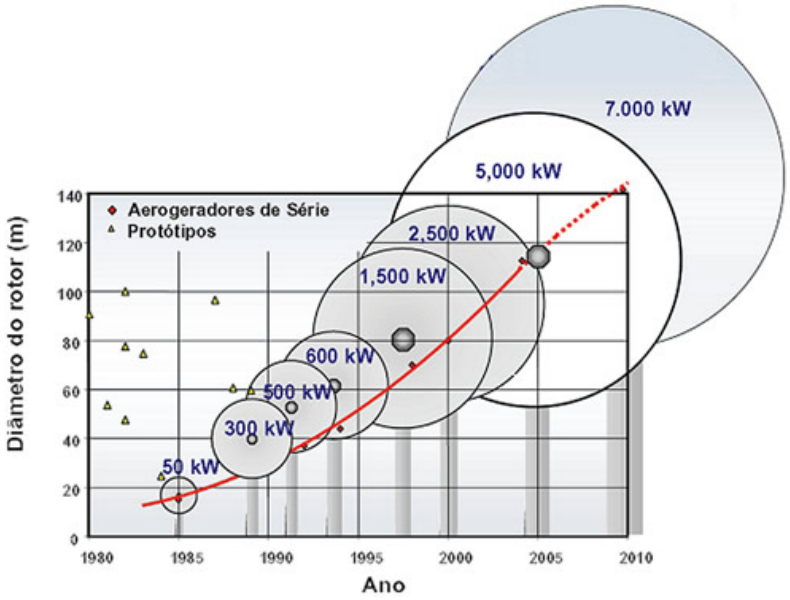



Figura 3.15: Relação de altura da torre e a potência da turbina eólica [14].

#### 3.3.5.3 A torre

As torres são necessárias para suportar a cabine, o rotor e para o posicionar à altura conveniente para um melhor aproveitamento do potencial eólico. Podem ser fabricadas em betão, aço ou ferro. Existem dois tipos de torres, tubulares e entrelaçadas, podendo ou não ser sustentadas por cabos tensores. A torre está sujeita a grandes esforços e tem que ser cuidadosamente dimensionada, dado o seu tempo de vida útil ser elevado, em média, 20 anos.

Normalmente, conforme referido, é feita em ferro, em três peças, unidas por parafusos e com a forma cilíndrica. A Fig. 3.15 mostra a relação da altura da torre em função da potência do aero-gerador e a sua evolução ao longo dos anos.

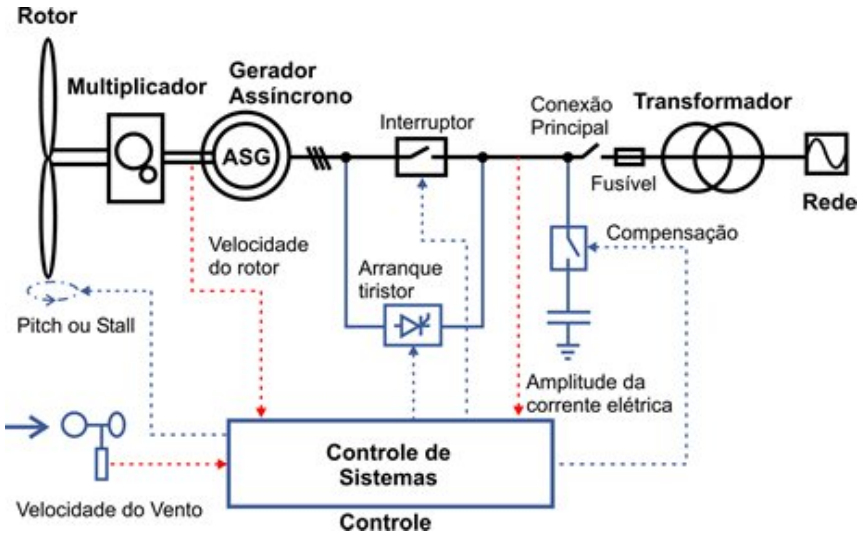
#### 3.3.5.4 Transmissão e Caixa de Velocidade

A transmissão, que engloba a caixa de velocidades, tem a finalidade de transmitir ao gerador a energia mecânica entregue ao eixo pelo rotor. É composta por elementos mecânicos, nomeadamente, engrenagens, acoplamentos, etc. Outra função é adaptar a baixa velocidade do rotor, geralmente entre 20 a 150 rpm, à velocidade do gerador, que se situa nas 1200 a 1800 rpm. Contudo, existem turbinas eólicas onde a caixa de velocidades foi suprimida, sendo tal facto acompanhado pela alteração dos geradores, utilizando-se, neste caso, geradores multi-pólos de baixa velocidade.

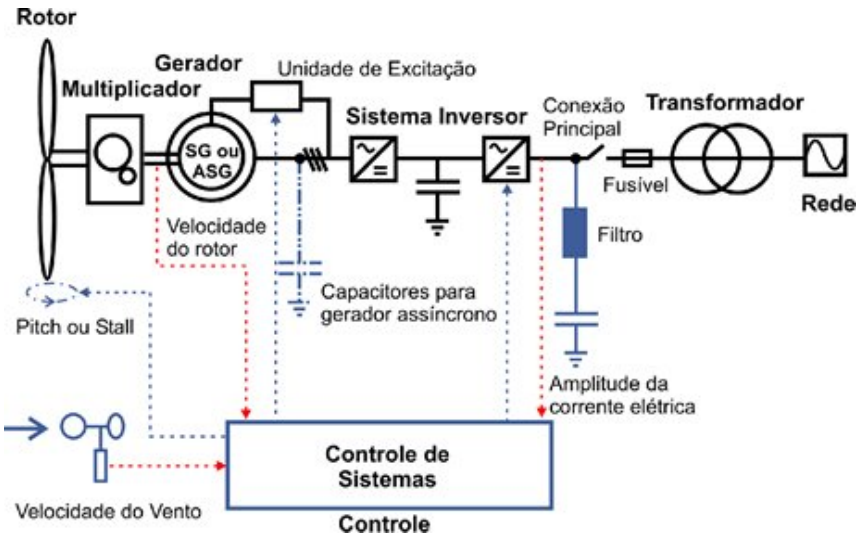
#### 3.3.5.5 O gerador

A conversão de energia mecânica em eléctrica é efectuada através de um gerador síncrono (alternador) ou de um gerador assíncrono (de indução).

Se a velocidade do veio no gerador se mantiver constante (através do controlo adequado da caixa de velocidades), o gerador de indução é a opção mais viável e económica tendo, no entanto, trocas de energia reactiva com a rede e, desta forma, necessidade de correcção do factor de potência.




(a) Aero-gerador de velocidade constante;




(b) Aero-gerador de velocidade variável, com conversor de frequência;

Figura 3.16: Aero-geradores de velocidade constante e variável [14].

A Fig. 3.16 mostra duas configurações possíveis. Na primeira, são utilizados geradores eléctricos assíncronos, ou de indução e, no segundo caso, utilizados geradores síncronos ou assíncronos.

#### 3.3.5.6 Sistemas de armazenamento da energia

Um dos aspectos negativos da energia do vento é que nem sempre está disponível quando é necessária e, por vezes, quando está disponível, não é necessária. Esta variabilidade do vento leva, por vezes, à necessidade de implementar sistemas de armazenamento. Nos sistemas interligados à rede eléctrica, a energia é injectada na rede e a adequação entre a produção e a procura de energia eléctrica é feita à custa das centrais convencionais, nomeadamente, centrais hídricas com bombagem. Assim, a energia eléctrica eólica, de uma forma geral, é sempre aproveitada. No caso dos ventos nocturnos, altura em que o perfil de consumo é inferior, por vezes a energia produzida é utilizada para bombear água para barragens, aumentando assim o potencial energético armazenado (isto acontece, genericamente, com a energia proveniente de outras fontes, situação que existe quando é mais caro desligar a produção de energia do que mantê-la em funcionamento em alturas que não seria necessária).

No caso de a energia eléctrica eólica ser utilizada como a principal fonte de energia, é necessário fazer o seu armazenamento de forma a adequar o perfil aleatório de produção com o perfil de consumo. As formas mais utilizadas de armazenamento são, baterias, ar comprimido e energia potencial.

#### 3.3.5.7 Acessórios

Nos acessórios englobam-se todos os componentes necessários ao bom funcionamento da turbina eólica. Incluem-se aqui, cabos, eixos, acoplamentos mecânicos, parafusos, rodas dentadas, rolamentos, etc.

### 3.3.6 Aplicações e aproveitamento do sistema eólico

A energia proveniente das turbinas eólicas pode ser utilizada em três aplicações distintas: sistemas isolados, sistemas híbridos e sistemas interligados à rede. Conforme o destino da utilização existem configurações básicas, havendo, no entanto, muitas partes comuns aos três tipos de utilização. As principais diferenças residem na potência e na necessidade ou não de uma unidade de armazenamento de energia:

- **Sistemas isolados** - Os sistemas isolados, normalmente, têm uma capacidade de produção baixa, sendo instalados em locais onde não existe energia eléctrica, usualmente, para aproveitamento doméstico. Regra geral, necessitam de uma unidade de armazenamento, excepto os sistemas utilizados para irrigação, moagem, etc. O armazenamento é, normalmente, feito em baterias ou, na forma de energia potencial, através da reserva em depósitos elevados. São também descritos na literatura sistemas de armazenamento por ar comprimido. O armazenamento em baterias necessita de conversores e sistema de controlo de carga e descarga das baterias de forma a aumentar a longevidade destas. Quando a descarga é efectuada para alimentar uma pequena instalação eléctrica, é necessária a utilização de inversores, quer sejam de estado sólido (eléctricos) ou rotativos (mecânicos);

- **Sistemas híbridos** - Os sistemas híbridos apresentam como fonte de energia várias hipóteses, tais como, turbinas eólicas, geradores a diesel, entre outros. A complexidade do sistema está relacionada com as fontes de energia, sendo necessário um controlo de todas as fontes para maximizar a eficiência do sistema. Os sistemas híbridos são usados para fornecer energia a um grande conjunto de consumidores que, normalmente, trabalham em corrente alternada;

- **Sistemas interligados à rede** - Os sistemas interligados à rede não necessitam de armazenamento de energia e são, normalmente, de grande porte, sendo, por vezes, constituídos por inúmeras turbinas eólicas. Estes sistemas maximizam a redução de perdas, no entanto, têm o problema de nem sempre o regime de ventos coincidir com o pedido de potência por parte da rede para satisfazer os consumos dos utilizadores.

### 3.3.7 Estudo teórico do aero-gerador

A construção dos aero-geradores beneficiou, inequivocamente, de todo o desenvolvimento teórico da aviação [89], nomeadamente, na construção das pás do rotor das turbinas eólicas. Neste ponto será abordado apenas o caso do aero-gerador mais utilizado actualmente, o de eixo horizontal com rotor de três pás.

#### 3.3.7.1 Potência disponível no vento

Para calcular a máxima potência passível de ser extraída da energia cinética do vento por uma turbina eólica, recorre-se à teoria da mecânica dos fluidos.

Devido à sua constituição, o ar pode ser considerado como um gás perfeito e, do ponto de vista matemático, segundo a lei dos gases perfeitos, a variação das suas propriedades é constante, isto é,

$$constante = P.\frac{Vol}{T} = P_0.\frac{Vol_0}{T_0} = P_i.\frac{Vol_i}{T_i} \tag{3.1}$$

onde $P$ é a pressão em Pa(*Pascal* [2]), $Vol$ é o volume em $m^3$ e $T$ a temperatura em *Kelvin* [3].

Segundo as leis da física, a energia cinética de qualquer corpo com massa $m$ ($k_g$) e velocidade $v$ (m/s) é dada por:

$$E_{cin} = \frac{1}{2}.m.v^2 (J = Joule) \tag{3.2}$$

Definindo densidade do ar $\rho$ [4], como a massa por unidade de volume ($k_g/m^3$), da mecânica de fluidos, pela lei da continuidade [5], temos que a massa do fluído mantém-se constante ao longo de um condutor, mesmo quando este tem secções diferentes (caso de mangueiras, ou, no caso do ar

[2] $Pascal = N.m^{-2}$

[3] $K = 273 + C^o$

[4] $\rho = 1,225 k_g/m^3$ em condições de pressão e temperatura normais

[5] Na mecânica de fluídos, o princípio da conservação da massa é traduzido pela equação de continuidade, ou seja, conservação de caudal. Se o fluído é contínuo e inviscido, então a massa que passa através da área A1 tem de ser igual à que passa na área A2 ($m^2$)

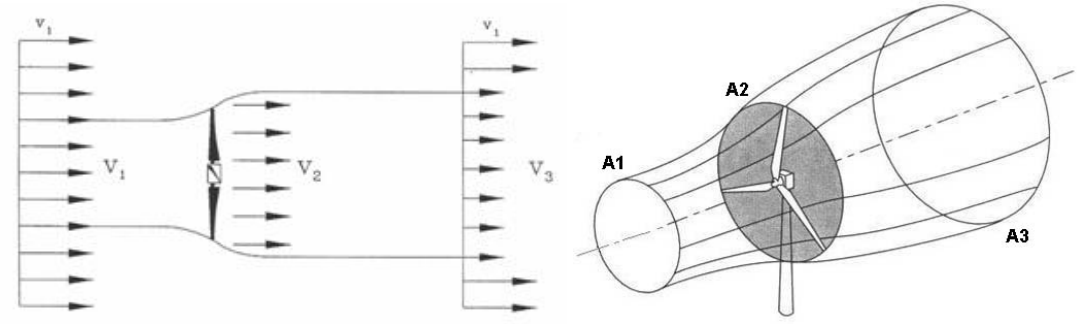



Figura 3.17: Relação entre a Velocidade do vento ao longo de diferentes secções da turbina [14].

ao passar numa turbina, Fig. 3.17):

$$\dot{m} = \rho_1 . A_1 . v_1 = \rho_2 . A_2 . v_2 \tag{3.3}$$

A eq. 3.3 define um caudal, massa por unidade de tempo (Kg/s), que será utilizada na eq. 3.2, para obter a potência [6] existente no vento (considerando velocidade constante, numa secção espacial específica da Fig. 3.17):

$$P = \frac{dE_{cin}}{dt} = \frac{1}{2} . \dot{m} . v^2 = \frac{1}{2} . (\rho . A . v) . v^2 = \frac{1}{2} . \rho . A . v^3 (W) \tag{3.4}$$

#### 3.3.7.2 Máxima potência extraível

Para calcular a energia extraível por uma turbina eólica, utiliza-se a diferença entre a energia cinética nas duas secções $A_1$ e $A_3$, respectivamente, antes e depois do vento passar na turbina, Fig. 3.17. Tipicamente $v_3 < v_1$, considerando-se que nas secções a velocidade é constante[7], isto é,

$$E_{rotor} = E_{área_1} - E_{área_3} = \frac{1}{2} . m . \left( v_1^2 - v_3^2 \right) \tag{3.5}$$

e, portanto, a potência extraída pela turbina eólica será:

$$P_{rotor} = \frac{1}{2} . \dot{m} . \left( v_1^2 - v_3^2 \right) \tag{3.6}$$

O fluxo de massa massa de ar no rotor é dado por $\dot{m} = \rho . A_2 . v_2$ (ver nota de rodapé), pela Fig. 3.17.

Segundo o teorema de Rankine-Froude a relação entre as velocidades $v_1$, $v_2$, $v_3$, é dada por $v_2 = \frac{v_1 + v_3}{2}$. Substituindo estas duas últimas expressões na eq. 3.6, vem:

$$P_{rotor} = \frac{1}{2} . \rho . A_2 . \frac{v_1 + v_3}{2} \left( v_1^2 - v_3^2 \right) = \frac{1}{2} . \rho . A_2 . v_1^3 . \frac{1}{2} . \left( 1 - \left( \frac{v_3}{v_1} \right)^2 \right) . \left( 1 + \frac{v_3}{v_1} \right) \tag{3.7}$$

[6] $W = J.s^{-1} = N.m.s^{-1} = K_g.m^2.s^{-3}$

[7] Na prática não acontece assim, pois no ponto mais alto a velocidade do vento é diferente do ponto mais baixo, em qualquer uma das três secções $A_1, A_2, A_3$

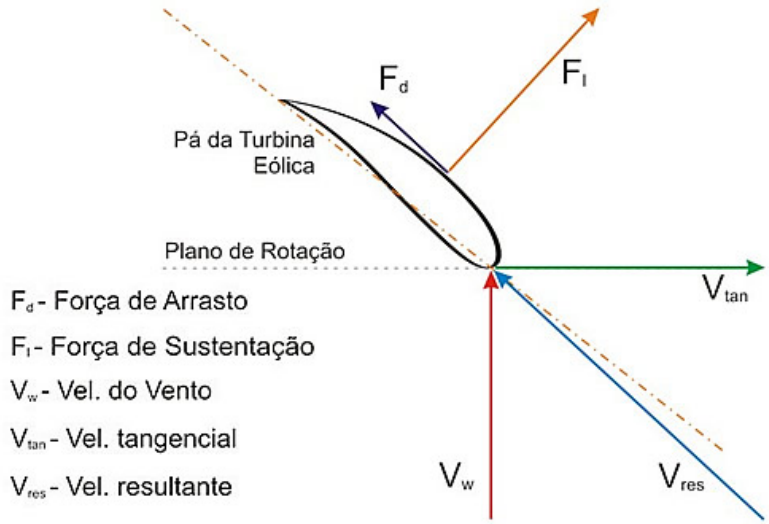



Figura 3.18: Forças actuantes numa pá [14].

onde se consideram as zonas $A_1$ e $A_3$ suficientemente afastadas para a pressão ter voltado ao valor normal e, portanto, iguais. De notar que, para compararmos a energia extraída pelo rotor e a energia do vento relativo à velocidade $v_1$, considera-se a aproximação $A_2 \approx A_1$. Ou seja, define-se $C_p$ como coeficiente aerodinâmico do rotor por:

$$C_p = \frac{P_{rotor}}{P_{vento}} = \frac{1}{2}.\left(1-\left(\frac{v_3}{v_1}\right)^2\right).\left(1+\frac{v_3}{v_1}\right) \tag{3.8}$$

que representa a percentagem de energia do vento convertida pela turbina em energia mecânica, e cujo valor máximo teórico é conhecido como limite de *Betz* ($16/27 = 0,593$). No entanto, salienta-se que este coeficiente foi calculado utilizando várias aproximações, nomeadamente, incompressibilidade do vento [8] ($\rho_1 \approx \rho_2 \approx \rho_3$) e áreas aproximadas entre as secções $A_1$ e $A_2$ da Fig. 3.17.

#### 3.3.7.3 Aero-dinâmica das pás

O estudo aerodinâmico das asas dos aviões é utilizado no projecto das pás do rotor dos aerogeradores; a Fig. 3.18 mostra as duas forças resultantes da acção do vento - força de sustentação $F_l$ e força de arrastamento $F_d$. A diferença nos aerogeradores, relativamente aos aviões, reside no facto de esta força de sustentação (responsável pela rotação do rotor), usualmente, ser tangencial. Outros condicionalismos advêm ainda de o movimento do rotor, a grande velocidade, poder ser considerado como um obstáculo à passagem do ar, tendo como consequência que a velocidade angular do rotor não pode aumentar indefinidamente.

[8] Aceitável para velocidades do vento muito inferiores à velocidade do som - $\rho_1 \approx \rho_2 \approx \rho_3$

#### 3.3.7.4 Aspectos complementares dos aero-geradores

Outra questão relevante é o efeito de esteira, devido à rotação das pás do rotor e à diminuição da energia cinética, que aumenta a turbulência a jusante do rotor. Este facto obriga, num parque eólico, a separar geograficamente cada aerogerador através de distâncias mínimas recomendadas, evitando a influência deste efeito e aumentando a vida útil dos equipamentos.

Para o controlo da velocidade do rotor existem duas técnicas distintas, uma consiste em alterar o ângulo de ataque da pá do rotor [9] relativamente ao vento e, a outra, através das perdas aerodinâmicas quando a velocidade do vento ultrapassa determinada velocidade [10]. O primeiro sistema é mais caro e complexo de construir, estando, portanto, mais sujeito a avarias, no entanto, permite maiores ganhos de eficiência. O segundo, mais barato e mais simples, torna-se menos eficiente sendo, actualmente, pouco utilizado. Uma diferença fundamental entre os dois tipos de turbinas é no arranque e na paragem. A turbina *pitch* auxilia nas duas situações, enquanto a turbina *stall* não tem binário para o arranque. No entanto, em termos práticos, as turbinas *pitch* também são auxiliadas no arranque de modo a diminuir o tempo dessa fase.

É interessante recalcular o coeficiente de conversão aerodinâmico da eq. 3.8, em função do ângulo da pá[11], da velocidade angular de rotação do rotor e a velocidade linear do vento, na extremidade das pás [2]:

$$C_p(\lambda,\beta) = 0,22\left(\frac{116}{\lambda_i} - 0.4\beta - 5\right) exp\left(-\frac{12,5}{\lambda_i}\right)$$

$$\lambda_i = \frac{1}{\frac{1}{\lambda+0,08\beta} - \frac{0,035}{\beta^3+1}} \tag{3.9}$$

$$\lambda = \frac{\omega_t . r_{rotor}}{v}$$

Verifica-se através da eq. 3.9, a necessidade de adaptar a velocidade de rotação do rotor em função da velocidade do vento, de modo a manter constante o factor $\lambda$ [12] no ponto óptimo para maximizar $C_p$. Na realidade, muitos dos aerogeradores fazem esta adaptação através do aumento da velocidade do rotor através da energia da rede - geradores eólicos de velocidade variável. Contrariamente, os geradores de velocidade constante, como não se adaptam à variação de velocidade do vento para manter $C_p$ no valor máximo, funcionam em condições não óptimas.

## 3.4 Turbinas submersas

A empresa *Marine Current Turbines* [15] começou a instalar em 2008 aquele que está a ser considerado pelos especialistas como o mais viável de todos os projectos para gerar energia eléc-

[9] Sistema *Pitch*

[10] Na aero-dinâmica dos aviões denominado de perda por *stall*, acontece quando o ângulo do *flap* da asa ultrapassa os 16 graus.

[11] Ângulo de *Pitch*, $\beta$

[12] Conhecido por *razão de velocidades na pá* ou *velocidade específica na ponta da pá* [2].



Figura 3.19: Construção de uma turbina submersa com rotor de duas pás [15].

trica a partir do movimento das marés (Fig. 3.19). Existem diversos projectos diferentes com o objectivo de explorar a energia das ondas e das marés para a geração de electricidade. Entre eles, há sistemas baseados em músculos artificiais, longas bóias flutuantes e bóias submersas, apenas para citar alguns que chegaram à fase de projectos-piloto.

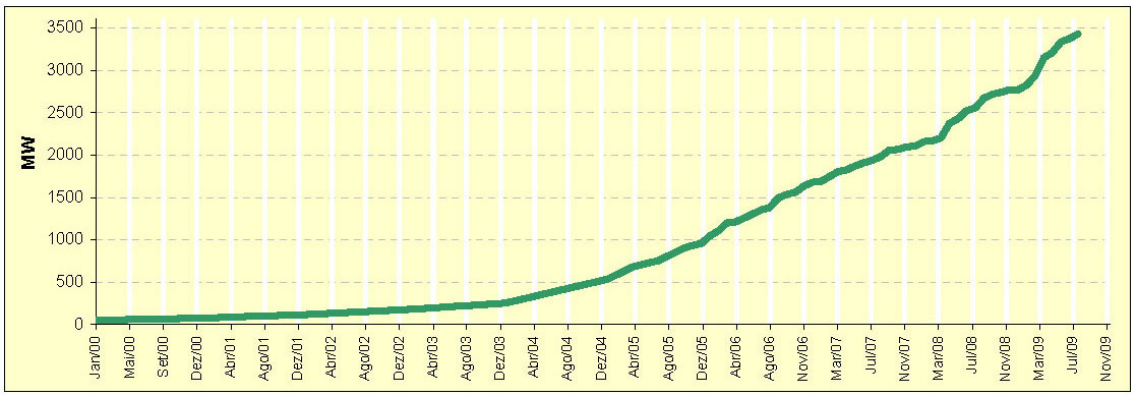
O novo projecto utiliza uma turbina submersa parecida com as turbinas eólicas e tem o mesmo princípio de funcionamento, com a diferença de que, ao invés do vento, é o movimento da água que faz girar as hélices.

Cada hélice mede entre 15 a 20 metros de comprimento, sendo instaladas num eixo que é fixo no fundo do mar por um sistema de auto-fixação, o que diminui os custos de instalação. As primeiras turbinas serão instaladas na costa da Irlanda do Norte. Segundo a empresa, o movimento das marés nessa região deverá fazer com que as turbinas rodem com uma velocidade entre 10 e 20 rotações por minuto, o que seria insuficiente para oferecer riscos para os animais marinhos. Cada turbina é capaz de gerar 1,2 MW de energia. No ano passado, a Inglaterra inaugurou uma infra-estrutura básica para viabilizar a inserção destes parques marítimos de geração de electricidade na rede de distribuição eléctrica do país. Tudo leva a pensar que o projecto, no futuro, poderá ser ainda mais ousado, incluindo na parte submersa uma turbina para aproveitamento da energia do mar e na parte superior, uma para aproveitamento eólico (dependendo do potencial eólico da localização geográfica).

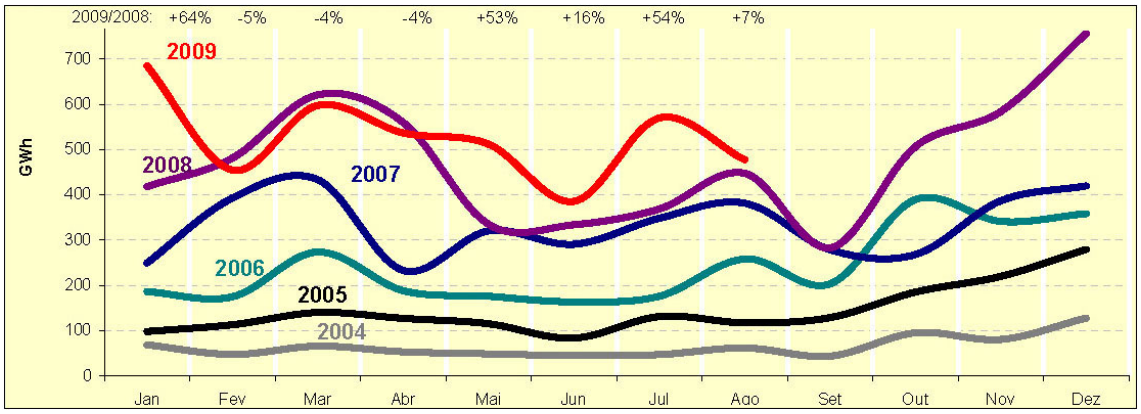
## 3.5 Potência eólica instalada no mundo e em Portugal

A energia eólica é hoje uma realidade e representa para as nações uma fonte de riqueza, dado que permite poupar outros recursos.

De acordo com o relatório da *Global Wind Energy Council* [90], existe um forte compromisso da União Europeia em incentivar a instalação da energia eólica, que actualmente tem mais de metade da potência mundial instalada. Na Europa dos 65946 MW instalados no final de 2008,




(a) Evolução da potência eólica instalada em Portugal Continental;




(b) Evolução mensal da produção eólica em Portugal Continental;

| | 2001 | 2002 | 2003 | 2004 | 2005 | 2006 | 2007 | 2008 | Ago 2009 | TCMA |
|---|---|---|---|---|---|---|---|---|---|---|
| **Potência instalada (MW)** | 114 | 175 | 253 | 537 | 1.047 | 1.681 | 2.108 | 2.770 | 3.430 | 57,7% |
| **Nº parques instalados** | 16 | 24 | 42 | 71 | 103 | 139 | 152 | 172 | 191 | 40,4% |
| **Nº de aerogeradores instalados** | 173 | 213 | 276 | 441 | 702 | 1.003 | 1.132 | 1.495 | 1.826 | 36,1% |
| **Potência média (MW)** | | | | | | | | | | |
| dos parques | 7,1 | 7,3 | 6,0 | 7,6 | 10,2 | 12,1 | 13,9 | 16,1 | 18,0 | 12,4% |
| dos aerogeradores instalados | 0,7 | 0,8 | 0,9 | 1,2 | 1,5 | 1,7 | 1,9 | 1,9 | 1,9 | 15,9% |
| dos aerogeradores instalados em cada ano | 0,5 | 1,5 | 1,2 | 1,7 | 2,0 | 2,1 | 3,3 | 1,8 | 2,0 | 20,3% |

(c) Caracterização da potência instalada em Portugal Continental;

| | Nº de parques | | | | | Potência (MW) | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| MW | 2006 | 2007 | 2008 | Ago 2009 | % 2009 | 2006 | 2007 | 2008 | Ago 2009 | % 2009 |
| **Total** | 139 | 152 | 172 | 191 | 100% | 1.681 | 2.108 | 2.770 | 3.430 | 100% |
| **≤1** | 23 | 23 | 23 | 23 | 13% | 16 | 16 | 16 | 16 | 0% |
| **>1 ≤10** | 63 | 67 | 73 | 73 | 42% | 296 | 327 | 359 | 344 | 10% |
| **>10 ≤25** | 40 | 43 | 48 | 63 | 28% | 599 | 634 | 743 | 960 | 28% |
| **>25 ≤50** | 8 | 12 | 19 | 19 | 11% | 264 | 446 | 599 | 649 | 19% |
| **>50** | 5 | 7 | 9 | 13 | 5% | 506 | 685 | 1.053 | 1.461 | 43% |

(d) Caracterização dos parques eólicos por intervalos, em Portugal Continental;

Figura 3.20: Panorama em Agosto de 2009 (Fotos DGEG, Renováveis - Estatísticas Rápidas).

cerca de 8877 MW foram instalados durante esse ano. A nível mundial, no final de 2008 estavam instalados cerca de 120798 MW, dos quais 27051 MW nesse ano.

Portugal tem cumprido metas ambiciosas e ocupa o $10^o$ lugar a nível mundial no *ranking* da potência instalada. Segundo os relatórios estatísticos da DGEG[13], em Agosto de 2009 havia 3430 MW de potência eólica instalada. A Fig. 3.20 apresenta o panorama em Portugal Continental em Agosto de 2009.

## 3.6 Conclusões

Neste capítulo abordaram-se os principais aspectos relacionados com o aproveitamento eólico para geração de energia, dando particular importância a questões históricas e técnicas.

[13] http://www.dgge.pt/

# Capítulo 4

# Organização e gestão da manutenção de parques eólicos

A organização e gestão da manutenção tem uma relevância crucial para manter as instalações e os equipamentos em condições adequadas de operacionalidade e fiabilidade, bem como para a determinação de rácios técnicos e financeiros, para um acompanhamento e melhoria continua da actividade de manutenção. Este capítulo introduz os procedimentos do ponto de vista daquela organização e gestão, desde a classificação e gestão dos objectos de manutenção, bem como da estratégia e implementação do controlo e planeamento da manutenção. Uma parte significativa destes objectivos é concretizado através de um sistema de informação para gestão de manutenção, designado por SMIT (Sistema Modular Integrado de Terologia) [68], que foi actualizado e melhorado, em consequência dos trabalhos desta dissertação, tal como descrito no anexo A.

Normalmente, numa unidade de produção, existem diversos tipos de manutenção, que coexistem em simultâneo. O modo predominante deve ser a manutenção planeada, a qual coexistirá sempre com a não planeada.

De acordo com o descrito na secção 2.2 (pág. 16) existe uma série de procedimentos periódicos que devem ser realizados para manter o bom funcionamento dos aerogeradores. Para organizar esta tarefa são codificados os componentes principais deste tipo de equipamento.

## 4.1 Manutenção

Todos os equipamentos sofrem desgaste, quer em utilização normal quer intensiva. Para o desempenho se manter dentro de um patamar de rendimento adequado, é necessário que as suas condições de funcionamento sejam as mais apropriadas. Este aspecto significa que, ao longo do seu ciclo de vida, os equipamentos necessitam de intervenções para colocar as suas condições de funcionamento dentro dos parâmetros adequados. A denominação "intervenções" significa, acções planeadas, não planeadas, reparações, inspecções e todas as operações desta natureza. Todas

estas acções fazem parte da actividade de manutenção, que corresponde a um conjunto de acções necessárias a garantir o bom desempenho dos equipamentos em funcionamento. Quando bem gerida, a manutenção é um factor relevante na economia de uma empresa, capaz de alterar índices de produtividade, níveis de qualidade de processos, diminuição de produtos defeituosos, e aumento da confiança dos clientes. Devido à evolução da sociedade, surgem novos desafios e novos obstáculos, tais como o custo, a escassez das energias não renováveis e a preservação do meio ambiente, entre outros. Todos estes factores alavancam a manutenção para um novo patamar - eficácia económica. A manutenção, logicamente, tem associados custos de peças-de-reserva, de mão-de-obra, de contratos com terceiros, de deslocações, de utilização de ferramentas, etc. Esta gestão financeira é preponderante na eficácia de uma organização, quer na escolha da altura certa da intervenção (nem demasiado cedo, nem demasiado tarde) quer na análise dos gastos desnecessários.

Associado à manutenção têm-se tempo de intervenção, tempo de paragem do equipamento por avaria - tempo de indisponibilidade - custos das reparações discriminados por custos internos, externos, entre muitos outros dados. Evidentemente, uma boa manutenção ajuda a minimizar as avarias, prevenindo imobilizações e aumentando o tempo de disponibilidade dos equipamentos.

Por outro lado, a manutenção ajuda a minimizar avarias e imobilizações do equipamento, aumentando o tempo de disponibilidade. A poupança oriunda de uma ausência de manutenção é facilmente superada pela ineficiência do desempenho dos equipamentos, nomeadamente:

- Por uma produtividade imperfeita, devido à deficiente prestação dos equipamentos;
- Por existirem paragens mais frequentes;
- Pelo aumento dos defeitos dos produtos e, por consequência, pela consequente perda de imagem junto dos clientes.

A manutenção, normalmente, é dividida em vários tipos, de acordo com a natureza da intervenção:

- **Manutenção não planeada (correctiva, curativa)** - Trata-se da intervenção após a avaria ocorrer. Este género de manutenção ocorre devido à falta de planeamento e, também, porque nem tudo é previsível a 100%. Idealmente, deve ocorrer só em equipamentos de pouca importância na produção;
- **Manutenção planeada sistemática** - Efectuada de forma periódica, incluindo, normalmente, inspecção e elaboração de trabalhos pré-estabelecidos. A periodicidade pode ser baseada num ou em vários parâmetros de funcionamento, tais como, horas de trabalho, número de peças produzidas, número de quilómetros percorridos, entre outros;
- **Manutenção planeada condicionada** - Efectuada em função do estado do equipamento, sendo a intervenção realizada caso determinado parâmetro atinja um valor limite previamente definido. Incorpora os seguintes passos:

  - Identificação do tipo de variável de controlo - Através de variáveis de funcionamento do equipamento que traduzem a sua condição (temperatura, vibração, estado do óleo, potência produzida, etc), e que permitem determinar, com antecedência, a possibilidade de uma avaria futura;
  - Determinação das consequências - Identificar o que pode acontecer se não houver intervenção de manutenção e determinação de eventuais falhas futuras por análise do comportamento das curvas das variáveis em monitorização.

Todos os equipamentos passam pelas seguintes fases: desenho, fabricação, instalação e montagem, utilização e manutenção, fim de vida. O grau de desgaste aceitável poderá não ser muito elevado, até aos extremos, mas sim o tempo razoável e necessário para que isso aconteça. Portanto, a altura certa para fazer a manutenção não será quando o equipamento ou bem estiver perto de uma falha ou paragem mas sim quando for detectado uma possível avaria futura dentro de patamares admissíveis. Assim, as principais acções no decorrer de uma inspecção de manutenção podem-se resumir no seguinte:

- Avaliar o estado do objecto de manutenção;
- Verificar quais as razões do seu desgaste;
- Indicar as acções para corrigir os problemas;
- Perceber se o problema é casual ou recorrente, para determinar condições futuras de repetibilidade da anomalia;
- Avaliar as leituras efectuadas sobre parâmetros indicativos que sejam possíveis e ou necessários para vigiar o estado de desempenho dos equipamentos.

As inspecções são importantes, como se verá no caso dos aerogeradores, mas, no geral, fazer uma boa inspecção passa por verificar alguns dos seguintes itens:

- Ruídos subjacentes a danos estruturais;
- Inspecção visual ou através de equipamento de visão, termográfico, binóculos, etc, para determinação de danos estruturais;
- Apertos dinamométricos;
- Temperaturas de funcionamento anormais, de óleos ou de equipamentos;
- Viscosidade e outras variáveis de controlo de fluidos responsáveis pela lubrificação;
- Limpeza do equipamento;
- Fugas de fluídos;
- Ruídos de funcionamento anormais;

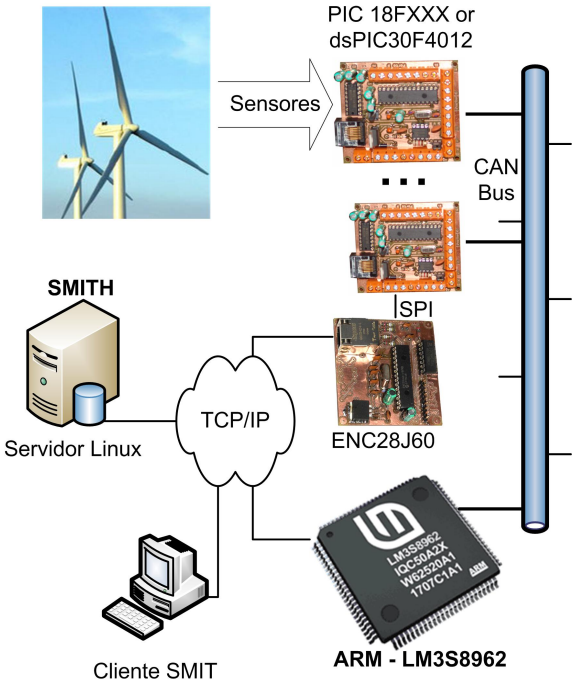



Figura 4.1: Sistema de Manutenção Integrado proposto.

- etc, ....

No decorrer destas inspecções pode ser desencadeada uma acção de manutenção correctiva ou agendada uma acção de manutenção planeada para uma futura oportunidade.

A manutenção planeada condicionada utiliza instrumentos de medida *on-line* ou não, para determinação do nível de deterioração dos parâmetros de controlo. Na prática, utilizando um eufemismo ligado ao sector da saúde, este género de manutenção baseia-se na determinação do "estado de saúde" do equipamento, através da medição dos "sinais vitais" do equipamento. Tal similaridade não é inoportuna, até porque existem sistemas de determinação de causas de doenças que funcionam de forma semelhante aos sistemas de detecção de causas de avarias em equipamentos. A grande diferença é a semântica utilizada.

Resumindo, na perspectiva duma gestão de manutenção eficaz e económica, desenvolveu-se, no desenrolar do presente trabalho, uma metodologia de um sistema de aquisição de dados de baixo custo, direccionada para a manutenção condicionada.

## 4.2 Sistema de manutenção integrado

A organização e gestão da manutenção de parques eólicos para ser levada a efeito, necessita de um *software* de gestão da manutenção e de um sistema de monitorização dos valores de alguns parâmetros chave dos equipamentos.

A Fig. 4.1 apresenta o modelo base, destacando-se os seguintes aspectos da composição do sistema:

- **Servidor de base de dados do SMIT** - Caracteriza-se por funcionar em sistemas *Unix*, podendo ser utilizado o *Linux*, *FreeBSD* ou outros sistemas compatíveis. Incorpora o seguinte *software* livre, ou funcionalidades:
  - *PostgreSQL* como motor de base de dados, onde estão definidas as tabelas do SMIT. O sistema foi desenhado com ponderação, tendo sido todas as funções de relacionamento e validação de dados incorporadas ao nível da linguagem procedimental do *PostgreSQL*. Agrega cerca de 140 tabelas e 150 funções de base de dados;
  - Servidor de páginas com linguagem *PHP* incorporada - *Apache+PHP* - incluí a implementação de gestão de ficheiros de manuais, multimédia, etc, associados a equipamentos, ferramentas, ordens de trabalho, etc, carregados e acessíveis através de *Browser*;
  - *Software* de análise numérica - *Octave* e *R* - integrado com a base de dados do SMIT, através de acesso directo às tabelas. A inclusão destas ferramentas faculta a definição de funções de análise de dados em linguagem numérica, suportando várias primitivas, tais como análise espectral, análise de séries temporais, etc;
  - Gestão de automatismos, tais como envio de emails automáticos, criação de ordens de trabalho automaticamente, sistema de *backup* de dados, etc;
  - *WebServices* ou ligações *odbc* - permite a ligação à base de dados através de ligações estruturadas, embora sejam utilizadas em último recurso. Relativamente aos *WebServices*, implementados em *PHP*, trata-se de uma tecnologia que permite isolar o nível de implementação de terceiros, sem necessidade de permitir acesso ao núcleo do sistema;
- **Clientes SMIT para o ambiente *Windows*** - Através deste sistema, os operadores (administração, gestor da manutenção, chefe de departamento, aprovisionamento, técnico de manutenção, chefe da qualidade, chefe da produção, OM em projecto e utilizadores em geral) interagem com o sistema registando a informação de manutenção. O sistema foi desenvolvido modularmente;
- **Módulo mestre de gestão** - Este módulo de *hardware* permite gerir todo o fluxo de dados entre os módulos escravos de campo e o sistema central SMIT. A diferença destes módulos, relativamente a sistemas comerciais, é a sua capacidade de sincronização de relógio pela rede IP. A sua função é ler a configuração do SMIT e controlar os módulos escravos, "obrigando-os" a fazer a aquisição dos dados de forma sincronizada com um horário pré-definido, que é partilhado por todos os módulos instalados desta categoria. Comunica com o SMIT via IP e com os módulos escravos via *CAN*. Na sequência dos trabalhos, três soluções foram caracterizadas com diferentes níveis de desempenho:
  - Módulo baseado na *DStini400* - Programação em *Java* e em C (*Keil C*). É um modulo importante, no entanto, nos testes de sincronização, revelou-se demasiado lento.

Esta situação adveio dos programas estarem escritos em *Java* como consequência do compilador de C não estar disponível por questões financeiras. Permitiu, no entanto, validar o conceito de uma forma rápida em termos de desenvolvimento. O protocolo de sincronização utilizado foi o SNTP;

– Módulo baseado em PIC18F2685 + ENC28J60 - Programação em C. Utiliza os microcontroladores da *Microchip* para construir o módulo integrando os meios de comunicação - *CAN* a 1Mbps e *Ethernet* a 10Mbps. Neste módulo implementou-se o *stack* TCP/IP em conjunto com as funções do protocolo *CAN*. A sincronização está disponível a dois níveis protocolares - SNTP e PTP (parcialmente);

– Módulo baseado no ARM LM-3S8962 - Programação em C. Incorpora meios de comunicação *CAN* a 1Mbps e *Ethernet* a 100Mbps.A grande diferença deste microcontrolador, relativamente aos anteriores, provém de um módulo de *timestamp* incorporado no *hardware* de *Ethernet*, cuja função é registar a hora de chegada e saída dos pacotes *Ethernet*. Este aspecto é fundamental para realizar sincronizações de relógio com diferenças na casa dos nanossegundos. Esta propriedade permite utilizar o protocolo de sincronização PTP de forma mais eficaz, ao anular a incerteza temporal entre a recepção do pacote *Ethernet* e o seu processamento pelo *firmware*;

- **Módulo escravo de aquisição** - Recebe os sinais de aquisição pelo barramento *CAN* e executa a aquisição temporizada de acordo com esses sinais, reencaminhando os dados pela rede *CAN* a 1Mbps. Este módulo, em concordância com as necessidades de velocidade de aquisição, utiliza dois microcontroladores, um de baixo desempenho (18F2685, 10MFlops) e outro de alto desempenho (dsPIC30F4012, 30MFlops).

Em termos de inovação, o SMIT, comparativamente a outros *softwares* comerciais, é de fácil utilização e compreensão. É inovador na representação matricial de inter-ligações entre os objectos de manutenção. Até à data, todas as relações entre objectos de manutenção eram representadas exclusivamente em estrutura em árvore. Com o conceito matricial, todos os objectos podem ser interligados entre si, o que é uma das concepções inovadoras introduzidas. É ainda inovadora a arquitectura utilizada para resolver o problema de dispor geograficamente vários módulos de aquisição e conseguir sincronizá-los através da rede IP. A noção associada ao protocolo PTP só agora despontou o interesse na indústria de automação, cuja relevância é importantíssima num controlo distribuído. Sintetizando, a inovação presente nesta dissertação, assume-se claramente no conjunto dos vários "blocos" que constituem o sistema, cuja integração se apresenta como uma mais-valia científica face ao estado da arte.

A Fig. 4.2 ilustra um esquema semelhante ao da Fig. 4.1 mas, agora, incorporando várias possibilidades de interligação com sistemas comerciais de aquisição de dados, como é o caso dos sistemas de instrumentação da *National Instruments*, que utilizam o *Labview* e de equipamentos da automação industrial, na figura autómatos *Beckhoff*. Esta é outra valência importante, a integração de ferramentas comerciais, na recolha de informação sensorial.

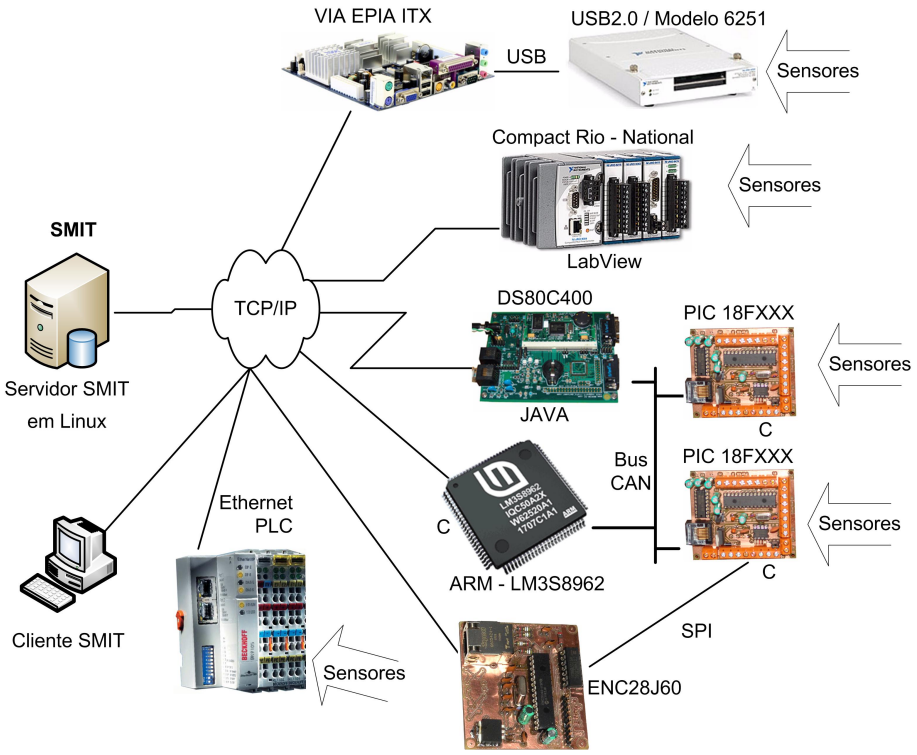



Figura 4.2: Interligação do SMIT com vários sistemas, incluindo os comerciais.

### 4.2.1 O *software* de gestão da Manutenção - SMIT

O SMIT (Fig. 4.3) é um programa cliente servidor, multi-base de dados, permitindo a instalação de vários clientes e a configuração de vários sistemas na mesma plataforma. O programa é acedido através do ambiente *Windows*. O SMIT permite a optimização dos recursos da manutenção através das seguintes ferramentas modulares:

- Caracterização dos Objectos de Manutenção;
- Gestão de Fornecedores (equipamentos, peças, serviços);
- Gestão de Recursos Humanos afectos à Manutenção;
- Gestão de Ferramentas e Utensílios Especiais;
- Gestão do Aprovisionamento e Stocks de Peças-de-Reserva;
- Diagnóstico de Avarias, Emissão de Relatórios, Análises e Planos de Acção de Melhoria;
- Emissão, Coordenação, Acompanhamento e Encerramento das Ordens de Trabalho;
- Gestão de Pedidos de Intervenção por parte dos utilizadores genéricos dos equipamentos;
- Gestão gráfica de Planos, OT, Pedidos de Intervenção;

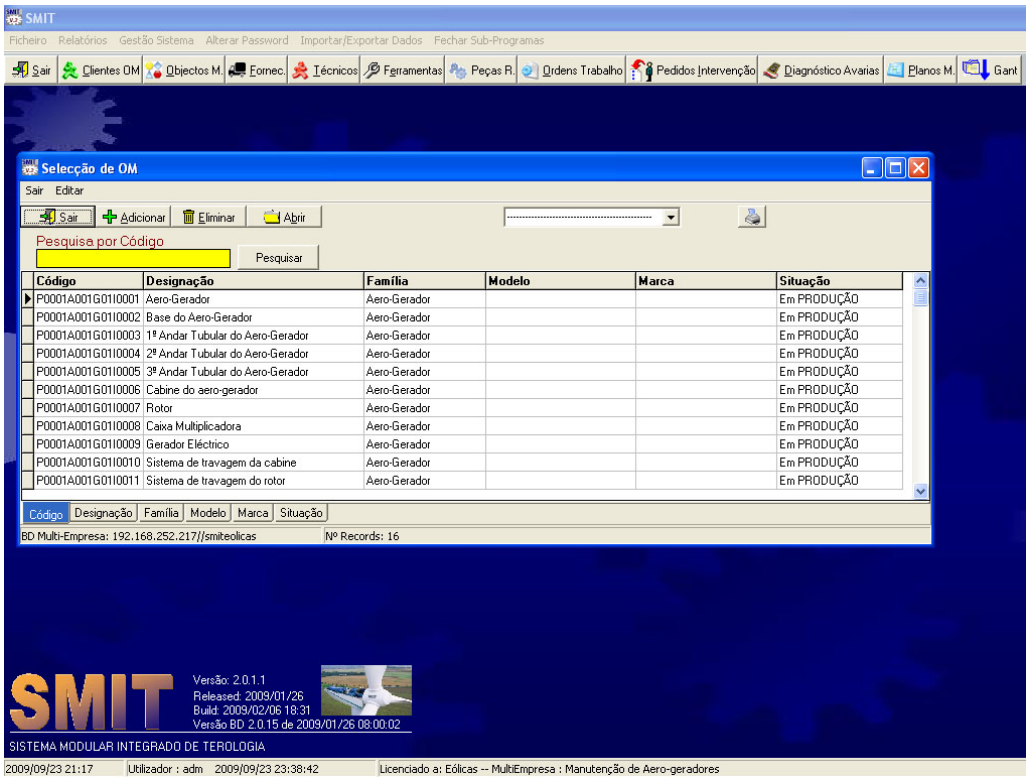



Figura 4.3: SMIT: Cliente, para ambiente *Windows*.

- Gestão da Manutenção Planeada (Planeada Sistemática e Condicionada);
- Diversos relatórios, nos quais se incluem relatórios de indicadores da manutenção, tais como MTBF (*Mean Time Between Failures*), MTTR (*Mean time to Repair*) e MTTS (*Mean Time To be Stopped*), para além de relatórios dos diversos custos;
- Exportação de dados para Excel;
- Relatórios flexíveis directamente para o Excel;
- Exportador/importador de dados através de perfis flexíveis de definição da estrutura de dados.

O SMIT é uma ferramenta de apoio à gestão de manutenção que apoia esta actividade a atingir melhores condições de competitividade, produtividade e eficiência, nomeadamente, através de:

- Redução do tempo de preparação e emissão de Ordens de Trabalho (OT);
- Emissão de OTs detalhadas com definição de recursos e tempos previstos para execução dos trabalhos;
- Estabelecimento de prioridades de actuação, perante uma carga de trabalhos diversa, em face da importância do equipamento, da urgência dos trabalhos e dos recursos disponíveis (humanos e técnicos);

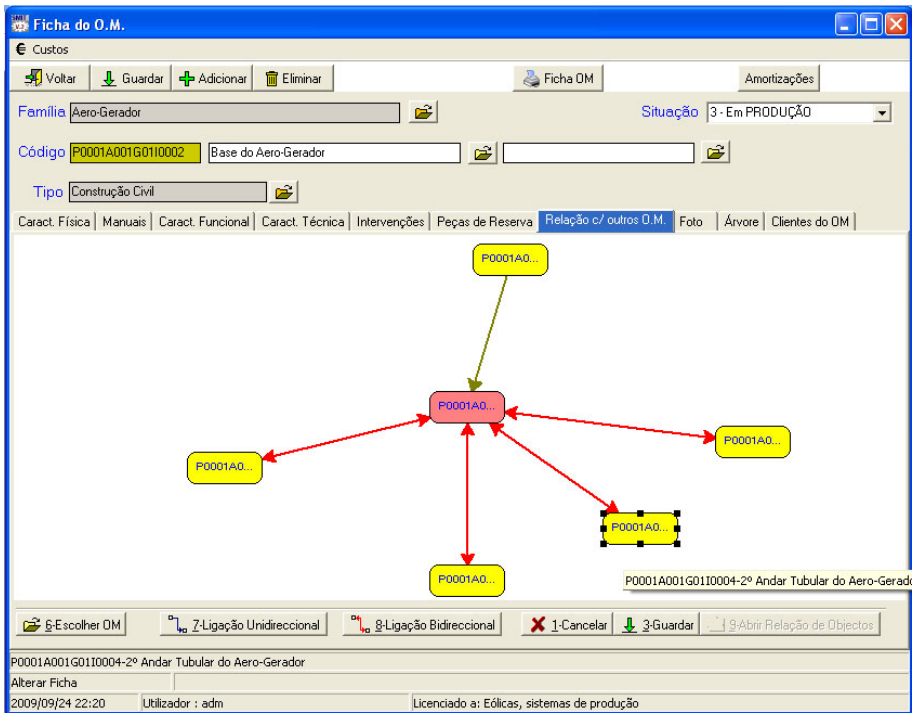



Figura 4.4: SMIT: Relações entre os diferentes objectos de Manutenção.

- Desenvolvimento de planos diários de intervenção e de efectuar a gestão de atrasos e trabalhos em carga, de modo a diminuir o tempo de resposta e o tempo de indisponibilidade dos objectos de manutenção;
- Planeamento da manutenção através da análise do histórico e respectiva análise de falhas e modos de avaria;
- Planeamento detalhado de inspecção preventiva de modo a cumprir todos os requisitos de operacionalidade dos objectos de manutenção;
- Gestão de peças-de-reserva, garantindo a existência das quantidades adequadas em função da criticidade dos equipamentos ou da urgência do trabalho e através da gestão dos respectivos fornecedores, para que seja possível efectuar o seu aprovisionamento em tempo útil;
- Emissão e gestão de encomendas e pedidos a fornecedores de peças-de-reserva, ferramentas, etc...
- Planeamento dinâmico da manutenção, actualizando automaticamente os planos de manutenção dos objectos de manutenção em função da evolução dos trabalhos realizados.

Um dos aspectos inovadores deste *software*, como referido anteriormente, depara-se na relação entre os vários Objectos de Manutenção, convencionalmente em árvore. No SMIT, os objectos

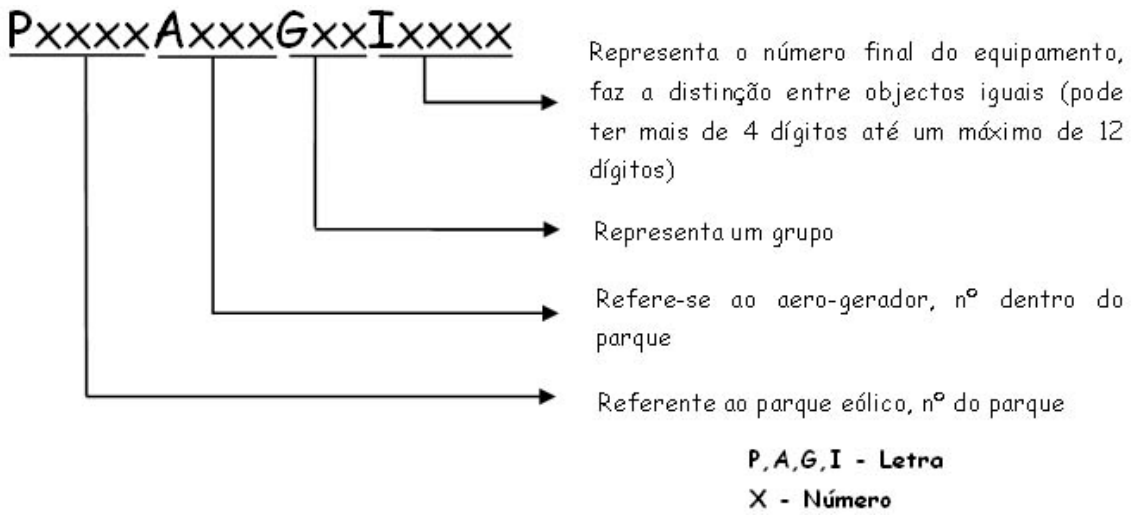



Figura 4.5: Constituição dos códigos dos equipamentos.

relacionam-se numa estrutura matricial (Fig. 4.4), permitindo, assim, estabelecer todo o género de relações existentes nos equipamentos e instalações reais. Esta possibilidade permite a obtenção dos mais diversos indicadores de forma ainda mais amigável.

Outro dos aspectos dignos de realce, é a possibilidade da calendarização da manutenção planeada ser passível de alteração pelo gestor da qualidade/produção, sobrepondo-se às datas aconselhadas pelo SMIT. As datas previstas para a intervenção podem assim ser alteradas interactivamente, através de interface gráfico, pelo responsável e, em simultâneo, há a possibilidade de emissão de um pedido de intervenção planeado - um conceito único do SMIT, que agrega pedidos de intervenção correctivos e planeados. Todas as datas previstas para as futuras OTś preventivas são calculadas em *background* com ou sem influência de dados externos, tais como, evolução dos parâmetros de planeamento e períodos de indisponibilidade do equipamento para o departamento de manutenção.

O anexo A apresenta mais detalhes técnicos relativos ao *software* de manutenção SMIT, desenvolvido na sequência desta dissertação.

## 4.3 Codificação dos objectos de Manutenção

Na gestão da manutenção, normalmente, codificam-se os equipamentos numa estrutura em árvore permitindo assim associar todos os seus elementos funcionais podendo incluir componentes. No SMIT, podem ainda existir relações bidireccionais, conforme descrito anteriormente. Outra característica dos OMs relaciona-se com os seus códigos, os quais podem ser constituídos por grupos e sub-grupos, podendo estes obedecer a alguma sistematização prévia ou serem códigos aleatórios. No caso presente, será fundamental incluir no código do Objecto de Manutenção (OM), a codificação do parque eólico, do aero-gerador, eventualmente do grupo de família a que pertence

| Código OM | Descrição | Tipo |
|---|---|---|
| P0001A001G01I0001 | Aero-Gerador | instalação |
| P0001A001G01I0002 | Base do Aero-Gerador | instalação |
| P0001A001G01I0003 | 1º Andar Tubular do Aero-Gerador | instalação |
| P0001A001G01I0004 | 2º Andar Tubular do Aero-Gerador | instalação |
| P0001A001G01I0005 | 3º Andar Tubular do Aero-Gerador | instalação |
| P0001A001G01I0006 | Cabine do aero-gerador | instalação |
| P0001A001G01I0007 | Rotor | equipamento |
| P0001A001G01I0008 | Caixa Multiplicadora | equipamento |
| P0001A001G01I0009 | Gerador Eléctrico | equipamento |
| P0001A001G01I0010 | Sistema de travagem da cabine | equipamento |
| P0001A001G01I0011 | Sistema de travagem do rotor | equipamento |
| P0001A001G01I0012 | Sistema de Fixação das pás do rotor | equipamento |
| P0001A001G01I0013 | Sistema de Aquecimento | equipamento |
| P0001A001G01I0014 | Controlador de *Pitch* | equipamento |
| P0001A001G01I0015 | Inversor electrónico | equipamento |
| P0001A001G01I0016 | Sensores de vento | equipamento |

Tabela 4.1: Exemplo de codificação.

e o item em concreto. Porquê a importância da codificação? Embora os OMs, conforme representado na Fig. 4.4, possam ser interligados entre si, a realidade para o operador torna-se mais fácil, através do código. Esta codificação é um dos pilares para uma boa organização da Gestão da Manutenção.

A Fig. 4.5 explica como foi projectada a codificação dos objectos associados a um parque eólico. Como no SMIT, todos os códigos podem ter 25 caracteres, optou-se por reservar cinco dígitos para referenciar o parque eólico, quatro para o aero-gerador do parque, três para o grupo (neste caso apenas será usado um grupo) e, finalmente, cinco para o equipamento/instalação do aero-gerador.

Com a estrutura de codificação descrita, construiu-se a Tab. 4.1, que constitui a base de trabalho da manutenção de aero-geradores, segundo a filosofia do presente trabalho.

## 4.4 Estratégia de planeamento das Ordens de Trabalho

Conforme comentado na secção 2.2 (pág. 16), existe uma série de acções de manutenção planeada sistemática que devem ser executadas rotineiramente, para garantir uma boa operacionalidade de um aerogerador. A estratégia, para ser inserida no *software* SMIT, utiliza os planos de manutenção que podem ser definidos com periodicidade dependente da evolução de um parâmetro ou, então, o próprio plano pode ser indexado a outro já existente. Este aspecto da indexação é importante, por obrigar todos os planos indexados entre si a originarem uma Ordem de Trabalhos na mesma data. Tal facto é preponderante, por exemplo, para fazer a manutenção das pás do rotor, o aerogerador necessita de estar não operacional. Há uma série de tarefas que podem, e

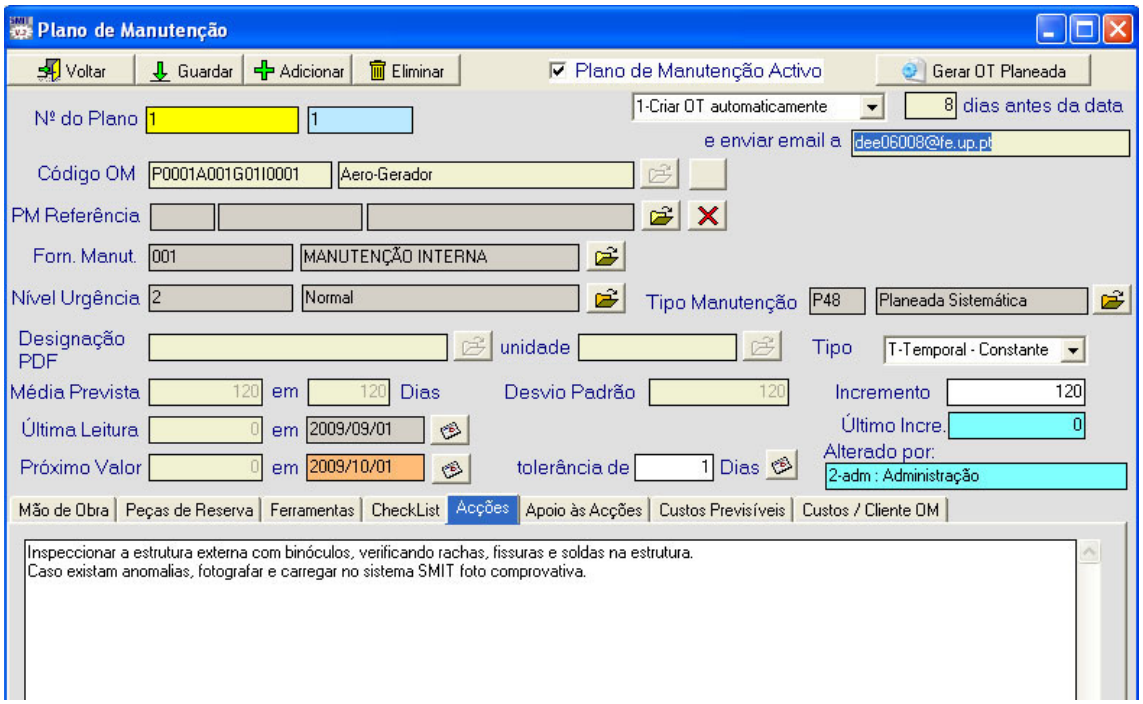



Figura 4.6: Um plano de Manutenção inserido no SMIT.

devem, ser efectuadas em simultâneo, pois assim melhora-se a disponibilidade do equipamento e racionalizam-se recursos. Estas acções estão condicionadas à não existência de nenhum motivo que iniba a simultaneidade de várias tarefas, desde que existam recursos para a sua efectivação.

Em termos de SMIT, nos Planos de Manutenção, há necessidade de indicar as acções, ou uma lista de tarefas, podendo também serem indicados quais os técnicos responsáveis pelos trabalhos, as peças-de-reserva a substituir e as ferramentas necessárias à execução dos trabalhos. O SMIT incorpora ainda um sistema de manuais, que podem conter ficheiros em formato electrónico, pdf, vídeos, texto em *Word*, folhas de cálculo, etc. A codificação dos planos de manutenção é constituída pelo código do OM, em conjunto com um campo numérico que indica o número de plano. De acordo com os exemplos apresentados previamente; **1/P0001A001G01I0001**, significa o primeiro plano para o objecto OM P0001A001G01I0001 - aerogerador.

A Tab. 4.2, exemplifica o planeamento das acções a realizar preventivamente num aerogerador, sendo, neste caso, o parâmetro de controlo, o tempo. Evidentemente, a calendarização da ordem de trabalhos, poderia ser função do tempo, do número de horas de funcionamento e do regime de ventos suportado durante a operação. Na Fig. 4.6 apresenta-se a inserção no SMIT do primeiro plano de manutenção indicado na Tab. 4.2.

O algoritmo de planeamento será discutido no capítulo 7.

| Código PM | Descrição do Plano | Periodicidade |
|---|---|---|
| 1/P0001A001G01I0001 | Verificar toda a pintura do aerogerador; caso exista corrosão, reparar. Verificar rachas, fissuras e soldas na estrutura. Caso existam anomalias, fotografar e carregar a foto no SMIT | 4 meses |
| 2/P0001A001G01I0001 | Verificar elevador e linha de vida | 3 meses |
| 1/P0001A001G01I0002 | Inspecção do cimento e base do aerogerador. Caso exista alguma anomalia, tipo fissura, reparar se for possível | 3 meses |
| 1/P0001A001G01I0003 | 1º Andar Tubular do aerogerador. Verificar parafusos e a tensão dos reapertos com X Nm. Caso haja parafusos partidos, devem ser substituídos os quatro parafusos adjacentes | 3 meses |
| 1/P0001A001G01I0004 | Indexado ao Plano 1/P0001A001G01I0003 | |
| 1/P0001A001G01I0005 | Indexado ao Plano 1/P0001A001G01I0003 | |
| 1/P0001A001G01I0006 | Verificar temperatura ambiente da Cabine do aerogerador (feito através do sistema de monitorização) | |
| 1/P0001A001G01I0007 | Parar o rotor, colocando uma das pás na horizontal, travada; de seguida, inspeccionar por dentro todas as pás do Rotor. Caso se verifiquem anomalias, documentar e, num prazo curto, caso se justifique, substituir a pá do rotor | 6 meses |
| 1/P0001A001G01I0008 | Massa de rolamentos; substituir totalmente | 6 meses |
| 2/P0001A001G01I0008 | Verificar óleo das engrenagens, se estiver bom manter; caso não tenha sido mudado nos últimos 5 anos, substituir. Enviar amostra para análises. | 12 meses |
| 1/P0001A001G01I0009 | Verificar o bloco anti-vibração que interliga o gerador eléctrico ao rotor | 3 meses |
| 1/P0001A001G01I0010 | Verificar o travão de paragem da cabine, óleo e disco, se for caso disso. Substituir disco, caso esteja no limite admissível | 4 meses |
| 1/P0001A001G01I0011 | Verificar nível de óleo do travão hidráulico do rotor, verificando a sua pressão, bem como os acumuladores de pressão | 3 meses |
| 1/P0001A001G01I0012 | Sistema de fixação das pás do rotor, verificar os sistemas de *backup* electrónico e verificar a sua completa funcionalidade, dado que em caso de falha de energia, as pás "têm de ser" retiradas de produção | 3 meses |
| 1/P0001A001G01I0013 | Verificar e mudar filtros do sistema de aquecimento/arrefecimento | 4 meses |
| 1/P0001A001G01I0014 | Verificar rolamentos do sistema de *Pitch* e electrónica de controlo | 3 meses |
| 1/P0001A001G01I0015 | Verificar em pleno funcionamento com câmara termo-gráfica o inversor electrónico | 6 meses |
| 1/P0001A001G01I0016 | Verificar e calibrar os sensores de vento | 4 meses |

Tabela 4.2: Exemplos de planos inseridos no SMIT, um dos quais respeitante ao apresentado na Fig. 4.6 .

## 4.5 Conclusões

Neste capítulo foi descrito, de forma sintética, a evolução do *software* de gestão da manutenção - SMIT - que incorpora várias tabelas e funções de base de dados, alguns relatórios e módulos para gestão da manutenção de aero-geradores. Quanto ao sistema de aquisição de dados, como é natural, foi também alvo de grande investimento, o qual será descrito nos próximos capítulos. Os aspectos actualmente tratados servem de base para a gestão da informação relativa aos aerogeradores, sendo a suporte para os restantes aspectos tratados na presente dissertação.

No próximo capítulo, serão introduzidos os modelos teóricos a utilizar nos aero-geradores para manutenção preventiva condicionada e detecção de avarias *on-line*, bem como, uma descrição mais detalhada do sistema de aquisição (pseudo-)síncrono. Salientam-se as grandes dificuldades em conseguir protocolos de colaboração com entidades detentoras de parques eólicos. Do ponto de vista do desenvolvimento, foram assumidos alguns pressupostos realistas sobre alguns aspectos. No entanto, foi possível validar o trabalho, mediante sigilo, tendo sido possível obter informação referente à produção de um parque eólico no Norte do País, graças à colaboração do INEGI/FEUP.

# Capítulo 5

# Manutenção preditiva de aero-geradores

Neste capítulo expõem-se os mecanismos necessários, do ponto de vista físico, para a criação de condições para a implementação da manutenção preditiva em aero-geradores. Por consequência, aborda questões relacionadas com a implementação prática dos dispositivos utilizados. O grande objectivo do sistema descrito neste capítulo prende-se ainda com a utilização de equipamentos de baixo custo, mantendo uma fiabilidade adequada. Na óptica comparativa com soluções comerciais, trata-se de uma vantagem relevante, dado ser possível baixar cada vez mais os custos de sistemas deste género possibilitando o alargamento da sua utilização. Outro aspecto importante é a possibilidade do sistema ser escalável/replicável de forma fácil e eficiente.

Nesta dissertação desenvolve-se uma metodologia de abordagem para a manutenção de parques eólicos, normalmente com diferentes aero-geradores distribuídos geograficamente por algumas dezenas/centenas de metros, tornando-se importante relacionar temporalmente os dados adquiridos entre os diferentes aero-geradores de um mesmo parque eólico. Porém, alguns parques têm apenas um anemómetro num ponto central, tornando-se relevante possuir uma ferramenta de aquisição de dados distribuída, que seja relativamente síncrona, de modo a permitir correlacionar os diferentes dados adquiridos em diferentes pontos geográficos.

## 5.1 A motivação

A primeira grande motivação dos especialistas das áreas da automação industrial, instrumentação virtual e tecnologias de redes IP, advém da constatação da falta de aproveitamento de muitos dos actuais protocolos de rede IP, que são extremamente úteis para a partilha de variáveis num meio distribuído e, no entanto, até há relativamente muito pouco tempo, pareciam áreas disjuntas. Concretamente, existe muito desenvolvimento na área de redes em transferência distribuída de

dados, com largos anos de utilização e, portanto, já na fase de maturidade do seu ciclo de vida, cuja proficuidade é relevante na área da instrumentação e automação industrial.

Só há poucos anos "acordou a corrida" das empresas fabricantes de material de automação industrial na integração de conectividade IP nos seus produtos de controlo industrial, desde autómatos a dispositivos específicos. Mesmo com esta mudança, ainda subsistiram resistências à utilização de protocolos abertos como o SNMP[1], tendo apenas a sua utilização sido recentemente iniciada por algumas marcas, tais como, por exemplo, a *Schneider Automation*, no seu mais recente PLC *Telemecanique Modicon M340*. Para esta área da automação, a norma IEC 61850 específica com rigor como se deve processar a sincronização de relógios entre dois equipamentos, independentemente do meio físico de comunicação.

Havendo no mercado microcontroladores de baixo custo, com inúmeras funcionalidades de conectividade, tais como *Ethernet*, *CAN*, *RS232*, *RS485*, é de todo relevante agregar estes aspectos para criar uma ferramenta de apoio à manutenção preditiva dos aero-geradores.

A outra grande motivação foi apurar, nomeadamente a nível das empresas líderes dos sistemas de aquisição de dados, a total ausência de mecanismos de sincronização naquela aquisição em diferentes pontos geográficos, à data de início dos trabalhos associados a esta dissertação. Esta constatação e o conhecimento dos protocolos de rede IP para sincronização, foram a semente para o presente desenvolvimento descrito nas secções deste capítulo.

## 5.2 Protocolos IP para sincronização temporal de relógios

Este é um dos temas sensíveis em várias áreas da actividade humana, cujo exemplo extremo é o da temporização de acções entre diversos intervenientes num cenário de guerra, o que mostra a relevância da questão. O mecanismo de sincronização das acções desenvolvidas neste tipo de tarefa é importante, sendo actualmente o GPS um dos mecanismos que permite "acertar o relógio"[2] a nível mundial.

Na área das redes IP existem alguns protocolos cuja diferença está na precisão com que é possível sincronizar os relógios internos dos diferentes equipamentos. Assim, algumas das opções protocolares passíveis de serem utilizadas são as seguintes:

- **NTP**[3] - protocolo de sincronização do relógio cuja informação é transferida entre equipamentos através de computadores utilizando para isso pacotes UDP. Permite a troca de mensagens entre vários equipamentos e actualiza o relógio interno a partir de uma fonte fiável. Este protocolo aceita desvios elevados ($\leq 128ms$) durante algum tempo ($\leq 900s$);
- **SNTP** - uma versão mais simples do NTP, cuja diferença reside ao nível do registo temporário dos dados transmitidos em troca de mensagens anteriores. Este protocolo é propício

[1] Protocolo utilizado em redes IP para partilha de variáveis entre os diferentes equipamentos.
[2] Relógio de tempo real
[3] NTP - ftp://ftp.registro.br/rfc/rfc1305

para utilizar em sistemas embebidos[4] como é o presente caso, sendo a sua precisão máxima de 1ms. A norma correspondente à quarta versão está documentada na RFC2030 havendo, no entanto, alterações correctivas publicadas na RFC4330[5];

- **IEEE 1588 PTP**[6] - trata-se de um protocolo recente, pois permite sincronizar os relógios de vários equipamentos com uma discrepância de $\pm$ 10 ns, sem grandes recursos computacionais. A versão actual desta norma denomina-se IEEE 1588-2008 e foi publicada em 2008[7];
- **SynUTC** - proposto pela Universidade Técnica de Viena, trata-se de uma alternativa ao protocolo PTP, mas sem grande aceitação no mercado. A ideia base dos autores é recolher o relógio presente no sinal de GPS e distribuí-lo de forma eficaz pela rede *Ethernet* com um grau de precisão semelhante ao PTP.

Quer o PTP, quer o SynUTC especificam a possibilidade da existência de *hardware* especial ao nível físico da interligação da rede, registando por *hardware* o instante da chegada de dados via *Ethernet*. Este procedimento melhora a precisão na medida do tempo entre a chegada do pacote de dados ao equipamento e o seu processamento pela camada de *software*, melhorando drasticamente o seu desempenho, atendendo a que este intervalo de tempo deixa de ser um valor aleatório.

Dos protocolos apresentados, foram utilizados o SNTP e o PTP, ambos por serem adequados a sistemas embebidos e, adicionalmente, o segundo pelas características de precisão indicadas.

### 5.2.1 SNTP

O protocolo NTP (sendo o SNTP uma versão simplificada do NTP), identifica os servidores NTP (relógios mestre), por estratos[8], de 0 a 16. Um servidor no estrato 16 é considerado como desactivado. O zero é reservado para um relógio atómico, por exemplo um receptor de GPS, iniciando-se a numeração para os servidores NTP no estrato 1 (servidor dependente de um relógio atómico). Todos os outros assumem mais uma unidade relativamente ao anterior, isto é, um servidor NTP de estrato 2 sincroniza o seu relógio num servidor NTP de estrato 1. Devido à numeração, apenas podem existir 15 dependências hierárquicas. O processo de sincronismo é feito com base num horário UTC baseado no GMT. Localmente, o servidor trata o tempo de acordo com a diferença horária da sua zona específica.

O NTP é demasiado abrangente pois é utilizado a nível mundial nas redes de dados para sincronizar servidores de páginas, ficheiros, agências bancárias, governos, etc, e incluí especificações, por exemplo, para conseguir detectar um ataque informático ao nível da tentativa de alteração do horário do servidor. A precisão obtida situa-se entre 1ms a 50ms, dependendo da rede onde se

[4] Do Inglês: *Embedded system* - sistema computacional desenhado para efectuar um conjunto dedicado de funções, normalmente com restrições de tempo real. Actualmente, estes dispositivos controlam a maior parte dos equipamentos, como máquinas de lavar, secar, etc.

[5] SNTP - ftp://ftp.registro.br/rfc/rfc4330

[6] *Precision Time Protocol*, proposto inicialmente pela *Agilent* baseado no trabalho de *John Eidson* um dos responsáveis pela aprovação do primeiro *standard* em 2002

[7] IEEE 1588-2008 - http://ieee1588.nist.gov/

[8] Do Inglês: *Stratum*

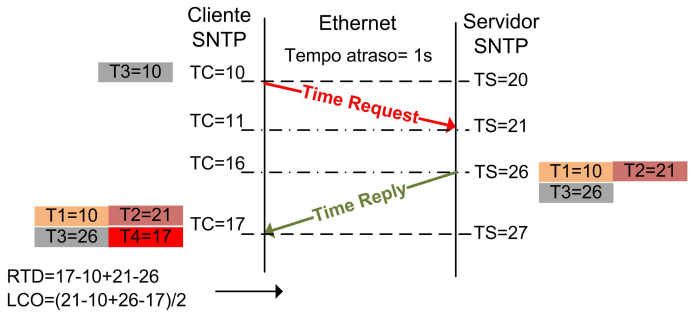



Figura 5.1: Mensagens SNTP enviadas entre mestre e escravo.

insere. Devido ao grau de complexidade, deu origem à simplificação SNTP para permitir distribuir um relógio horário a equipamentos de baixa complexidade computacional - caso típico dos sistemas embebidos.

Ao nível do SNTP, com arquitecturas cuidadas, são possíveis precisões na casa dos $\mu$segundos. O SNTP só deve ter um nível hierárquico, isto é, um servidor atómico e o cliente no dispositivo final. Cada cliente SNTP apenas troca mensagens com um servidor SNTP (deve ser do estrato 1).

No cliente SNTP devem ser produzidas duas variáveis: desvio do relógio (LCO) e o tempo de viagem (RTD) dos pacotes de ida e volta. O SNTP pode operar em modo cliente/servidor (*unicast*) ou em modo publicar/subscrever (*multicast*). Normalmente, no modo cliente/servidor, o cliente SNTP envia um pedido de tempo[9] e aguarda pela resposta através da qual pode calcular as duas variáveis já referidas. Em concreto, a informação que é recebida no cliente é a indicada na Fig. 5.1:

- **T1 - Tempo de saída no cliente SNTP**[10]: é o tempo de saída do pacote de pedido de acerto de relógio por parte do cliente SNTP, na sequência do seu pedido;
- **T2 - Tempo de chegada ao servidor SNTP**[11]: é o tempo do servidor à chegada do pedido de tempo do cliente;
- **T3 - Tempo de saída da resposta no servidor SNTP**[12]: é o tempo do servidor no instante de saída da resposta para o cliente;
- **T4 - Tempo de chegada ao cliente SNTP**[13]: é o instante temporal no momento da chegada da resposta do servidor.

As duas variáveis necessárias para acertar o relógio no cliente SNTP são calculadas da seguinte forma:

$$\begin{aligned} LCO &= \frac{T2-T1+T3-T4}{2} \\ RTD &= (T4-T1)-(T3-T2) = T4-T1+T2-T3 \end{aligned} \tag{5.1}$$

[9] *Time Request*
[10] *Originate Timestamp*
[11] *Receive Timestamp*
[12] *Transmit Timestamp*
[13] *Destination Timestamp*

Além da possibilidade de acertar o relógio, o cliente pode ainda alterar a frequência base do seu relógio, de forma a garantir que, de futuro, não exista deslizamento na contagem relativamente ao servidor, por ambos terem frequências diferentes. Supondo que T é o período de pedidos de tempo do cliente ao servidor SNTP e $f_{actual}$ a frequência de relógio, então a nova frequência base do relógio local deve ser actualizada de acordo com a equação:

$$f_{nova} = f_{actual} \cdot \frac{1}{1 - \frac{LCO}{T}} \tag{5.2}$$

Em termos de funcionamento do protocolo, o cliente, quando envia um pedido, deve colocar no campo T3 o valor da hora local. O servidor, ao receber o pedido, copia o valor de T3 para T1, e coloca os valores adequados nos campos T2 e T3, reenviando-os de volta ao cliente SNTP (ver Fig. 5.1). Este ao recebê-los, determina o valor de T4 e age em conformidade com as equações já apresentadas.

O modo *Multicast* não é utilizado nesta dissertação, podendo ser consultada a norma RFC4330 para mais pormenores.

### 5.2.2 IEEE 1588 PTP

O IEEE 1588 PTP assume-se hoje como um dos protocolos base da *Ethernet* industrial, onde as temporizações assumem especial importância. Este protocolo utiliza *hardware* específico para "carimbar" a hora a que determinado pacote foi recebido. Desta forma, é possível estimar o tempo entre a recepção física do pacote e o momento em que o *software* iniciou o seu tratamento. O *Jitter*[14] introduzido pela rede é calculado através da troca de vários pacotes PTP entre os dois equipamentos que pretendem sincronizar o relógio - que é uma das especificações do PTP.

**Funcionamento do PTP**

A função básica do PTP é fazer com que o relógio mais preciso na rede sincronize todos os outros equipamentos. Cada equipamento ligado a uma porta de rede denomina-se de relógio comum. Existem dois relógios, o Mestre e o Escravo. Em princípio, qualquer relógio pode executar tanto a função de mestre como a de escravo. A precisão de um relógio é classificada pelo protocolo em classes (estratos), sendo a classe mais elevada a que possui o valor 1. A selecção dos melhores relógios da rede é realizada automaticamente através do algoritmo BMC [91]. Apenas um dos relógios é o mestre[15], podendo haver escravos[16] a receber a informação do mestre e a desempenhar, ao mesmo tempo, a função de relógio mestre para outros equipamentos[17]. A base de sincronização protocolar é então a seguinte:

a) Organizar os relógios numa hierarquia mestre-escravo, com base nas propriedades da informação do relógio nas mensagens *multicast* do tipo *Sync*;

[14] Variação aleatória entre o tempo de transmissão de pacotes *Ethernet* entre os dois equipamentos
[15] Denominado *GrandMaster Clock*
[16] Denominado *Boundary Clock*
[17] Denominados os *PTP clients*

b) Cada escravo sincroniza-se com o RTC do mestre através de um conjunto de mensagens específicas (mensagens *Sync*, *Delay_Request*, *Follow_Up* e *Delay_Response*).

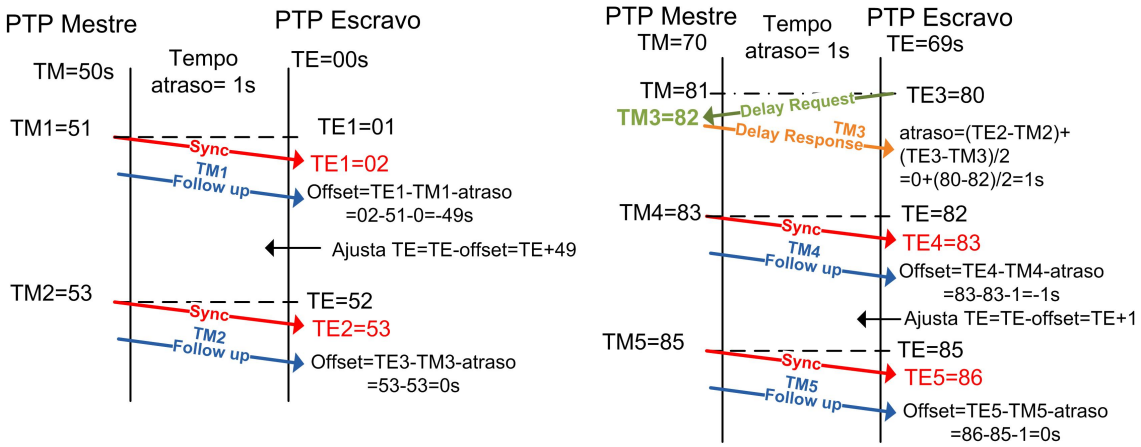



Figura 5.2: Mensagens PTP enviadas entre mestre e escravo.

A Fig. 5.2 apresenta-nos a forma como o protocolo PTP funciona, bem como as mensagens enviadas entre os dois dispositivos de forma a garantir o sincronismo dos relógios. Apenas as mensagens do tipo *Delay_request* têm origem no escravo e servem para medir o atraso de propagação no meio, à qual o mestre deve responder com uma mensagem do tipo *Delay_response*. As restantes mensagens têm origem no mestre, sendo da sua iniciativa. Um aspecto importante deste protocolo é o facto de considerar a duração da transmissão dos pacotes igual em ambos os sentidos, para permitir estimar o atraso de propagação. Outro aspecto a ter em consideração é que nem sempre os relógios inseridos em sistemas embebidos, de tempo real, são precisos ou, dito por outras palavras, podem adiantar-se ou atrasar-se, dependendo da implementação interna desse mesmo relógio (são os casos de relógios por *software*). Este aspecto não é contudo muito relevante pois o PTP estará a enviar consecutivamente mensagens de 2 em 2s e, para esta ordem de grandeza, os relógios dos sistemas embebidos são extremamente precisos.

Da Fig. 5.2 conclui-se que, caso exista uma alteração drástica no relógio mestre, em relativamente pouco tempo, reflecte-se no escravo, devido às mensagens de sincronismo (*Sync*).

Há, no entanto, questões relacionadas com o impacto directo no desenvolvimento desta estratégia, tais como a latência do sistema, nomeadamente nos equipamentos activos de rede ao longo de um trajecto. Neste caso, o tempo entre a recepção e o processamento dos pacotes PTP nestes mesmos equipamentos tem um comportamento que, normalmente, é aleatório e dependente do tráfego na rede.

Em resumo o funcionamento do protocolo é o seguinte:

a) O mestre envia uma mensagem *Multicast Sync*, registando o tempo (tm1) e, os escravos, após receberem a mensagem, registam o seu tempo interno (te2). A frequência destas mensagens é de dois segundos, $T_{sync} = 2s$;

b) Seguidamente, o mestre envia uma mensagem *Follow UP* com o seu tempo (tm1) para os escravos;
c) A diferença entre os dois tempos é o atraso do mestre para o escravo $dm2e = tm1 - te2$;
d) Os escravos enviam mensagens *Delay Request*, guardando o tempo de envio (te3) e o mestre, ao receber a mensagem, regista o tempo de recepção (tm4). A frequência destas mensagens é uniformemente distribuída entre 2 a 30 $T_{sync}$ unidades;
e) Seguidamente, o mestre envia uma mensagem *Delay Response*, com o tempo tm4;
f) A diferença entre os dois tempos é o atraso do escravo para o mestre $de2m = te3 - tm4$;
g) A propagação na rede é dada por $t_{prop} = (dm2e + de2m)/2$;
h) O acerto do relógio no escravo deve ser efectuado de acordo com a quantidade $\Delta t = dm2e - t_{prop}$.

### 5.2.3 Sincronismo em sistemas reais

O problema de sincronizar dois relógios deve levar em conta o desvio de frequência entre o relógio do emissor e o relógio do receptor. Este problema é característico dos sistemas nas telecomunicações podendo-se exemplificar para cada um dos sistemas analógico e digital, respectivamente, os receptores de FM e os receptores de televisão digital (TDT[18]). Um dos componentes essenciais na sintetização e sincronização de relógios é desempenhado pelas PLLs. As PLLs analógicas são circuitos para gerar um sinal em frequência, sincronizado com uma referência dada. No caso se serem analógicas, denominam-se de PLLs analógicas; se forem digitais DPLLs. No caso de serem implementadas por *software* denominam-se ainda como PLLs por *software* - SPLLs.

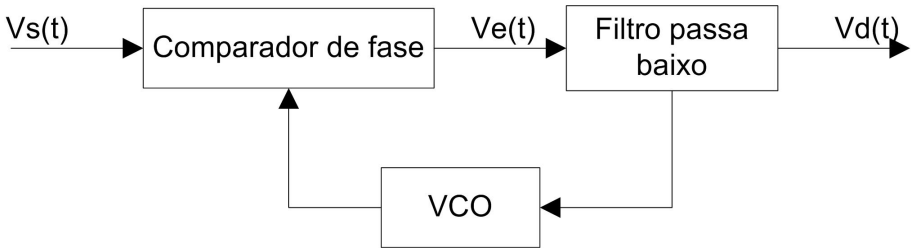



Figura 5.3: PLL analógica para descodificação de FM.

A Fig. 5.3 apresenta um esquema genérico de um desmodulador de FM, baseado numa PLL [92, 93], onde o sinal $V_s(t)$ se encontra modulado em frequência. O comparador de fase fornece um sinal proporcional ao desfasamento entre os dois sinais sendo, de seguida, depois de filtrado, fornecido ao oscilador controlado por tensão (VCO). O circuito tenta seguir a frequência do sinal de entrada, levando o erro de fase a zero; para isso, é necessário alterar a tensão do VCO de forma a que este oscile de acordo com a frequência instantânea de $V_s(t)$. A tensão aplicada ao VCO pode

[18] Sigla Portuguesa, em Inglês, DVB-T

ser amplificada sendo, no caso de um sinal FM, proporcional ao sinal que modulou em frequência a portadora (operação na zona linear de funcionamento).

No caso da transmissão de televisão digital [94], existe o problema básico de o receptor precisar de apresentar as imagens à mesma cadência de aquisição da câmara que produziu o filme (do lado do emissor). Caso contrário, ou os *buffers* no receptor/emissor enchem, ou ficam vazios, obrigando o receptor à perda de imagens ou a paragens, respectivamente. Para sincronizar o receptor a uma cadência fixa, o emissor codifica um sinal de relógio nos pacotes de dados. No receptor, a cadência desse sinal é utilizada para sincronizar o relógio em fase e em período, de forma a garantir uma cadência igual à da aquisição na recepção. O sinal codificado denomina-se PCR e depende directamente do sinal de relógio do emissor. Para efectuar essa sincronização utiliza-se, no receptor, uma PLL, neste caso digital. A Fig. 5.4 apresenta um esquema simplificado para um sistema de transmissão de vídeo digital síncrono a 27 MHz. Um sinal de relógio é colocado no *stream* de vídeo (SCR/PCR) sendo, no receptor, detectado e extraído através de uma PLL digital, que gera um relógio síncrono com o emissor, sendo esta a base de tempo para efectuar a descodificação e apresentação das imagens obtidas do descodificador de vídeo. Evidentemente, esta é uma "imagem" simplificada, apenas para realçar a importância do método de sincronismo dos relógios entre o emissor e o receptor. Neste caso de transmissão de vídeo, apenas interessa a sincronização relativa dos relógios, em frequência e fase.

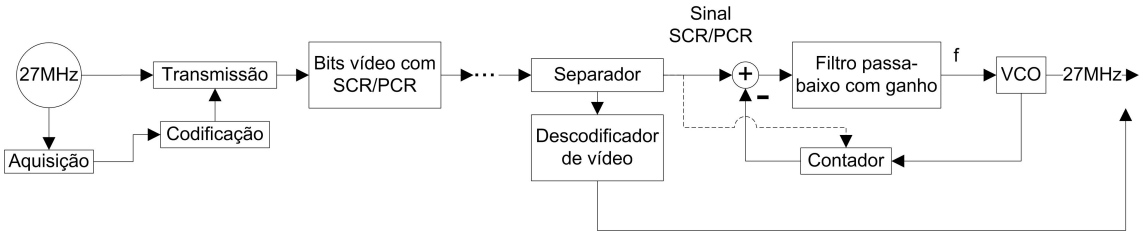



Figura 5.4: PLL digital para descodificação de televisão digital.

### 5.2.4 PLL na sincronização absoluta de relógios

Em termos formais, considere-se um relógio mestre com o qual se pretende fazer a sincronização em termos absolutos, cujo período é de $T_m = 1ns$, dando origem a duas variáveis, segundos e nanossegundos, i.e., $M_s : M_{ns}$. A cada ciclo deste relógio a variável nanossegundos é incrementada e, caso transborde, normalizada para os *segundos*.

Considere-se, como exemplo de um nó, um dos microcontroladores utilizados nesta dissertação, a operar a uma frequência de $f_{no}$ =50Mhz[19], sendo o seu período de relógio de $T_{no} = 20ns$. Pretendem-se manter neste nó duas variáveis $N_s : N_{ns}$ sincronizadas em valor absoluto com as

[19] A frequência poderá ser diferente, por ex. 40Mhz, de acordo com o microcontrolador usado. O seu valor influencia a precisão obtida no processo.

variáveis $M_s : M_{ns}$. A manutenção destas variáveis, normalmente, é efectuada através de uma interrupção do sistema denominada de *System Tick Interrupt*, sendo 10ms[20] um valor aceitável para a sua cadência. Desta forma, o valor de $N_{ns}$ é incrementado de $10ms$ ($10.000.000ns$) e, no caso de transbordar, será normalizado para os segundos. Nos microcontroladores, para esta interrupção ocorrer é necessário programar um contador de *hardware* (CNT_TICK), tal como descrito, com o valor de $CNT_TICK_MAX = 500.000 = 10ms/20ns$. O contador inicia-se em zero e conta até ao limite (500.000), gera uma interrupção (permite efectuar a manutenção de $N_s : N_{ns}$), reiniciando novamente a zero. Para se determinar a hora correcta é necessário ler o valor de CNT_TICK e somá-lo às variáveis $N_s : N_{ns}$. A precisão neste sistema será, no limite máximo, da ordem dos $20ns$, cujo valor é impossível de alcançar, pois trata-se do tempo de execução de uma instrução. A Fig. 5.6 exibe um diagrama da descrição anterior.

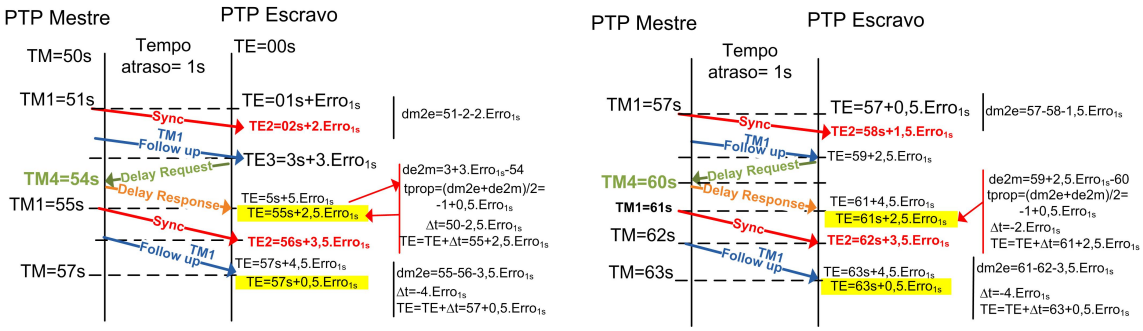



Figura 5.5: Diagrama temporal considerando um desvio de frequência no nó escravo.

Supondo diferenças na frequência de oscilação de ambos os relógios, o que é perfeitamente aceitável, sendo o emissor a referência, pode-se, sem perda de generalidade, modelizar esse diferencial, aqui designado por erro no relógio do receptor, isto é,

$$T_{no} = 20.T_m + \Delta T_{no} = 20ns + \Delta T_{no} \tag{5.3}$$

Para $\Delta T_{no}$ positivo significa que o período do nó escravo é superior ao do mestre, como consequência contará mais lentamente o tempo, ou seja introduz um erro negativo na contagem do relógio dada por:

$$\begin{aligned} Erro_{10ms} &= -CNT_TICK_MAX.\Delta T_{no} \\ Erro_{1s} &= 100.Erro_{10ms} \end{aligned} \tag{5.4}$$

Como a frequência do mestre era de $T_m = 1ns$, logo tem-se:

$$\Delta T_{no} = T_{no} - 20.T_m \tag{5.5}$$

Desta descrição fica claro que é objectivo obrigar $\Delta T_{no} \rightarrow 0$ sendo, para isso, necessário alterar o valor de $CNT_TICK_MAX$. Este controlo não é mais do que a função de uma PLL, neste caso,

[20] Valor dependente do microcontrolador, sendo obrigatório não gerar muitas interrupções por segundo. O seu valor não influencia o processo proposto, caso seja superior.

obrigatoriamente implementada por *software*. A Fig. 5.5 apresenta a evolução do relógio no nó escravo em função das mensagens do PTP, considerando o erro apresentado. Aplicando apenas a técnica de alterar as variáveis $N_s : N_{ns}$, constata-se ser impossível anular o erro, conforme o diagrama mostra. O controlo do parâmetro $CNT_TICK_MAX$ é então obrigatório, devendo ser alterado aquando do cálculo do erro $\Delta t$ quer na recepção de mensagens *Follow UP* (caso não seja enviada uma mensagem de *Delay Request*) quer na recepção de mensagens *Delay Response*, ou em ambas. Esta actualização do período base da interrupção *System Tick* está directamente relacionado com o funcionamento de uma PLL. Uma nota adicional refere-se ao valor de $CNT_TICK_MAX$ que tem os seguintes limites: não pode ser negativo, nem ultrapassar o tamanho máximo admissível do registo. Este aspecto significa aplicar um limiar ao valor calculado, antes de o reflectir no parâmetro $CNT_TICK_MAX$.

A Fig. 5.6 apresenta o diagrama integrado do processo de controlo do relógio local. As experiências realizadas reportam-se à alteração dos limites da função de limiar e da função de filtragem. O capítulo 8 apresentará os resultados relativos a este ponto.

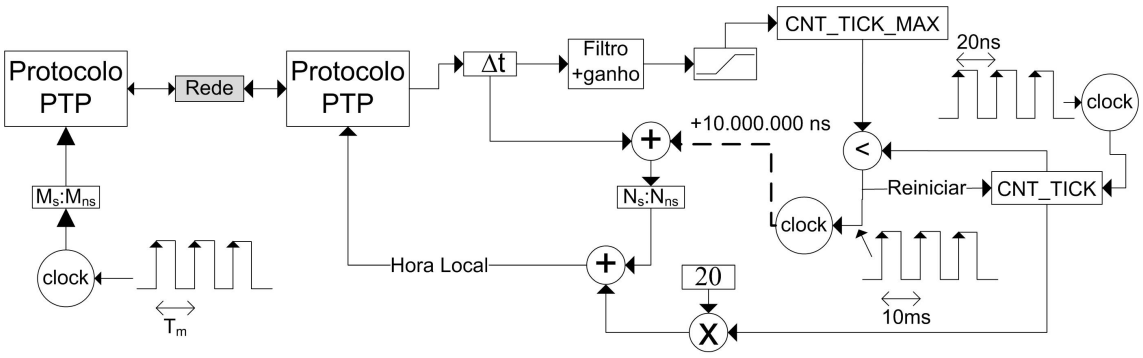



Figura 5.6: Diagrama da actualização de relógio.

### 5.2.5 Contributo para o sincronismo dos relógios

No diagrama de mensagens apresentado na Fig. 5.5, pode-se observar que o valor de $\Delta t$ não se mantém constante (no caso da PLL estar desligada, ou seja, sem protocolo a funcionar). Oscila, para o exemplo apresentado, entre $-2.Erro_{1s}$ e $4.Erro_{1s}$, embora a primeira situação seja predominante (ver resultados no capítulo 8).

Efectuando uma ligeira alteração no protocolo PTP, nomeadamente nas mensagens *Follow UP*, podem incluir-se nestas o tempo de envio da própria mensagem. Registando ainda no nó destino o tempo de recepção desta mensagem, seria possível obter o intervalo temporal decorrido nos dois sistemas e, portanto, estimar directamente o erro de desfasamento de frequência no nó escravo, $\Delta Erro$.

A Fig. 5.7 apresenta o diagrama integrado do processo de controlo, agora com a alteração introduzida, bem como o diagrama temporal para as modificações propostas de forma a calcular o desvio de frequência entre o nó mestre e o nó escravo. Obviamente, as questões de índole prática,

tais como "carimbar" correctamente as mensagens no envio e na chegada, são essenciais para o funcionamento desta nova alteração.

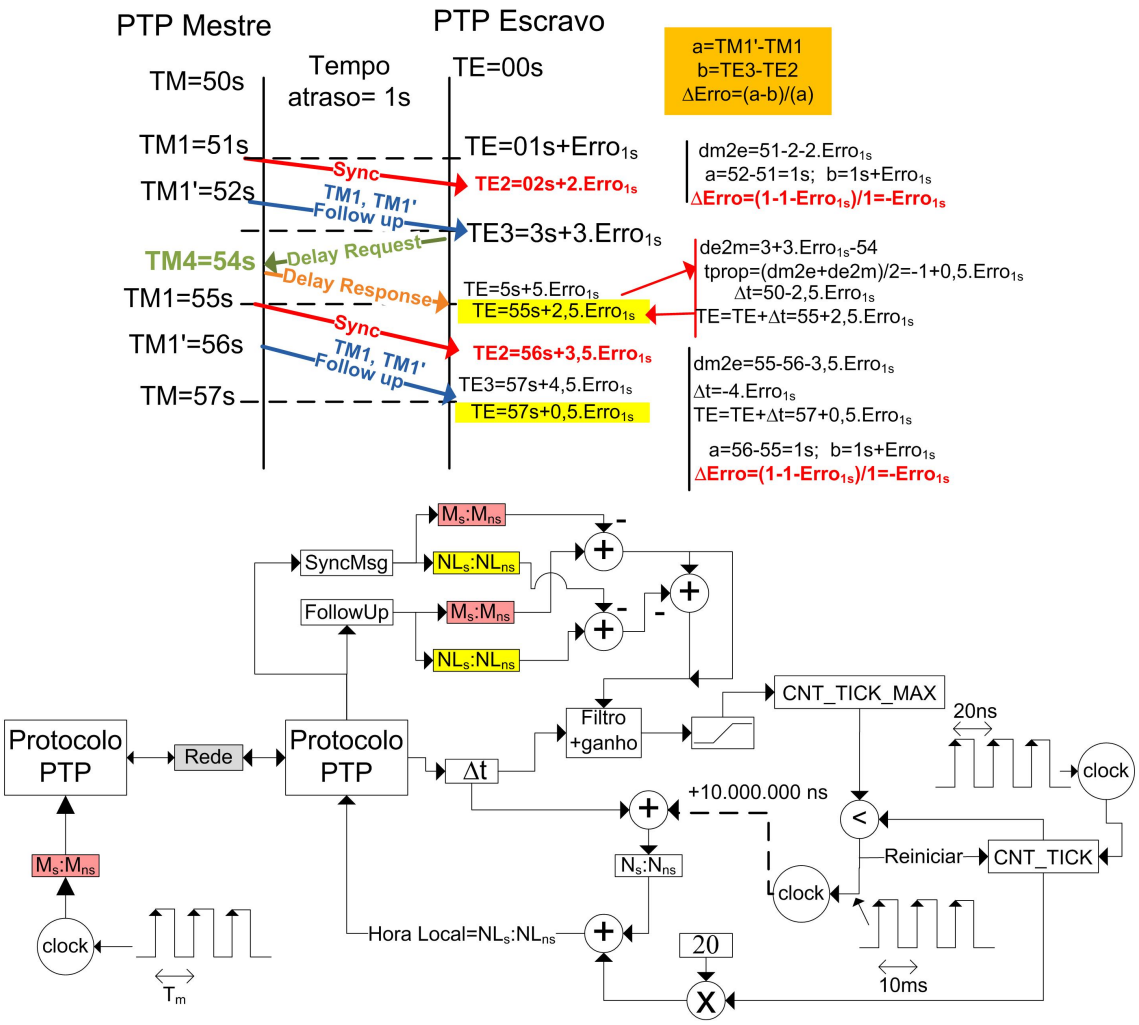



Figura 5.7: Modificação ao PTP para cálculo do $\Delta Erro$.

Na Fig. 5.7, o antigo $\Delta t$ continua igualmente a ser fornecido ao filtro passa baixo, para verificar a estabilidade de $\Delta Erro$, dado que o cálculo depende do intervalo temporal entre o envio das duas mensagens, o que pode causar instabilidade numérica. Se isto ocorrer (intervalo muito pequeno), o $\Delta Erro$ deixa de ser considerado nos cálculos.

Como se pode constatar nos diagramas das Figs. 5.6 e 5.7, ambas utilizam a teoria da PLL para anular o erro entre o período do relógio do emissor e do receptor. Trata-se de um conceito presente em muitos dos equipamentos do dia-a-dia e cuja utilidade para a resolução deste problema é preponderante.

## 5.3 Sistema para manutenção preditiva

Com base nos protocolos descritos na secção prévia, desenvolveu-se uma filosofia integrada de aquisição de dados de forma síncrona para a realização da manutenção preditiva de equipamentos eólicos.

O relógio mestre é o do horário do sistema *Linux* onde se encontra alojado o servidor do SMIT, que executa, em simultâneo, dois servidores de relógio, o SNTPD[21] (funciona como servidor com estrato 1) e o PTPD[22] (funciona como *GrandMaster Clock*), para distribuir o horário do servidor aos módulos de *gateway Ethernet-CAN* (explicado na secção 4.2, pág. 55, consultar ainda a Fig. 5.8). A versão baseada em microcontroladores PIC tem implementado o cliente SNTP, enquanto o ARM-Cortex tem implementado o cliente do PTP. O diagrama da interligação dos vários módulos encontra-se representado na Fig. 5.8. Com esta metodologia os relógios locais de cada módulo *gateway Ethernet-CAN* encontram-se sincronizados com o mestre - servidor *Linux*. A pergunta que se coloca é a seguinte: como vai esse relógio reflectir-se ou chegar aos módulos de aquisição de dados através da rede *CAN*? A solução para este problema poderia ser implementada nas seguintes formas possíveis:

- O módulo *gateway Ethernet-CAN* funcionaria como *Boundary Clock* no caso do PTP ou como servidor de estrato 2 (por se encontrar dependente do servidor SMIT) no caso do SNTP. Esta filosofia implica a implementação da parte de servidor de cada um destes protocolos nos respectivos módulos *gateway* bem como nos módulos de aquisição, que teriam de ter implementado os clientes dos protocolos PTP/SNTP e, simultaneamente, manterem um relógio de tempo real gerindo as temporizações para realizar as aquisições no tempo certo;
- O módulo *gateway Ethernet-CAN* gere as temporizações e, atempadamente, manda um sinal de aquisição para a rede *CAN* obrigando os módulos de aquisição mencionados na mensagem a procederem a essa aquisição. Esta modalidade dispensa a implementação da parte de servidor dos protocolos de distribuição de relógio bem como a propagação das suas mensagens sistemáticas a cada um dos nós da rede *CAN*. No entanto, aumenta o processamento no próprio módulo podendo limitar o número de nós *CAN* que podem ser suportados sobre o seu controlo;
- Uma temporização síncrona, que seria ainda uma forma mais simples, na qual se definiria um relógio de *clock*, enviado para a rede *CAN* e, cada módulo de aquisição só poderia adquirir sinais em múltiplos da taxa indicada pelo módulo *gateway Ethernet-CAN*, lançando os dados na rede nos intervalos temporais da mensagem de *clock CAN* da *gateway*.

Das três hipóteses apresentadas, apesar de existirem ainda outras possíveis, seria interessante implementar as três filosofias efectuando num estudo comparativo em função da carga da rede *CAN* e do número de nós, para determinar qual a mais apta e em que situações. Optou-se porém por estudar a segunda alternativa. Justifica-se a opção, por: a primeira carece da propagação do

[21] OpenBSD's ntpd -http://www.openntpd.org/
[22] http://ptpd.sourceforge.net/

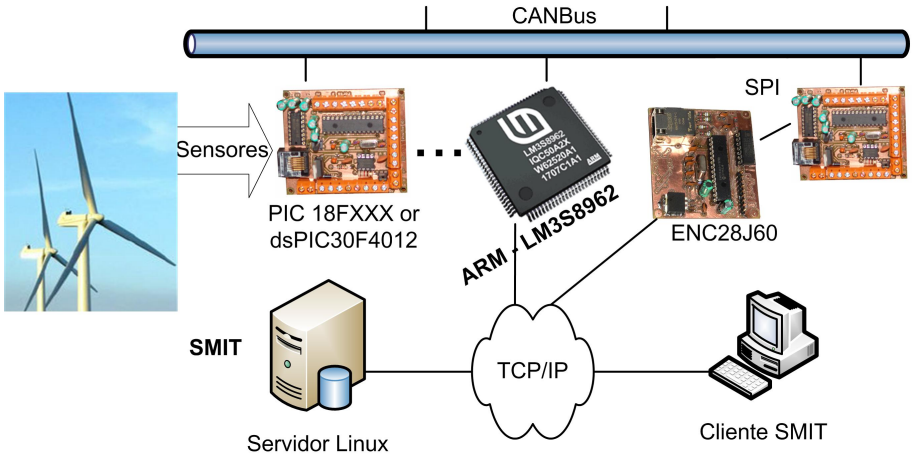



Figura 5.8: Sistema de suporte à manutenção preditiva.

relógio até cada nó e nestes é inevitável um RTC local (nem sempre possível); a última envolve limitar a frequência de aquisição dos módulos.

### 5.3.1 Gestão do módulo *gateway Ethernet-CAN*

Após a decisão tomada, nos módulos *gateway Ethernet-CAN*, doravante referidos como nós mestre (os módulos de aquisição serão referidos como nós E/S *CAN*), foi criada uma estrutura de dados mantida em memória, definindo toda a temporização das aquisições dos módulos E/S *CAN* e cujas mensagens de aquisição devem chegar a esses módulos no tempo certo para a aquisição ocorrer dentro da normalidade. Evidentemente, há limites para o número de nós E/S *CAN* geridos pelo nó mestre, bem como para a taxa de aquisição. Em caso de ruptura, deve-se dividir o troço do nó mestre em dois nós mestre.

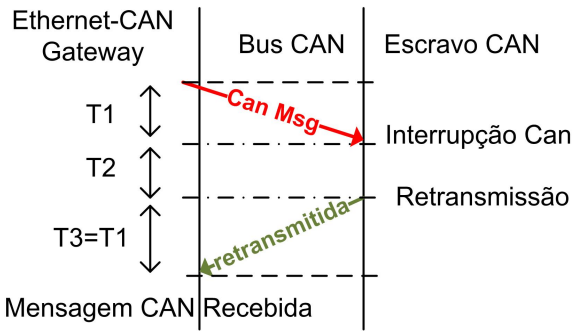



Figura 5.9: Esquema de mensagens para medir o tempo de propagação na rede *CAN*.

Dado que os sistemas que fazem a aquisição de dados comunicam através de uma rede industrial *CAN*, há necessidade de enviar as mensagens a esses equipamentos alguns instantes antes dos dados serem adquiridos. Assim, torna-se necessário, em primeira instância, calcular o tempo de

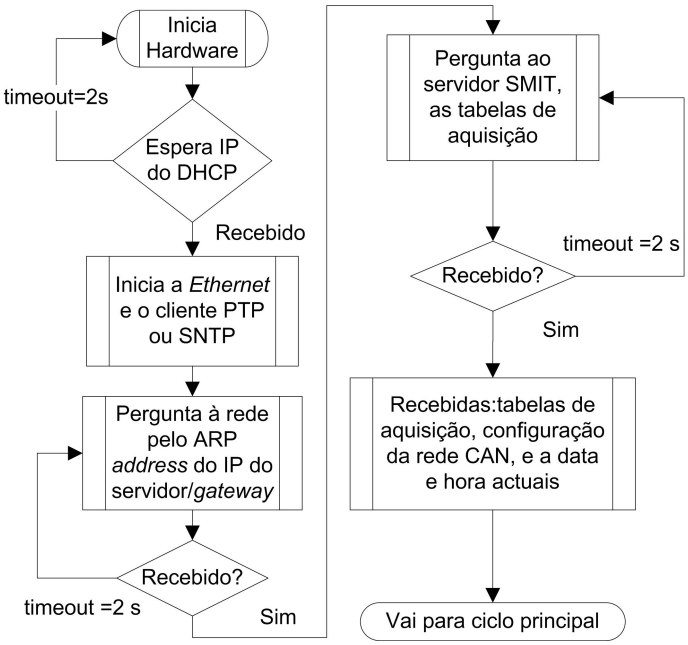



Figura 5.10: Algoritmo executado no nó mestre, modo configuração.

transmissão entre o nó mestre e os nós E/S *CAN*, levando em conta o tamanho da mensagem *CAN*. A Fig. 5.9 apresenta o método para efectuar esta medida, utilizando uma estratégia semelhante ao PTP para medir o tempo de propagação no meio físico, isto é, o nó mestre envia uma mensagem e, ao receber a resposta, divide por dois o tempo decorrido ($T_1 \approx T_3 \approx T$ e $T_2 \ll T$), determinando assim o referido atraso de propagação na rede *CAN*. As mensagens de aquisição enviadas para a rede *CAN* pelo nó mestre devem então ser enviadas no instante $t$:

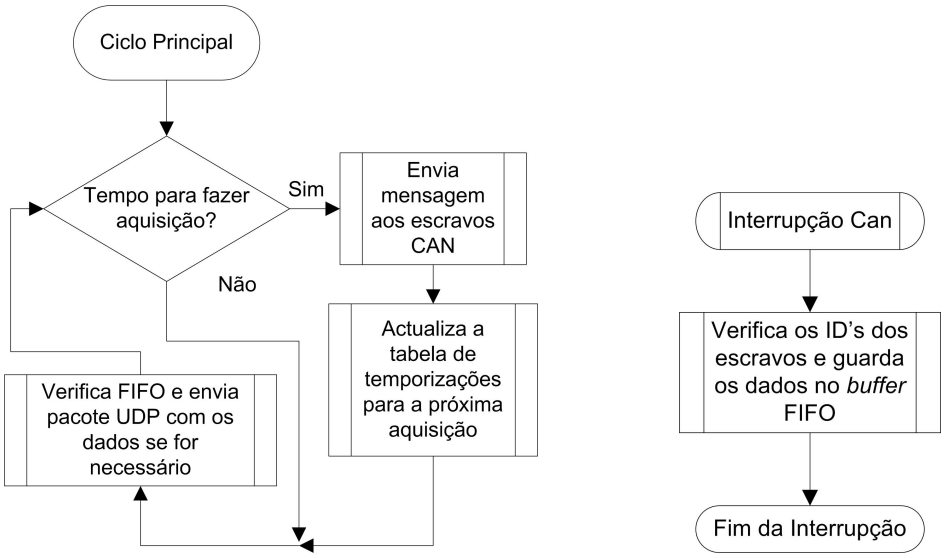
$$\begin{aligned} tempo_{propagacão} &= \frac{T_1+T_2+T_3}{2} \approx \frac{T+0+T}{2} = T \\ t &= tempo_{requerido} - tempo_{propagacão} \end{aligned} \tag{5.6}$$

Para este tempo ser calculado em tempo real no sistema, foi reservada uma mensagem de *CAN* especifica para medir este tempo, caso seja necessário.

As Figs. 5.10 e 5.11 apresentam o fluxograma completo executado pelo nó mestre, distinguindo-se dois estados: a) O módulo está em modo de configuração, à procura na rede dos dados referentes a si, b) O módulo está em modo de funcionamento normal, controlando a aquisição.

Para o primeiro modo de funcionamento (Fig. 5.10) o nó mestre executa os seguintes procedimentos:

- Obter um IP por DHCP (processo normal nas redes IP), facilitando a sua inserção em qualquer rede local, sem necessidade de mudar o *firmware*;
- Iniciar o cliente PTP ou SNTP conforme a situação;
- Obter o endereço IP do servidor SMIT através de um processo de *broadcast*;




(a) Fluxograma do ciclo principal

(b) Fluxograma do modo de interrupção

Figura 5.11: Algoritmo executado no nó mestre, aquisição.

- Perguntar ao servidor SMIT pelas suas tabelas de aquisição, de forma a controlar os nós E/S *CAN*;
- Receber as tabelas recolhendo ainda a informação dos parâmetros de inicialização da rede *CAN*, o tempo em segundos+nanossegundos do servidor, bem como a tabela de tempos de propagação na rede *CAN*, calculadas externa/internamente através de outro/mesmo programa.

Após este modo de *setup*, o nó mestre entra no estado de controlo da aquisição dos dispositivos colocados na rede industrial *CAN* e dos quais tem conhecimento através da tabela de aquisição que recebeu. O fluxograma referente a este estado encontra-se na Fig. 5.11.

Para minimizar problemas de "atenção" imediata às solicitações, os meios de comunicação no nó mestre foram implementados por interrupção (recepção de pacotes *Ethernet* e de mensagens *CAN*). Igual procedimento foi elaborado nos nós E/S *CAN* na recepção de mensagens *CAN*.

Para dar algum tempo de processamento ao servidor SMIT, os dados recebidos no nó mestre não são logo enviados, mas sim guardados localmente numa tabela circular com capacidade para 54 kbytes e enviados em mensagens UDP de 1024 bytes ao servidor SMIT. A correcta sinalização de pacote bem recebido é efectuada pelo servidor SMIT ao respectivo nó mestre e só nesta situação os dados são retirados da tabela circular para dar espaço a novos dados. Na situação da tabela circular esgotar a sua capacidade devido à incapacidade de comunicação entre o nó mestre e o servidor SMIT, os novos dados recebidos são descartados sendo esse facto contabilizado através de um contador; logo que exista conectividade, este facto é enviado ao servidor para este tomar conhecimento da anomalia, isto é, perda de dados.

O controlo de falhas ao nível da aquisição dos nós E/S *CAN* também é efectuado no módulo mestre da seguinte forma: sempre que uma mensagem *CAN* de aquisição é enviada para a rede industrial, é incrementado um contador; na recepção do dado adquirido, esse contador é decrementado ficando a zero. Se a resposta do nó de aquisição falhar, no próximo ciclo de aquisição o contador não estará a zero, significando assim uma anomalia. Este facto gera uma mensagem especial para o servidor SMIT e a aquisição desse nó é cancelada. Este tipo de anomalia pode acontecer por várias razões, tais como: o nó bloqueou; a taxa de aquisição é elevada demais para a velocidade do nó; a rede *CAN* tem demasiados nós e, portanto, gera muito tráfego, não permitindo aos nós *CAN* com menor prioridade "escoar" a sua informação (ver anexo B).

Evidentemente, a configuração da rede de aquisição, tal como apresentado pela Fig. 5.8 deve ser cuidadosamente projectada no número de equipamentos na rede *CAN* e na taxa de aquisição de cada um, caso contrário, todos os módulos sobre a supervisão do nó mestre poderão entrar em ruptura (obviamente, no sentido de não desempenharem a sua função).

### 5.3.2 Gestão do módulo *CAN* de aquisição

Os módulos de aquisição ligados à rede *CAN* são relativamente fáceis de conceber do ponto de vista do projecto. São simplesmente responsáveis por receber mensagens de aquisição, procederem à respectiva aquisição e, de seguida, enviar os dados recolhidos.

Uma outra metodologia de aquisição, passível de implementar nestes módulos, seria caracterizada pela seguinte sequência de passos:

a) O módulo adquire dados a uma taxa fixa definida;

b) Para os dados recolhidos, calcula a cada instante a média, o valor máximo e o mínimo;

c) Sempre que solicitado pelo módulo mestre, envia os dados, nomeadamente, o último valor e os valores médio, máximo e mínimo;

d) Após responder ao módulo mestre, reinicializa os valores e volta ao item a).

Estes módulos podem ainda receber mensagens especiais, às quais devem responder de imediato para medir o tempo de propagação no bus *CAN* a pedido do nó mestre.

No caso de aquisição de vários pontos por amostra, como é o caso da situação da aquisição de dados de vibração por acelerómetros, apenas recebem uma mensagem de aquisição e de seguida, fazem a obtenção dos *n* pontos à taxa pré-programada e reencaminham a informação para o nó mestre.

## 5.4 Configuração do sistema na base de dados do SMIT

Toda a arquitectura de aquisição de dados tem de ser configurada no sistema SMIT para permitir que os nós *gateway Ethernet-CAN* recebam correctamente as respectivas configurações. Este passo é efectuado através do módulo de Manutenção de Condição, através do qual se indica a informação pretendida em conjunto com o algoritmo de tratamento de dados (anexo C).

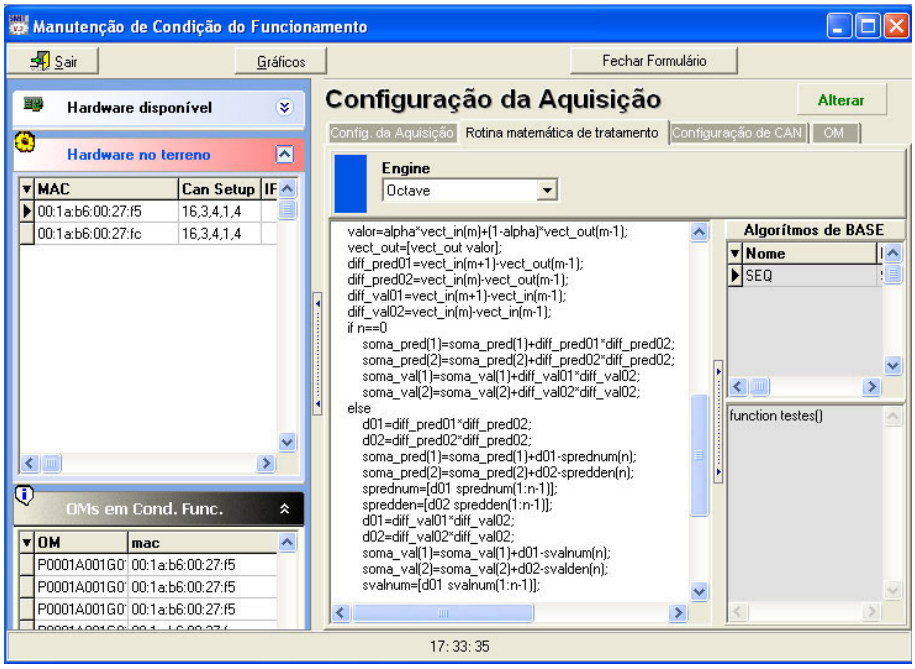



Figura 5.12: Módulo de configuração da condição de funcionamento dos diferentes objectos de manutenção. Indicação de rotinas matemáticas de tratamento de dados.

### 5.4.1 Planos de Manutenção de condição

O módulo de manutenção de condição tem por função, entre outros aspectos, permitir ao utilizador do *software* configurar o sistema de aquisição. Na presente versão, é suportado o *hardware* desenvolvido na sequência da presente dissertação. A Fig. 5.12 apresenta uma das opções de configuração onde se pode indicar a rotina de tratamento matemático dos dados, quer seja em linguagem *Octave* ou do *software R*. Para além disso, o utilizador deve indicar em qual das linguagens escreveu o programa. Existem ainda disponíveis, como se pode verificar no lado direito da figura, as funções matemáticas pré-definidas de raiz, sendo um ponto de partida para desenvolvimento de novos algoritmos. Outra das particularidades desde módulo, Fig. 5.13, é permitir ao utilizador indicar os parâmetros de aquisição dos dados. Não se trata de indicar qual a frequência de aquisição, pois esse parâmetro é directamente programado em cada módulo para sinais onde, em cada amostra, sejam necessários vários pontos. Exemplo disso é a aquisição de dados para cálculo de uma transformada de *Fourier*, como é o caso da análise de vibrações. O utilizador, nesta situação, apenas indicará por quantos pontos é feita a amostra, sendo a taxa de amostragem fixa e apenas indicada no *firmware*. Todos os restantes parâmetros da aquisição podem ser alterados pelo utilizador, nomeadamente:

- **Início da aquisição** - Indica o instante temporal caracterizado pela data, hora, minuto, segundo e nanossegundo a partir do qual o sistema deve iniciar a recolha de dados;
- **Fim da aquisição** - Instante temporal no qual deve terminar a aquisição, sendo, no entanto, possível indicar que a recolha de dados nunca deverá terminar;
- **Incremento da aquisição** - O tempo entre amostras, que para amostras simples, i.é, de

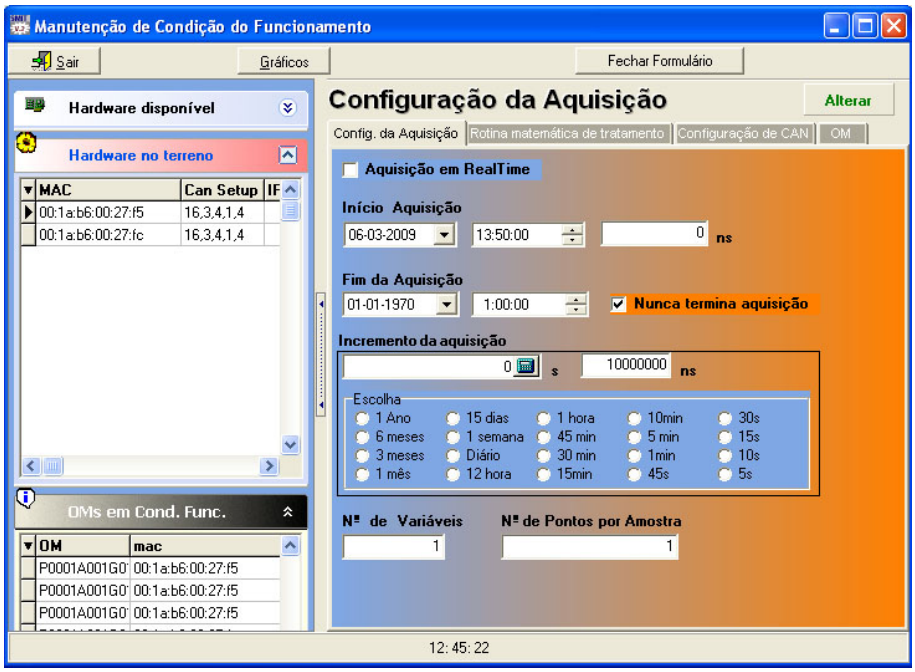



Figura 5.13: Módulo de configuração da condição de funcionamento dos diferentes objectos de manutenção. Parâmetros de aquisição.

apenas um ponto, equivale a indicar a frequência de aquisição; evidentemente, o módulo e a rede de comunicações deverão suportar a taxa de informação programada;

- **Número de variáveis** - Por exemplo, numa aquisição de vibração poderá ser efectuada por acelerómetro de três eixos, dando assim origem a três componentes em cada amostra;
- **Número de pontos por amostra** - Utilizando o exemplo já dado relativo às vibrações pode-se, por exemplo, numa intervenção de manutenção, indicar com que periodicidade é necessário efectuar uma aquisição de vibrações, mas onde cada aquisição possa ser constituída por vários pontos, cujo valor é então indicado neste campo;
- **Aquisição em tempo real** - Esta opção elimina o parâmetro de **incremento na aquisição**, e "obriga" os equipamentos no terreno a efectuarem a aquisição à máxima taxa possível.

No anexo B serão apresentadas as características dos módulos de aquisição cuja interligação com esta secção deste capítulo seja natural e, numa primeira análise, todos os equipamentos no terreno de entrada/saída utilizam uma rede *CAN* interligada a um módulo cuja função é controlar a aquisição e transferir os dados da rede *CAN* para a rede *Ethernet*, tendo como destino o servidor SMIT. O módulo é denominado de *gateway* de dados. Por opção não tem nenhuma configuração, excepto o seu endereço de rede *Ethernet*, o *MAC Address*. É exactamente este parâmetro que deve ser indicado no módulo, tal como pode ser visualizado na Fig. 5.14. Além deste parâmetro é também indicada a informação para inicializar o controlador de *CAN* do módulo *gateway*. Cada uma destas placas pode controlar vários módulos de aquisição de entrada/saída, interligados no mesmo meio físico *CAN*. A única limitação é a velocidade de operação que deve ser igual em todos os módulos ligados ao meio físico.

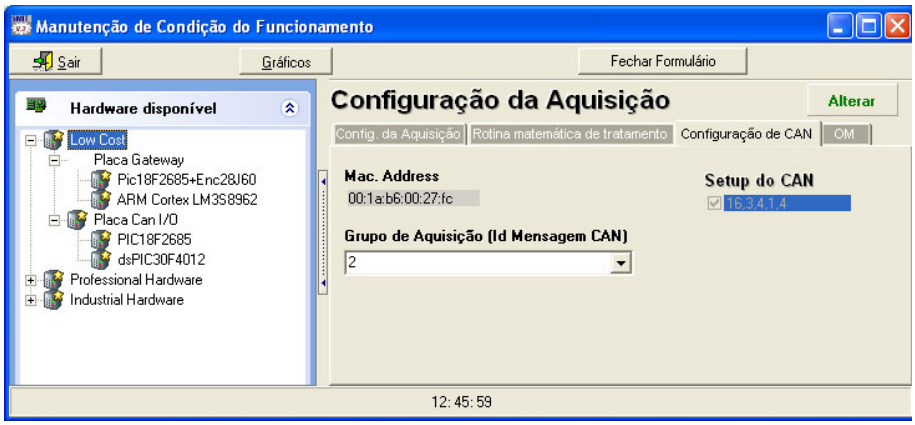



Figura 5.14: Módulo de configuração da condição de funcionamento dos diferentes objectos de manutenção. Configuração de um módulo de E/S *CAN*.

Nesta configuração indica-se a identificação da mensagem *CAN* que permite identificar univocamente cada módulo de aquisição. No mesmo meio *CAN*, só pode existir um módulo a transmitir, um tipo de mensagem (ID mensagem *CAN*) e, por simplificação, estipulou-se que o valor associado apenas assume valores múltiplos de dois. A vantagem desta codificação está explicada em pormenor no anexo B, embora seja importante referir que com esta codificação, o módulo *gateway* endereça todos os equipamentos com uma única mensagem. A desvantagem do processo passa pela diminuição do número de diferentes mensagens disponíveis, limitando desta forma o número de equipamentos interligados no mesmo meio físico.

Finalmente, menciona-se um aspecto ainda não referido: em que condições é que o programa deverá gerar uma Ordem de Trabalhos em função dos dados recolhidos no campo? Os limites de operacionalidade dos valores medidos no terreno, tal como indicado no anexo C, são programados nas rotinas matemáticas de tratamento dos dados. Caso a rotina dê indicação de violação dos limites, é gerada uma Ordem de Trabalhos para o Objecto de Manutenção em questão; de seguida o processo de aquisição interrompe até ao momento da conclusão dos trabalhos, iniciando-se novamente a partir desse instante.

Este comportamento excluí o estudo apresentado na secção 7.2 (pág. 109), podendo vir a ser alterado, no futuro, para a sua integração.

## 5.5 Aquisição de dados

Para validar o mecanismo de aquisição de dados utilizaram-se três tipos diferentes de sensores: um acelerómetro de três eixos; um anemómetro de vento de copos e um anemómetro de ultra-sons (Fig. 5.16).

O anemómetro de ultra-sons utilizado é o modelo WMT50 da *Vaisala*, com controlador digital incorporado, cuja interligação com o sistema de aquisição é efectuado através da porta série. Há necessidade de configurar o anemómetro previamente enviando-lhe comandos *ascii*.

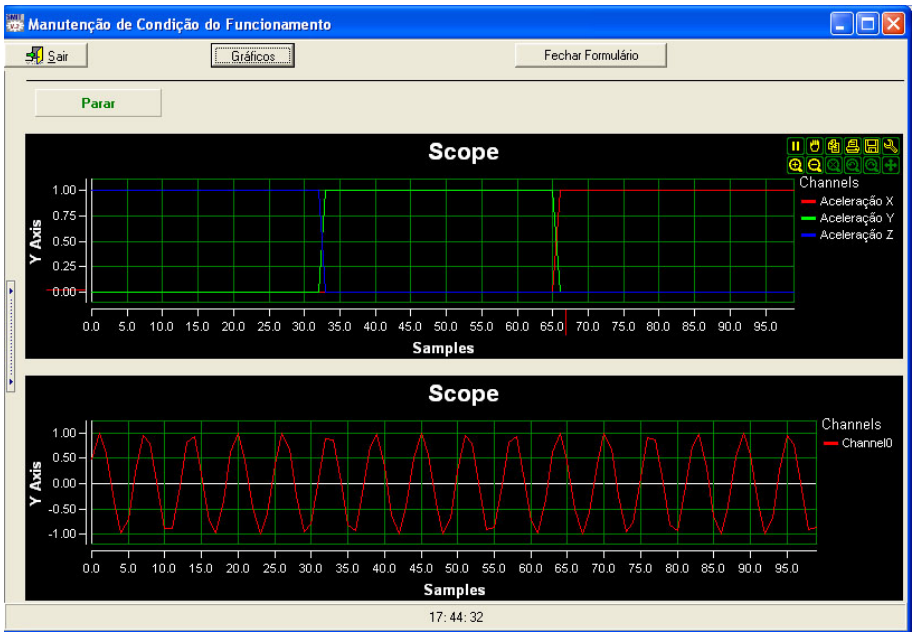



Figura 5.15: Módulo de configuração da condição de funcionamento: visualização da recepção de dados.

O anemómetro analógico de copos funciona como um gerador AC e fornece uma tensão analógica proporcional à velocidade de rotação. A frequência da tensão é directamente proporcional ao número de rotações do anemómetro. Este sinal é convertido para digital e, de seguida, é contabilizado o número de pulsos por segundo, sendo essa a informação enviada pelo nó de E/S.

O acelerómetro de três eixos utilizado é o modelo LIS3L06AL da ST[23], que mede acelerações máximas de 2g/6g conforme configurado e apresenta tensões de saída compatíveis com o módulo de aquisição, podendo ser interligado directamente. A sua largura de banda máxima é de 1.45 kHz.

A Fig. 5.17 apresenta o diagrama de fluxo de informação, com os dados e sinais de controlo, na etapa de funcionamento, de um nó mestre. Torna-se clara a ideia principal de "distribuir" o relógio do servidor SMIT a todos os módulos que efectuam a ponte entre a rede IP e a rede *CAN*. Nesse módulo mestre são gerados os sinais de temporização para aquisição e recolhidos os dados recebidos dos nós de aquisição e, por último, enviados ao servidor SMIT para processamento e salvaguarda. Todo o processo de troca de informação na rede *Ethernet* é feito através de pacotes UDP.

Outro dos aspectos salientes da Fig. 5.17 é a relevância do conjunto mas, em particular, da parte mais sensível do sistema, isto é, o nó *Ethernet-CAN*; a sua inoperacionalidade obriga à paragem do sistema. Um dos aspectos importantes para trabalho futuro, seria criar redundância neste ponto do sistema, através da inclusão de um segundo módulo ligado à mesma rede *CAN* e realizar estudos de qual o melhor comportamento do elemento introduzido. Algumas ideias para hipotéticas possibilidades do seu comportamento poderiam ser as seguintes:

[23] http://www.st.com

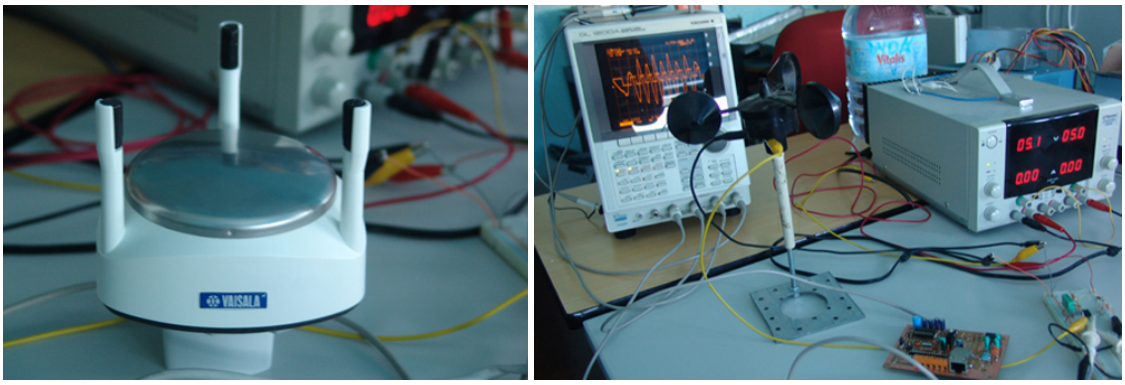



Figura 5.16: Esquerda: anemómetro de ultra-sons, direita: anemómetro analógico de copos.

- Verificar os sinais de aquisição na rede *CAN* e determinar, desta forma, o estado de funcionamento do nó mestre principal; em caso de falha assumir as suas funções;
- Receber todos os dados na íntegra e reencaminhá-los para o servidor SMIT, libertando esta tarefa do nó mestre principal; em caso de avaria de um dos dois nós, o nó sobrante assumiria a tarefa por inteiro.

## 5.6 Conclusões

Neste capítulo foi apresentado o sistema de suporte para a manutenção preditiva de aero-geradores, com ênfase para a estrutura modular implementada. O sistema permite divulgar um relógio entre os vários módulos, garantindo, sensivelmente, a mesma hora em todos os pontos do sistema de aquisição. Este processo permite, por exemplo, no mesmo parque adquirir em sintonia os mesmos dados em diferentes aero-geradores.

No que respeita a algumas opções no âmbito do *software*, facilmente podem ser alteradas e desenvolver-se novas teorias sobre o funcionamento do *puzzle* descrito neste capítulo. No entanto, como é evidente, o sistema está projectado para trabalhar com vários sensores com diferentes taxas de amostragem. Na actual implementação, a configuração proposta é a seguinte: ao nível das vibrações, efectuar aquisição, por exemplo, uma vez por dia; a análise da potência e da velocidade do vento ser efectuada de cinco em cinco minutos; e a análise de temperatura na caixa de engrenagens de minuto a minuto, etc.

No caso da metodologia passar pela monitorização em contínuo, a capacidade de processamento existe nos nós E/S *CAN* bastando reescrever o *firmware*. Em concreto, poderiam monitorizar o equipamento e, ao assinalar ligeiros desvios da "normalidade", iniciar o envio de dados ao servidor; caso a situação se revertesse parava o envio da informação.

O respectivo cliente do SMIT referido na secção 5.4 desempenha um papel importante na configuração do sistema, sendo ainda nesta fase indicado pelo utilizador o método de tratamento dos dados. A ideia é tornar este processo mais amigável para o utilizador, nomeadamente através de

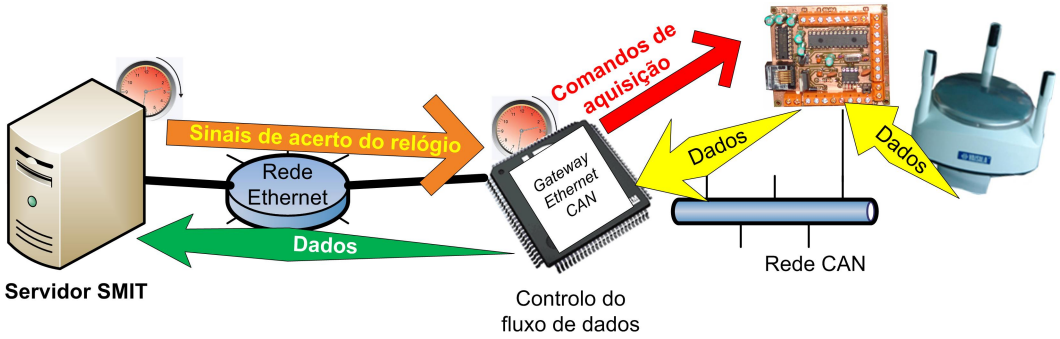



Figura 5.17: Diagrama do fluxo de informação no sistema global.

programação gráfica de arraste e ligação dos módulos por fios, como se de um esquema electrónico se tratasse, mas o desenvolvimento efectuado, neste âmbito, não se encontra ainda estável, nem se mostra particularmente relevante para a concretização desta dissertação.

O próximo capítulo apresentará os modelos preditivos de ruptura.

# Capítulo 6

# Modelos de predição

Neste capítulo faz-se a descrição dos modelos mais adequados à manutenção preditiva, direccionados aos geradores eólicos.

Os temas apresentados incluem dados de fiabilidade relativos a parques eólicos e algoritmos de classificação da condição de funcionamento - a partir de um dado conjunto de medidas é possível inferir se o equipamento está avariado ou em bom funcionamento. Foram estudadas duas situações, condicionadas à disponibilidade da informação existente. Fez-se a análise de um gerador de indução sob a perspectiva da medida das correntes eléctricas e das vibrações, a detecção macroscópica do estado de funcionamento do gerador através da curva de potência e desenvolveram-se algoritmos de previsão da evolução de um sinal temporal, a partir da análise de séries temporais, na perspectiva de antecipar valores e adequar as operações de manutenção.

Os detalhes encontram-se ao longo do texto onde serão expostas as principais vertentes dos assuntos acima descritos, repartidas por alguns dos anexos, sendo os seus resultados apresentados no capítulo 8.

## 6.1 Fiabilidade de parques eólicos

De acordo com alguns estudos publicados referentes à fiabilidade dos equipamentos presentes nos aero-geradores, a sua homogeneidade não é uma característica, em consequência dos diferentes sistemas de cada marca.

Segundo o relatório da Sandia [95], a operação de manutenção de turbinas eólicas representa cerca de 20% do custo total do projecto. Desta forma, a manutenção planeada, seja sistemática ou condicionada, é vital, na perspectiva de racionalizar os custos globais de operação.

As turbinas eólicas são constituídas por vários sistemas complexos, que devem todos funcionar correctamente para o bom desempenho global. Normalmente, não existe redundância nesses sistemas, o que leva, em caso de falha, à paragem do sistema, com os respectivos custos directos e

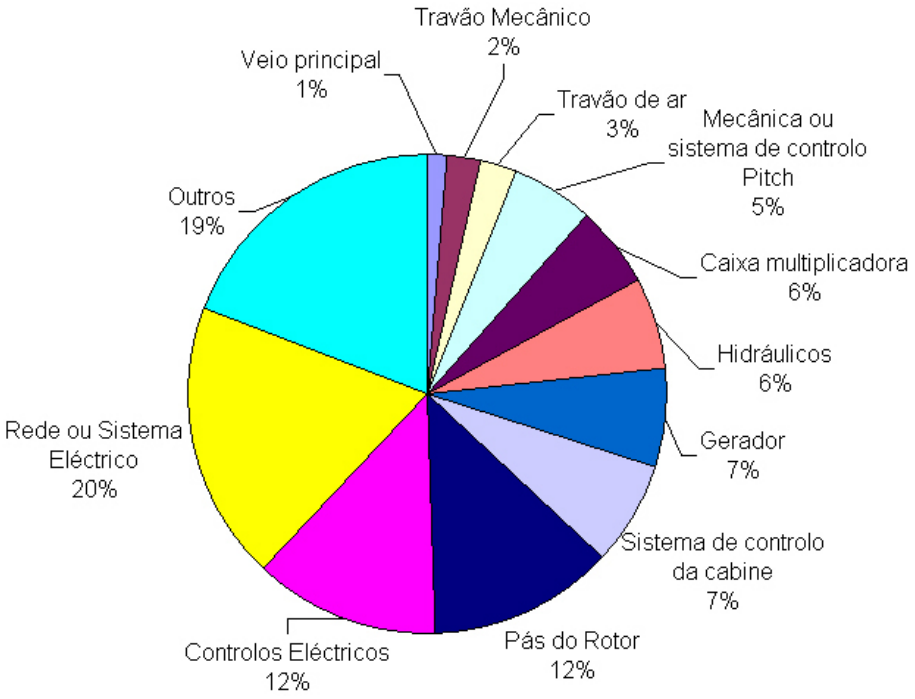



Figura 6.1: As avarias típicas e a sua distribuição pelos diferentes equipamentos de uma turbina eólica, dados referente à Tab. 6.2 [16, 17].

indirectos associados. Nesta perspectiva, a monitorização da condição de funcionamento é importante, pois permite monitorizar o grau de "saúde" dos diversos órgãos do sistema [18, 96, 97].

Embora seja difícil prever quando um gerador vai ter um curto-circuito, existem outras variáveis cuja monitorização é, desde há largos anos, utilizada noutras actividades, tais como, por exemplo, as variáveis associadas aos equipamentos rotativos, como vibração, temperatura, etc. Muitas das áreas ligadas à parte mecânica dos aero-geradores podem utilizar técnicas transpostas de outras áreas do conhecimento, embora com as necessárias adaptações, pois a variação de "carga" de trabalho neste género de tecnologia é diferente e dependente do regime de ventos.

As Figs. 6.1 e 6.2 caracterizam as falhas mais típicas bem como os componentes envolvidos neste tipo de equipamentos. Na Fig. 6.2 é representado um gráfico em diagrama de barras, onde aparecem descritas as taxas de falhas de forma decrescente, podendo-se identificar uma distribuição de *Pareto*. O gráfico indica ainda quais os tempos de paragem nas turbinas eólicas, sendo que o gerador e a caixa de transmissão obrigam às paragens mais demoradas, apesar da frequência relativa ser baixa.

A Fig. 6.3 faz a comparação entre várias turbinas com diferentes potências de produção e mostra que a taxa de avarias é maior em geradores de maiores potências. Entre outras razões, tal facto pode ser explicado pela necessidade de equipamentos mais complexos, torres mais altas, rotores com um maior diâmetro, tendo como consequência um maior número de problemas associados.

Para se perceber qual a fiabilidade de cada elemento [16], as Tabs. 6.1 e 6.2 apresentam valores típicos para MTBF e MTTR. Da sua análise conclui-se da existência de diferenças elevadas entre

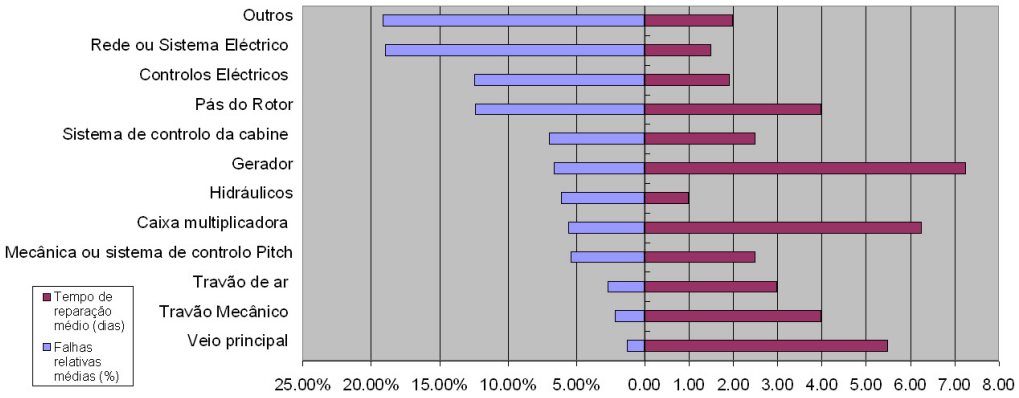



Figura 6.2: Distribuição de falhas de um grupo de turbinas eólicas e as respectivas consequências, em termos dos custos indirectos; dados referente à Tab. 6.2 [16, 17].

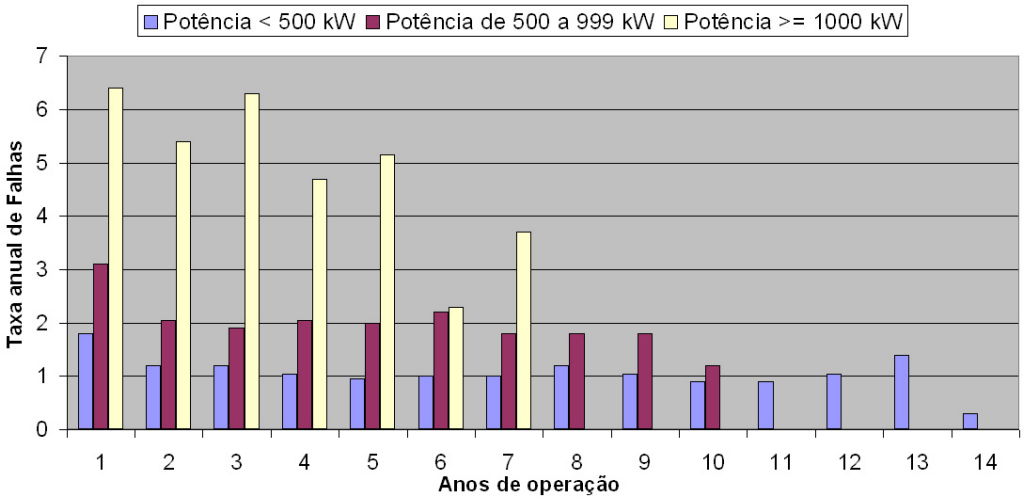



Figura 6.3: Percentagem de falhas em função da potência da turbina [18].

| **Componente do sistema** | **Aero-geradores (Alemanha) MTBF(horas)** | **Aero-geradores (Dinamarca) MTBF(horas)** |
|---|---|---|
| Pás do rotor | 39297 | 252033 |
| Travão de ar | 180078 | 1286050 |
| Travão Mecânico | 223447 | 627055 |
| Veio principal | 365339 | 807174 |
| Caixa multiplicadora | 87174 | 218871 |
| Gerador | 73234 | 365534 |
| Sistema de controlo da cabine | 69504 | 318903 |
| Controlos Eléctricos | 39205 | 175561 |
| Hidráulicos | 79363 | 285195 |
| Rede ou Sistema Eléctrico | 25708 | 450643 |
| Mecânica ou sistema de controlo de *Pitch* | 90472 | 1236712 |
| Outros | 25449 | 51871 |

Tabela 6.1: Dados de Fiabilidade de aero-geradores, MTBF de aero-geradores instalados na Alemanha e Dinamarca [16, 17].

os diferentes componentes.

## 6.2 Modelos para predição de avarias

As várias técnicas actualmente existentes para predição de avarias, em muitos casos, baseiam-se em cálculos da estimativa da evolução futura de parâmetros chave. Através destes métodos de predição dos valores dos parâmetros, infere-se o seu próximo valor e, por consequência, o estado do sistema por comparação entre as várias zonas de funcionamento. Garantir que determinado parâmetro se encontra dentro de uma gama de valores é uma metodologia usada no controlo estatístico de processos, embora sem predição, servindo-se apenas dos valores actualmente medidos.

Nos estudos realizados neste trabalho, na área da detecção de avarias, destacam-se:

- **Os que genericamente determinam avarias** - utilizou-se um classificador que, para cada medida realizada, determina se o aero-gerador se encontra em situação de avaria ou não. Este método utiliza as variáveis da potência produzida *versus* velocidade do vento e velocidade do rotor do aero-gerador;

- **As que determinam avarias nas partes rotativas do gerador eléctrico** - usam os dados de um acelerómetro e as correntes de linha para inferir se existem avarias no gerador;

- **Os métodos de predição de séries temporais** - podem ser utilizados genericamente para estimar a evolução de sequências de dados sensoriais típicos. Este processo pode ser aplicado no domínio do tempo ou no domínio espectral, concretamente, na amplitude de frequências chave, as quais são calculadas com base em parâmetros do gerador.

| **Componente do sistema** | **Aero-geradores (Alemanha) MTTR(horas)** |
|---|---|
| Pás do rotor | 96 |
| Travão de ar | 72 |
| Travão Mecânico | 96 |
| Veio principal | 132 |
| Caixa multiplicadora | 150 |
| Gerador | 174 |
| Sistema de controlo da cabine | 60 |
| Controlos Eléctricos | 46 |
| Hidráulicos | 24 |
| Rede ou Sistema Eléctrico | 36 |
| Mecânica ou sistema de controlo de *Pitch* | 60 |
| Sensores | 36 |
| Cubo do rotor | 84 |
| Estrutura de apoio/cabine | 72 |
| Outros | 48 |

Tabela 6.2: MTTR de um grupo de aero-geradores Alemães (MTBF apresentado na Tab. 6.1).

### 6.2.1 Classificação do estado da curva de potência com base em SVM

Numa perspectiva geral, para se determinar uma avaria no gerador utiliza-se um método de classificação, neste caso multi-objectivo, denominado SVM (*Support Vector Machine*) [98, 99, 100]. Este método concorre directamente com os algoritmos de redes neuronais, tendo a vantagem de apresentar uma maior facilidade na escolha dos parâmetros e também na fase de treino. Tipicamente este método foi inicialmente criado para decisões binárias (verdadeiro/falso) podendo ser adaptado para fazer classificações de vários grupos. Na sua forma de aplicação, são efectuadas várias medidas de características, tais como, por exemplo, temperatura, corrente, tensão, pressão, etc. Este conjunto de valores define um estado para o equipamento. A técnica de SVM classifica este conjunto de valores associando-lhe um estado. Evidentemente, antes de ser utilizado num ambiente de produção, o algoritmo de SVM deve ser treinado com medidas e com os respectivos estados para, em produção, poder efectivar classificações correctas.

Na literatura, tal como foi acima referido, existem vários métodos de classificação, sendo um deles as redes neuronais. Os modelos de *Markov* também se adequam à classificação de vectores de medidas, embora utilizando transições probabilísticas.

Para a correcta utilização do SVM, na fase de treino é considerado um conjunto $D$ de vectores:

$$D = \{(\vec{x}_1, y_1) \cdots (\vec{x}_i, y_i)\} \, com \, \vec{x}_i \in \Re^n, y_i \in \{a, b, c, \cdots\} \tag{6.1}$$

onde $\vec{x}_i$ é o vector de medidas e $y_i$ é a classificação atribuída a esse vector.

Sem perda de generalidade e para uma melhor representação gráfica, considere-se apenas o caso de $\Re^2$ e $y_i \in \{1, -1\}$. Neste caso, o SVM determina a fronteira entre as diferentes amostras,

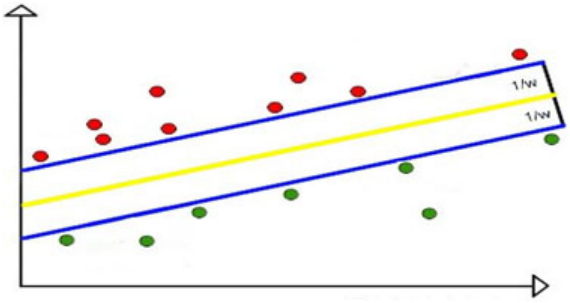



Figura 6.4: Em $\Re^2$ linear, o SVM calcula a fronteira que minimiza a distância aos pontos que definem a fronteira.

as quais têm de ser fornecidas indicando o grupo a que pertencem. Supondo que a linha que separa os conjuntos tem distância 1 ou -1, conforme indicado na Fig. 6.4 pelas linhas azuis, estas satisfazem os seguintes critérios:

$$\begin{cases} \vec{w}.\vec{x}+b=-1 \\ \vec{w}.\vec{x}+b=1 \end{cases} \tag{6.2}$$

O valor de $\vec{x}$ é a entrada, $\frac{2}{\|\vec{w}\|}$ é a distância entre as duas categorias (margem) que se pretende maximizar. Equivale a minimizar a norma:

$$\begin{gathered} \min_{\vec{w}} \left\{ \frac{1}{2}\vec{w}^T.\vec{w} \right\} = \min_{\vec{w}} \left\{ \frac{1}{2} \parallel \vec{w} \parallel \right\} \\ sujeito\ a: \ y_i \left[ \vec{w}.\vec{x}+b \right] \geq 1, i=1,\cdots,n \end{gathered} \tag{6.3}$$

Este problema é denominado problema da optimização quadrática. A solução indica que a linha amarela pode ser calculada pela combinação linear dos elementos pertencentes ao conjunto D e que a predição da função $f(\vec{x})$ para uma nova amostra $\vec{x}$ é dada por:

$$f(\vec{x}) = sign\left( \sum_{i=1}^{n} y_i \alpha_i (\vec{x}.\vec{x}_i) + b \right) \tag{6.4}$$

onde $n$ é o número de elementos do conjunto $D$, $\alpha$ e $b$ são parâmetros usados pelo SVM com o objectivo de construir uma solução óptima (indicada na figura pela linha amarela). No entanto, existem casos em que os dados não são linearmente separáveis (Fig. 6.5). Nesta situação aplica-se um mapeamento para dispor os elementos classificados de forma a que possam ser separados linearmente (no caso de $\Re^n$ um plano, no caso de $\Re^2$ uma recta). As funções que fazem este mapeamento são denominadas na literatura como *kernel functions*. As mais usuais e documentadas como produzindo bons resultados nos estudos realizados na literatura são: função linear, quadrática, polinomial, RBF e MLP. Dependendo da escolha da função de *kernel*, o SVM poderá produzir boas ou más classificações. A função que, normalmente, é recomendada é a RBF. Mais detalhes podem ser consultados em [100, 101, 102, 103, 104].

No presente trabalho é relevante e importante decidir o modo como se deve aplicar e, mais importante ainda é decidir quais as variáveis indispensáveis para uma correcta validação do processo

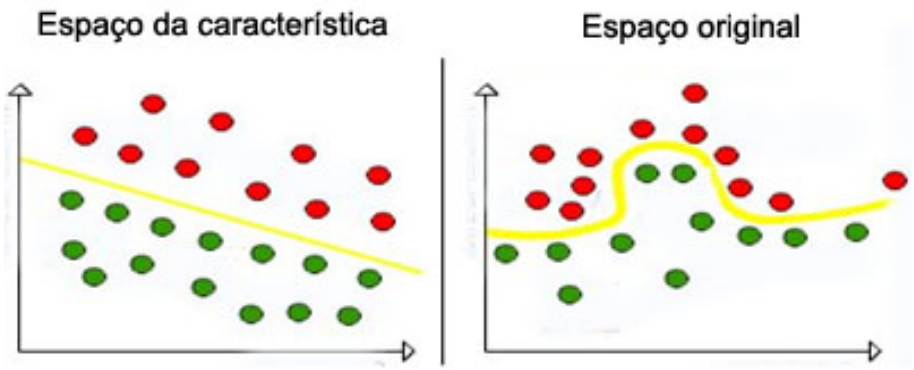



Figura 6.5: As funções de *kernel* efectuam um mapeamento de forma a linearizar o espaço.

em questão, isto é, determinar o estado do aero-gerador com base nas medidas de campo.

O vector (conjunto de dados obtidos das características do aero-gerador), poderia, e deveria, contemplar a potência, velocidade de rotação do veio do rotor e do veio do gerador, velocidade do vento, alinhamento da cabine do aero-gerador com a direcção do vento, ângulo das pás do rotor e estado dos travões do rotor. Em principio, estas são as variáveis essenciais, do ponto de vista "macroscópico" para determinar o estado de operação do aero-gerador (Eq. 6.5).

$$\vec{x}_{aero-gerador} = \begin{bmatrix} Velocidade\ Vento \\ Direcção\ do\ vento\ -Direcção\ da\ cabine \\ Velocidade\ veio\ rotor \\ Velocidade\ veio\ gerador \\ Potência\ activa \\ Estado\ do\ sistema\ de\ travagem \end{bmatrix} \tag{6.5}$$

Devido à indisponibilidade da medição de algumas destas variáveis, nomeadamente, alinhamento da cabine, ângulo das pás do rotor e estado dos travões do rotor, optou-se por utilizar os dois vectores de classificação (para averiguar qual o melhor) indicados nas equações 6.6 e 6.7,

$$\vec{x}_{aero-gerador} = \begin{bmatrix} Velocidade\ Vento \\ Potência\ activa \end{bmatrix} \tag{6.6}$$

ou

$$\vec{x}_{aero-gerador} = \begin{bmatrix} Velocidade\ veio\ rotor \\ Velocidade\ veio\ gerador \\ Potência\ activa \end{bmatrix} \tag{6.7}$$

A alteração descrita anteriormente facilita a análise e aplicação do classificador, tendo como consequência a redução do número de situações anómalas detectáveis. Como corolário desta afirmação, por exemplo, não é possível determinar se o sistema de travagem funciona correctamente.

Quanto ao número de estados, apenas é necessário utilizar um classificador com dois estados,

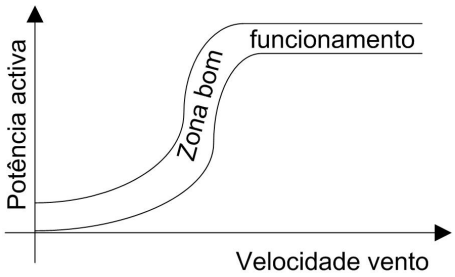



Figura 6.6: Faixa de treino considerada como em operação adequada da curva de potência do aero-gerador.

avariado ou em bom funcionamento, construído a partir da curva de potência experimental do aero-gerador. Evidentemente, haverá a necessidade de atribuir uma tolerância, isto é, a curva de potência será dada por uma "faixa" tal como se pode ver na Fig. 6.6 dentro da qual ainda se considera o equipamento em "operação adequada". Uma das sugestões para um trabalho futuro poderá passar por se considerarem várias zonas de condição de funcionamento, nomeadamente, bom estado, estado razoável, quase em avaria e avariado, permitindo assim efectuar uma transição mais suave entre as zonas de bom funcionamento e avariado. O problema reside na definição das taxas de transição entre os diferentes estados de avaria.

A aplicação deste algoritmo funciona de acordo com o referido anteriormente, isto é, como um indicador *on-line* do estado de funcionamento macroscópico do aero-gerador, cujos resultados são apresentados no capítulo 8.

### 6.2.2 Detecção de avarias nos geradores das turbinas eólicas

Devido ao seu baixo custo, robustez e facilidade de manutenção, os motores de indução têm larga aplicabilidade na indústria. As duras condições de trabalho, nomeadamente no arranque, provocam-lhes grandes aquecimentos e elevados esforços mecânicos, causa maioritária de futuras avarias. Interessa pois, por uma questão de qualidade, minimizar o impacto das paragens dos motores de indução nos sistemas de produção, maximizando o trabalho destes através de uma política de manutenção planeada.

A maioria das avarias nos motores são típicas e associadas às grandezas eléctricas e electromagnéticas envolvidas, pelo que, com ferramentas adequadas, é possível saber em tempo real qual o estado físico da máquina e prever as suas avarias. A detecção precoce de falhas poderá ser vital para impedir maiores danos nos sistemas e para melhorar a segurança das pessoas.

Existe um grande número de abordagens relacionadas com a detecção e diagnóstico de falhas nos motores e, segundo alguns autores, podem ser subdivididas em sete grupos:

1. Redundância física;
2. Redundância analítica;
3. Análise de sinais;

4. Sistemas baseados em conhecimento;
5. Aprendizagem com falhas;
6. Aprendizagem sem falhas; reaprendizagem *on-line*;
7. Aprendizagem sem falhas; verificação *on-line*.

O trabalho aqui desenvolvido enquadra-se no quinto grupo (aprendizagem com falhas), onde se pretende, com base nos dados recolhidos, identificar o estado do motor (bom/avariado). É secundário diagnosticar o tipo de avaria (espiras em curto-circuito, barra partida, etc.), embora seja exequível. Poderá questionar-se o porquê do estudo de motores quando o que se pretende é trabalhar com geradores? A resposta é simples, existem muitos aero-geradores que utilizam como gerador o motor de indução, ou então o gerador síncrono. Para o caso do gerador de indução, as situações estudadas no motor de indução podem, desta forma, apresentar um modelo semelhante de comportamento do gerador.

**O modelo do motor**

O motor de indução trifásico apresenta-se, normalmente, em dois formatos diferentes: rotor bobinado e em gaiola de esquilo (os enrolamentos são substituídos por barras metálicas).

A velocidade de sincronismo do rotor é dada pela expressão:

$$n_s = \frac{f}{p}.60\ (rpm) \tag{6.8}$$

em que **f** é a frequência e **p** o número de par de pólos. Assim, para um motor de 4 pólos, ou seja $p = 2$, a funcionar com a frequência da rede, 50Hz, o veio vai rodar a 1500 rpm.

**Deslizamento**

O deslizamento **s** [1] representa a diferença entre a velocidade de rotação do rotor e a velocidade síncrona:

$$s = \frac{n_s - n}{n_s} \Leftrightarrow n = n_s\,(1-s) \tag{6.9}$$

onde **n** é a velocidade real em rpm medida no rotor do motor. O escorregamento varia com a carga aplicada ao motor e, no caso do gerador, pelo binário resistente associado ao consumo.

A frequência da tensão do rotor é calculada pela fórmula:

$$f_2 = s\frac{p.n_s}{60} = s.f = \frac{n_s - n}{n_s}.f \tag{6.10}$$

Em função das avarias mais tipificadas, existem determinadas frequências do espectro das correntes que devem ser analisadas e que estão directamente relacionadas com os parâmetros dos motores, nomeadamente com o número de pólos e com o escorregamento (que depende da relação carga/binário resistente) [105].

[1] Do Inglês: *slip factor*

| **As avarias mais típicas** | **Causas prováveis** |
|---|---|
| Marcha trepidante | • carcaça mal fixa<br>• acoplamento mal equilibrado<br>• condutor de alimentação interrompido<br>• corpo estranho no entreferro |
| O motor não arranca | • interrupção da alimentação<br>• as escovas não assentam sobre os anéis<br>• tensão excessivamente baixa<br>• interrupção no arrancador |
| Arranque brusco | • resistência demasiado baixa, no arranque (rotor bobinado)<br>• arrancador parcialmente interrompido ou com contactos queimados<br>• arrancador mal ligado<br>• curto-circuito entre espiras do enrolamento do rotor |
| O motor arranca com dificuldade | • tensão na rede muito baixa<br>• queda de tensão excessiva nos condutores de alimentação<br>• carga excessiva<br>• um terminal do motor polifásico está ligado por erro ao neutro |
| O motor produz um zumbido no arranque | • resistências diferentes no reóstato de arranque<br>• curto-circuito entre espiras do rotor<br>• interrupção num enrolamento do rotor |
| Aquecimento excessivo do motor, em funcionamento | • carga excessiva<br>• tensão demasiado elevada (perdas elevadas no ferro)<br>• tensão demasiado baixa (consumo excessivo de corrente)<br>• condutor de fase partido (consumo excessivo de corrente)<br>• interrupção num dos enrolamentos do estator (consumo excessivo de corrente) |

Tabela 6.3: Avarias típicas num motor de indução [21].

As frequências típicas das correntes onde se deve analisar o espectro são, em concreto, $2.s.f$, $f$, $2.f$, $(1-2.s).f$ e $(1+2.s).f$ [105, 106]. Como ficou claro, estas frequências variam de acordo com o parâmetro $s$, que varia com a carga aplicada ao motor. No caso da análise de vibrações, as frequências são idênticas, embora não semelhantes.

De igual forma, os dados adquiridos por acelerómetro após aplicação da FFT permitem analisar a amplitude, a partir de frequências chave e aludir a quais são os equipamentos com avaria ou sem avaria. Evidentemente que, no caso da corrente, as frequências podem diferir e, normalmente, diferem das frequências analisadas. O anexo D apresenta mais detalhes sobre este assunto.

### 6.2.3 Predição com séries temporais

Com a evolução dos diversos campos da ciência, tais como, a economia, a indústria, a meteorologia e o ambiente, a necessidade de executar previsões para efectuar um planeamento cuidado tornou-se cada vez mais importante.

A previsão da evolução de determinadas variáveis permite melhorar o desempenho de vários sectores da economia, sendo exemplos típicos:

- Previsão de catástrofes ambientais, permitindo minimizar dramas humanos e materiais;
- Maximização dos lucros de investimentos em mercados financeiros e minimização de maus investimentos;
- Previsão de resultados eleitorais;
- Adequação da produção em ambientes industriais de forma a adequar o trabalho aos níveis de procura estimados;
- Previsão da quantidade de potência de produção disponível em função da previsão da velocidade de vento, na produção de energia eléctrica eólica.

Os exemplos multiplicam-se à medida que "se mergulha" nas diferentes áreas de actividade. Muitas técnicas têm sido apresentadas ao longo dos tempos, sendo de reconhecer três grandes metodologias: a utilização de métodos estatísticos, a utilização de modelos específicos adequados ao processo em questão e a utilização de metodologias da área da inteligência artificial.

Apesar dos inúmeros métodos, mais ou menos elaborados, todos apresentam erros na previsão, sendo um dos imperativos minimizar determinadas métricas adjacentes à medida desse erro.

Os métodos mais conhecidos na análise e previsão de uma série temporal [107, 108, 109] são os modelos Auto-Regressivos[2], Médias Móveis[3] e os modelos Auto Regressivos de Médias Móveis[4], bem como variantes, tal como é o caso do método ARIMA [110]. Os métodos propostos

[2] AR-*Auto Regressive*
[3] MA-*Moving Average*
[4] ARMA-*Auto Regressive Moving Average*

por *Box-Jenkis*[5] são dos mais eficientes e utilizados na previsão de séries temporais. Os modelos de alisamento exponencial propostos por *Holt-Winters*[6] na modalidade simples, dupla e tripa, constituem outra das abordagens muito utilizada. Alguns destes modelos têm, normalmente, associados parâmetros que devem ser indicados de forma a minimizar a métrica de erro escolhida, conduzindo usualmente a problemas de optimização não linear [107, 108, 109].

Algumas técnicas mais elaboradas analisam não só a série, mas as séries resultantes da 1ª, 2ª, ... n-ésima derivada, utilizando depois nessas séries resultantes alguns dos modelos já existentes.

As técnicas da área da inteligência artificial [111, 112, 113] mais utilizadas no estudo das séries temporais são os algoritmos genéticos e as redes neuronais, quer redes *feedforward* recorrentes ou com atraso temporal. Nas redes neuronais *feedforward* os sinais dos neurónios são sempre propagados da entrada para a saída e nas redes recorrentes, normalmente, é feita uma realimentação de neurónios posteriores para neurónios anteriores. As redes neuronais com atraso temporal estão perfeitamente adaptadas às séries temporais, dado os valores de entrada da 1ª camada de neurónios serem exactamente os valores da série no instante $t$, $t-1$, $t-2$, ..., $t-n$. Ainda nesta metodologia, alguns autores aplicaram técnicas híbridas, em concreto, a utilização de lógica *fuzzy* e de filtros de *Kalman* para estimar mais rapidamente alguns dos parâmetros da rede neuronal. Ainda na área ligada à inteligência artificial, uma das variações do SVM é aplicada na previsão de séries temporais através de uma técnica de regressão, SVR[7][114].

Na área dos algoritmos genéticos [115, 116, 117, 118, 119], a sua utilização está voltada para a regressão simbólica. Os algoritmos genéticos são em concreto um método de optimização estocástico dada a forma como procuram o máximo/mínimo não depender de fórmulas determinísticas como a informação do gradiente, mas sim inspirado em operadores como a mutação, reprodução e outros que introduzem o grau de aleatoriedade. Nesta técnica, a codificação do problema é um ponto-chave bem como os operadores evolutivos. Não existe necessariamente uma forma clara e intuitiva de determinar se o ponto óptimo foi encontrado. Normalmente, os algoritmos genéticos devem ser utilizados quando se desconhece a estabilidade de outras técnicas de convergência para o ponto óptimo, a não ser que se saiba à partida que essas técnicas são instáveis quando aplicadas ao problema em apreço.

Neste ponto é importante salientar a diferença existente entre previsão de valores futuros e aproximação de uma função aos valores já conhecidos, o que na literatura anglo-saxónica é denominado por *curve-fitting*. Nem sempre um bom algoritmo de aproximação de funções consegue um bom desempenho na previsão de valores futuros.

#### 6.2.3.1 Principais métricas de análise de erros para séries temporais

No anexo C é apresentada a simbologia matemática para alguns dos métodos existentes, dando ênfase a determinados aspectos essenciais à solução final, o que permitirá justificar o método escolhido a utilizar na previsão de avarias.

[5] *Multivariate Models*
[6] *Exponential Smoothing*
[7] SVR-*Support Vector Regression*

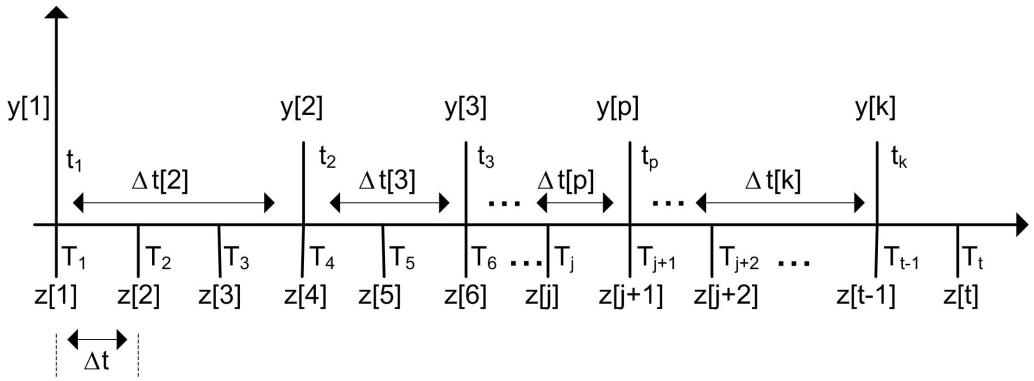



Figura 6.7: Relação entre a série $z[t]$ com amostragem periódica e a série aperiódica $y[k]$.

A notação utilizada $y[k]$ indicará a amostra no instante $k$, sendo $y[1]$ o primeiro elemento da série temporal e $\hat{y}[k]$ refere a notação para a estimação de $y$ calculada no instante $k$, mas pretendendo estimar, ou funcionando como estimador para $y[k+1]$. A notação $y[k]$ ao longo do texto poderá ser abreviada para a notação simplificada $y_k$. Considera-se que as amostras da série temporal são obtidas em intervalos regulares, normalmente associados a uma frequência de amostragem $F_s$.

Quando a série temporal for aperiódica, isto é, com intervalos variáveis entre cada amostra, considerar-se-á uma segunda série que indica os intervalos temporais entre cada amostra, sendo a notação $\Delta t[k] = \Delta t_k = t_k - t_{k-1}$, para $k \geq 1, t_1 = 0$ (considera-se que $\Delta t[1] = \Delta t$). Na situação de intervalos aperiódicos $y[k]$ apresenta-se com um significado diferente do descrito anteriormente, tal como mostra a Fig. 6.7.

Não perdendo de vista o objectivo, salienta-se que o pretendido é obter um estimador que permita prever a evolução futura da série temporal em análise. Assim, salvo seja explicitamente descrito o contrário, considera-se no texto que a palavra estimador significa um método para prever valores futuros da série temporal.

Para avaliar o desempenho da previsão várias métricas podem ser utilizadas para comparar o ajuste e a precisão da previsão [119, 120, 121], tais como:

- MSE - Erro médio quadrático (ver Eq. 6.11);
- RMSE - Raiz do erro médio quadrático (ver Eq. 6.12);
- MAE - Erro médio absoluto (ver Eq. 6.13);
- MAPE - Erro médio percentual absoluto (ver Eq. 6.14);
- CORRE - Coeficiente de correlação de *Pearson* (ver Eq. 6.15);
- TIC - Coeficiente de desigualdade de *Theil* (ver Eq. 6.16).

Considera-se ainda que a métrica é avaliada num intervalo h, imediatamente anterior à última amostra, sendo que ao assumir h=k-1 está-se a avaliar todo o conjunto de previsões desde o seu início. As respectivas expressões matemáticas para estes indicadores são então:

$$MSE = \frac{1}{h} \sum_{i=k-h}^{k-1} (y[i+1] - \hat{y}[i])^2 \quad [120] \tag{6.11}$$

$$RMSE = \sqrt{\frac{1}{h}\sum_{i=k-h}^{k-1}(y[i+1]-\hat{y}[i])^2} \text{ [121]} \tag{6.12}$$

$$MAE = \frac{1}{h}\sum_{i=k-h}^{k-1}|\,y[i+1]-\hat{y}[i]\,| \text{ [120]} \tag{6.13}$$

$$MAPE = \frac{100}{h}\sum_{i=k-h}^{k-1}\frac{|\,y[i+1]-\hat{y}[i]\,|}{y[i+1]} \text{ [120]} \tag{6.14}$$

$$CORRE = \frac{\sum_{i=k-h}^{k-1}(y[i+1]-\overline{y}[i+1])\left(y[i]-\overline{\hat{y}}[i]\right)}{\sqrt{\sum_{i=k-h}^{k-1}(y[i+1]-\overline{y}[i+1])^2 . \sum_{i=k-h}^{k-1}\left(y[i]-\overline{\hat{y}}[i]\right)^2}} \text{ [110, 121]} \tag{6.15}$$

$$TIC = \frac{\sqrt{\frac{1}{h}\sum_{i=k-h}^{k-1}(y[i+1]-\hat{y}[i])^2}}{\sqrt{\frac{1}{h}\sum_{i=k-h}^{k-1}y[i+1]^2}+\sqrt{\frac{1}{h}\sum_{i=k-h}^{k-1}\hat{y}[i]^2}},\ 0 \le TIC \le 1;\ e\ 0\ indica\ igualdade \text{ [120]} \tag{6.16}$$

#### 6.2.3.2 Principais métodos de predição

Os métodos expostos neste capítulo encontram-se detalhados no anexo C. No entanto, destacam-se os métodos apresentados com valor científico acrescentado, nomeadamente, o "alisamento exponencial" modificado, com adaptação do parâmetro $\alpha$ (ver secção C.1.4.2, pág. 210). O método de base do alisamento é conduzido pela seguinte equação:

$$\hat{y}[k] = \alpha.y[k] + (1-\alpha).\hat{y}[k-1] \tag{6.17}$$

onde normalmente o valor de $\alpha$ é fixo e a proposta neste trabalho passou por utilizar uma fórmula de adaptação a cada amostra, dada por (a justificação encontra-se na secção C.1.4.2, pág. 210):

$$\alpha = \frac{\sum_{i=n}^{k}(y[i+1]-\hat{y}[i-1]).(y[i]-\hat{y}[i-1])}{\sum_{i=n}^{k}(y[i]-\hat{y}[i-1])^2},\ n \ge 2 \tag{6.18}$$

Outro dos métodos, o algoritmo relativo à série de Taylor, também é uma proposta inovadora, no entanto, os resultados estão demasiado dependentes do tipo de dados da série, tornando este método mais sensível que o anterior (ver secção C.1.4.6, pág. 218).

O objectivo de qualquer dos métodos expostos é calcular $\hat{y}[k+m]$, para $m \ge 0$, com um grau de confiança aceitável.

## 6.3 Métodos de predição baseados em séries temporais aplicados à manutenção

Um dos objectivos desta dissertação é o seguinte:

"...prever o instante temporal onde se terá uma avaria[8] provável. Com esta informação lançar uma ordem de trabalhos de manutenção preventiva no *software* SMIT de modo a que, oportunamente, as causas que no futuro darão ensejo a avarias sejam corrigidas antes de acontecerem..."

A resposta a esta questão convenciona em primeira mão a continuidade de variáveis chave que devem ser observadas, as quais, em cada instante, fornecem a informação do "estado de saúde" dos equipamentos.

Por analogia com a vida humana e, retirando raras excepções, nenhum indivíduo passa instantaneamente de um estado de saúde óptimo para um estado de óbito (excepto acidentes). Existem uma série de exames médicos que permitem antever as patologias graves.

Mesmo em casos extremos como a exposição a agentes patológicos danosos para a vida humana, por exemplo, vírus, armas químicas ou, no caso de óbito por afogamento, provavelmente, um sistema de monitorização com alguma variável mensurável do corpo humano permitiria antever o que iria acontecer. No caso de afogamento, a monitorização da quantidade de oxigénio admitida pelo sistema pulmonar, quantidade de água absorvida ou o ritmo cardíaco, permitiria prever o falecimento antes deste se concretizar.

O que se pretende assim enfatizar com esta afirmação, é indicar o facto de existir uma probabilidade elevada e expectável dos sistemas a monitorizar manterem uma certa continuidade das variáveis monitorizadas, isto é, provavelmente serão funções sem descontinuidades. De igual modo, é também previsível que, se determinada funcionalidade dum equipamento começa a entrar num estado de deterioração, alguma variável mensurável do equipamento divirja dos valores nominais, encetando uma subida ou descida de forma contínua.

O exposto no parágrafo anterior é, supostamente, maioritariamente verdadeiro. Incorporando no entanto o resultado da intervenção da manutenção (com ou sem substituição de componentes avariados), após a causa da anomalia se encontrar corrigida os sinais monitorizados entrarão novamente dentro da gama de valores toleráveis. Este comportamento, estudado a nível da série temporal dessa variável, poderá indicar a existência de sazonalidades cujo período estará necessária e directamente relacionado com o tempo médio entre avarias. A outra hipótese provável de interpretação é a existência de uma descontinuidade ao nível do parâmetro avaliado, entre os instantes de pré-manutenção e de pós-manutenção.

Para ultrapassar este problema, se o tempo médio entre avarias é conhecido, qualquer um dos algoritmos apresentados pode ser utilizado sem grandes preocupações, **nomeadamente a nível da estimação do período da sazonalidade** (em vez de tentar estimar o comprimento da sazonalidade), caso contrário, depois de cada intervenção de manutenção, a série temporal será reiniciada novamente. Este último procedimento torna desnecessário medir com rigor o tamanho do período sazonal e na prática, até o removerá. Outra das particularidades do algoritmo é a necessidade de, para cada sinal, indicar os intervalos de tolerância, nomeadamente, intervalo de aviso e o intervalo crítico. Sempre que a análise da série temporal previr uma saída fora de um destes intervalos,

[8] Restringe-se o espectro das avarias ao domínio do estudo realizado, não sendo possível considerar-se qualquer tipo de avaria, pois depende de estudos mais aprofundados nos vários componentes de um aero-gerador.

lançar-se-á um aviso ao operador (pode ser o sistema informático a lançar uma Ordem de Trabalho). O caminho seguido neste trabalho, em qualquer dos algoritmos de análise da sequência temporal dos dados, passa pela reinicialização sempre que acontece uma intervenção.

Em todos os algoritmos expostos no anexo C, excepto no algoritmo da rede neuronal, é apresentada uma expressão para a estimação, constituída por uma fórmula fechada. Especificamente[9], tem-se:

$$\hat{y}[k+m] = f(\hat{y}[k], m, \cdots)$$

sendo o objectivo calcular **m** para o qual $\hat{y}[k+m] \geq Limite\{máximo\}$ ou $\hat{y}[k+m] \leq Limite\{mínimo\}$.

Interessa neste ponto explicitar como determinar o tempo até à avaria com base na aquisição dos dados de campo e dos resultados da série temporal.

De acordo com o escrito acima, a previsão pode ser efectivada para todos os valores de $m$ desde $0 \cdots m$, com $m \geq 0$, isto é, calculando a respectiva previsão nesses instantes iniciando em $\hat{y}[k+0]$, $\hat{y}[k+1]$,..,$\hat{y}[k+ponto]$,...,$\hat{y}[k+m]$ e, caso alguma das previsões $\hat{y}[k+ponto]$ esteja fora do intervalo de tolerância, obtêm-se o tempo desejado (não se calculam os restantes). No entanto, conforme é compreensível, este é um método de força cega.

Existem outras possibilidades viáveis, especificamente com o objectivo de diminuir o número de cálculos necessários. Um dos métodos consiste em dividir o intervalo $[0 \cdots m]$ a meio[10] (ponto $inteiro\left(\frac{m-0+1}{2}\right)$) e, de seguida, calcular a previsão nesse instante e nos limites do intervalo.

Se o valor prognosticado está fora da tolerância, o intervalo válido é o da esquerda $[0 \cdots \frac{m-0+1}{2}]$, caso contrário o outro, e assim sucessivamente até obter-se um sub-intervalo de tamanho um. Para explicitar melhor, considere-se um intervalo genérico $[l_{inferior} \cdots l_{superior}]$, ao se dividir a meio temos como ponto central $p_{central} = inteiro\left(\frac{l_{superior}-l_{inferior}+1}{2}\right)$, alcançando-se os dois intervalos $[l_{inferior} \cdots p_{central}[$ e $[p_{central} \cdots l_{superior}]$. As condições de funcionamento deste método de pesquisa obedecem às seguintes regras ordenadas (admitindo um bom comportamento da função):

- O cálculo da função no limite inferior do intervalo $\hat{y}[k+l_{inferior}]$, deu fora da tolerância: deve-se parar (só acontece para $l_{inferior} = 0$);
- $\hat{y}[k+l_{inferior}]$, $\hat{y}[k+l_{superior}]$ e $\hat{y}[k+p_{central}]$ deram dentro da tolerância: deve-se parar (só acontece no intervalo $[0 \cdots m]$);
- O cálculo da função no ponto central do intervalo $\hat{y}[k+p_{central}]$ deu **dentro** da tolerância: utilizar o intervalo superior como novo intervalo $[p_{central} \cdots l_{superior}]$, aplicando o mesmo raciocínio;
- O cálculo da função no ponto central do intervalo $\hat{y}[k+p_{central}]$ deu **fora** da tolerância: utilizar o intervalo inferior como novo intervalo $[l_{inferior} \cdots p_{central}]$, aplicando o mesmo raciocínio;

[9] Como $\hat{y}[k]$ é a previsão mais recente e $m$ o número de passos à frente pretendido, provavelmente $\hat{y}[k+m]$ poderá depender destes valores,..., embora não seja obrigatório, considerou-se assim por opção.

[10] Equivalente ao algoritmo de pesquisa binária, num vector ordenado, onde se visita o elemento do meio e se decide qual dos meios intervalos é inválido, de seguida, aplica-se sucessivamente o mesmo conceito ao intervalo válido, até obter-se um intervalo de um elemento.

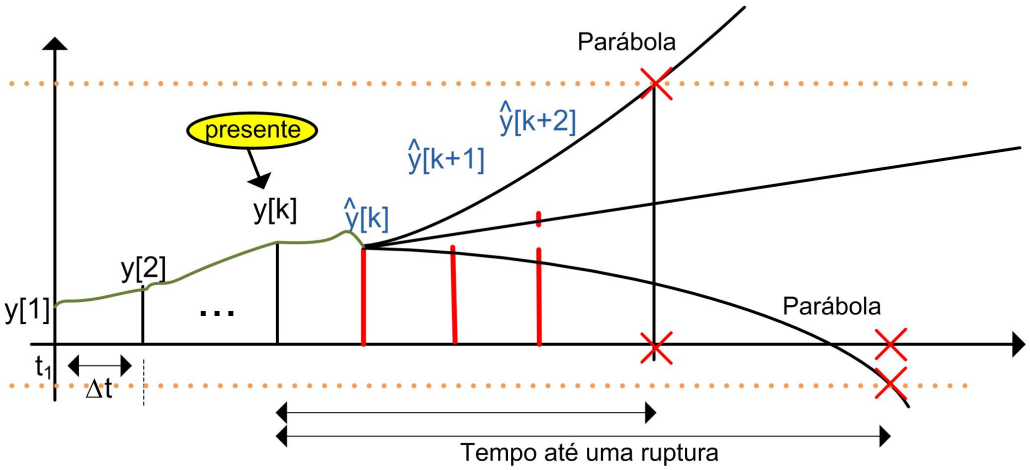



Figura 6.8: Interpolação parabólica para aumentar a fiabilidade da estimação do tempo para avaria.

- O cálculo da função no limite superior do intervalo $\hat{y}[k+l_{superior}]$ deu **dentro** da tolerância: deve-se parar (só acontece para $l_{superior} = m$);
- O cálculo da função no limite superior do intervalo $\hat{y}[k+l_{superior}]$ deu **fora** da tolerância: utilizar o intervalo superior como novo intervalo $[p_{central} \cdots l_{superior}]$, aplicando o mesmo raciocínio.

Há ainda um aspecto relevante a considerar, pois à medida que se aumenta o número de passos da previsão para os instantes seguintes, a fiabilidade do valor previsto decresce, isto é, torna-se menos confiável. Assim, uma das possibilidades de ultrapassar este problema passa pela seguinte proposta de desenvolvimento futuro: utilizar apenas a previsão dos três instantes consecutivos, nomeadamente, para $k$, $k+1$, $k+2$; em seguida interpolar estes pontos estimando uma parábola. No próximo passo determina-se a intercepção da parábola com os limites do intervalo e, de acordo com a distância ao instante actual, procede-se em conformidade. Este processo pode ser visualizado graficamente na Fig. 6.8 e tem como vantagem uma maior fiabilidade[11], bem como baixa drasticamente o número de cálculos no caso dum $m$ elevado.

Relativamente ao aumento da fiabilidade, o anexo C, na secção C.2 (pág. 219) apresenta algumas hipóteses de combinar os diferentes métodos de previsão, dado saber-se, à partida, que nenhum método tem características de "acertar" na previsão, independentemente dos dados. Depois do estudo realizado e da proposta anteriormente descrita no anexo C sobre os vários métodos de previsão, a questão que se coloca é a seguinte: Qual será o melhor processo para ser utilizado em determinada circunstância? As propostas apresentadas nesta dissertação como resposta a esta questão são as seguintes:

- **Usar apenas um método** - Utilizar o método do alisamento exponencial modificado apresentado, cujo $\alpha$ é adaptado a cada amostra - caso da previsão incluída nos sistemas embebidos, tipicamente para equipamentos pouco críticos;

[11] Por estimar pontos futuros mais próximos

- **Combinar dois a três métodos** - Utilizar vários métodos pesados de acordo com os indicadores de desempenho descritos na secção C.2 - para equipamentos de criticidade média;
- **Utilizar o melhor método a cada momento** - Para equipamentos críticos e com disponibilidade de poder de cálculo, para a mesma série utilizar vários métodos, validando o resultado do "melhor" método de acordo com o MSE, sendo o "melhor" determinado por uma característica de continuidade de estabilidade nas últimas $j$ amostras.

O factor $\Delta t$ (ver Fig. 6.8) não foi discutido, mas o seu valor deverá ser o mais "curto" possível, no mínimo dar a hipótese de existirem 100 amostras dentro de um intervalo de avaria - valor directamente relacionado com o tempo médio entre avarias do equipamento.

Outra das discussões que não foi feita e não será abordada nesta dissertação, é a determinação dos limites/intervalos de tolerância, dado que para diferentes equipamentos, mesmo dentro da mesma função, podem existir diferentes valores de tolerância. Em muitos casos, os fabricantes indicam valores nominais e tolerâncias de funcionamento; assim sendo, o problema estará resolvido à partida.

Outra das metodologias possíveis de experimentar é a de um processo genérico, que pode ser utilizado em qualquer área envolvente da manutenção e não exclusivamente nos geradores eólicos. Assim, considerando o *software* SMIT, onde são registadas as Ordens de Trabalho, quer planeadas quer não planeadas, aquela informação pode ser utilizada para a construção de uma série temporal. A abcissa da série será o número de horas de intervenção/paragem nos equipamentos, condicionada a determinada causa da avaria, dado que serão consideradas as causas das avarias independentes. Para uma correcta utilização, os operadores do SMIT devem classificar correctamente as causas das avarias nos diferentes equipamentos.

De notar a diferença entre os critérios de horas de intervenção e horas de paragem, condicionadas à origem da causa.

A série, para cada motivo de paragem, é construída mostrando os dias em que ocorreram falhas e quantas horas o equipamento permaneceu inoperante. Os modelos, neste caso e numa primeira análise não muito aprofundada, deveriam ser meramente estatísticos, nomeadamente através da utilização da distribuição de *Weibull* para caracterizar a fiabilidade dos equipamentos, apoiada no cálculo de probabilidade de avaria. No entanto, esta abordagem foge do âmbito da presente dissertação por necessitar de um histórico com uma dimensão temporal considerável.

## 6.4 Conclusões

Este capítulo apresentou os principais algoritmos utilizados para implementar métodos de detecção de avarias de forma preditiva.

O capítulo iniciou com a apresentação de dados de fiabilidade sobre dois conjuntos de aerogeradores, um da Alemanha e outro da Dinamarca. Estes dados permitem adquirir a percepção das avarias mais repetitivas e de quais os componentes com maior índice de falhas. Os dados apresentados foram obtidos através de referências bibliográficas.

De seguida é apresentado um método genérico para classificar a condição de funcionamento com base em técnicas de inteligência artificial, nomeadamente classificação baseada no algoritmo SVM, tendo como base dados como potência instantânea, velocidade de rotação dos veios, potência activa produzida, etc. O método foi validado com dados reais de aero-geradores do norte do país, representados nas Figs. 8.8 e 8.9 (pág. 133).

Outra metodologia apresentada, prende-se com o estudo da análise de vibrações de motores de indução industriais, com o intuito de determinar quais as metodologias indicadas para a determinação do instante de avaria no futuro. Este estudo é meramente indicativo dada a indisponibilidade de recolher informação de um gerador eléctrico de um aero-gerador. No entanto, exceptuando o aspecto da dimensão em termos mecânicos, permite algumas analogias.

Por último o capítulo expõe um estudo relacionado com a previsão de tendências utilizando séries temporais. Genericamente a previsão com séries temporais pode ser aplicada à informação sensorial recolhida através de um sensor.

O capítulo apresenta os temas de uma forma geral e os detalhes são explanados nos anexos C e D e os resultados apresentados no capítulo 8.

# Capítulo 7

# Modelos de planeamento da manutenção

Neste capítulo é abordada a problemática dos modelos de planeamento da manutenção. O SMIT tem incorporado em cada plano de manutenção o controlo de uma variável de funcionamento que é, por exemplo, o número de peças manufacturadas, o número de quilómetros percorridos, etc, o que permite estimar datas aproximadas para a realização da próxima intervenção planeada. No entanto, este método, apenas permite decidir datas de manutenção relativamente a um Objecto de Manutenção (OM); contudo, também pode ser feita a indexação de um plano de manutenção a um segundo, garantindo, por exemplo, a simultaneidade para a realização dos trabalhos nos dois OMs ou em qualquer número destes. Esta opção torna-se particularmente pertinente em situações onde haja interdependências entre vários equipamentos, tais como numa linha de produção ou numa célula de fabrico.

O problema fica ainda mais complexo se a intenção for a optimização global da gestão da manutenção, nomeadamente contabilizando as perdas de produção e os custos de deslocações, para além dos custos directos da manutenção.

Na próxima secção aborda-se o algoritmo de planeamento da manutenção para um OM e, na secção seguinte, é apresentada uma estratégia que visa a optimização de uma solução global.

## 7.1 Algoritmo para planeamento da manutenção

O algoritmo apresentado foi proposto inicialmente em [68], para resolver o problema da predição em intervalos irregulares de valores observados.

Reportando à Fig. 6.7 (pág. 99) para observação dos valores da série $y[k]$ em intervalos irregulares e, devido a este facto, deve-se ainda considerar a série dos instantes temporais $t[k]$ nos quais são obtidos os valores do parâmetro de controlo $y[k]$. Assim, a proposta para o cálculo de $\hat{y}[k]$ passa pela indicação inicial do valor médio e do valor do desvio padrão, os quais vão sendo

actualizados após cada leitura da seguinte forma:

$$\begin{gathered}
\Delta y[k] = y[k] - y[k-1] \\
\Delta t[k] = t[k] - t[k-1] \\
\overline{y}[k] = \frac{\Delta y[k]}{\Delta t[k]} \\
\overline{\hat{y}}[k] = \overline{y}[k] + (1-a)^{\Delta t[k]} \left(\overline{\hat{y}}[k-1] - \overline{y}[k]\right) \\
t[k+1] \approx \hat{\Delta t}[k] = \frac{Incremento}{\overline{\hat{y}}[k]} \\
y[k+1] \approx Incremento + y[k],\ acontecer\ no\ instante\ t[k+1]
\end{gathered} \tag{7.1}$$

Estas equações consideram a sequência $y[k]$ monótona crescente ou decrescente, o que, de facto, acontece na generalidade das situações. Exemplos disso são os seguintes: número de horas de funcionamento; número de quilómetros percorridos; número de peças fabricadas; evolução do desgaste de determinado componente, etc. O valor é crescente, decrescente ou, ainda, mantém-se constante por falta de utilização do equipamento. Assim, a próxima intervenção deve ser realizada exactamente no instante

$$t[k+1] \approx \hat{t}[k] = t[k] + \hat{\Delta t}[k] \tag{7.2}$$

Na Eq. 7.1 o factor *Incremento* é um dado suprido e indica, por exemplo, para o caso do indicador "quilómetros percorridos", qual o intervalo de quilómetros entre revisões. Outros dados fornecidos à partida são a média e o período de tempo no qual se prevê que essa média aconteça, bem como o desvio padrão (ver Fig. A.8, pág. 156).

Para aferir da qualidade da estimativa pode-se ainda estimar o desvio padrão, utilizando:

$$\begin{gathered}
s^2[k] = \Delta t[k] \left(\overline{\hat{y}}[k] - \overline{y}[k-1]\right)^2 \\
\hat{s}^2[k] = s^2[k] + (1-b)^{\Delta t[k]} \left(\hat{s}^2[k-1] - s^2[k]\right) \\
s = \sqrt[2]{\hat{s}^2[k]}
\end{gathered} \tag{7.3}$$

O processo descrito nas Eq. 7.1 e 7.3 utiliza dois alisamentos exponenciais de constantes $a$ e $b$, para estimar a média e o desvio padrão da série aperiódica [68].

A alteração introduzida neste algoritmo tem como objectivo permitir recalcular a previsão no caso de, entre dois instantes consecutivos da série $y[k]$, serem observados valores que não estavam previstos. No exemplo das manutenções dos automóveis com parâmetro de controlo dos quilómetros, equivale a afirmar que o dono do automóvel, com alguma regularidade, comunica à oficina o número de quilómetros do veículo e esta, em função desse valor mais recente, reactualiza a data da próxima revisão. Este é um exemplo facilmente aplicável a uma empresa de transportes que possua uma frota quer para transporte de mercadorias quer de passageiros.

Para realizar a alteração, considera-se que nos instante $t[k]$ são realizadas as intervenções, com a respectiva Ordem de Trabalhos (OT) sendo, portanto, nessa ocasião que o valor de $y[k]$ é oficialmente registado. Assim, examinando o instante $t[k]$ no qual foi estimado o valor de $\hat{t}[k]$ e, por consequência, o próximo instante $t[k+1]$ para a intervenção. Considera-se agora que entre

estes dois intervalos são realizadas leituras sem, no entanto, ser criada nenhuma OT. A pergunta é a seguinte: como se devem actualizar os cálculos à luz das Eq. 7.1, 7.2 e 7.3 ?

A resposta é simples:

- Se $y(t[k]+\Delta t) \geq y[k]+Incremento$, então o valor lido ultrapassou o limite esperado, o que significa a necessidade imediata de uma OT planeada;
- Se $y(t[k]+\Delta t) < y[k]+Incremento$, então utilizam-se as Eq. 7.1 aplicadas aos extremos $(y[k],\ t[k])$ e $(y(t[k]+\Delta t),\ t[k]+\Delta t)$, actualizando a nova estimativa do instante temporal onde deve ser efectuada uma nova acção planeada - data da OT planeada (estimativa de $\hat{\Delta}t[k]$).

Este procedimento pode ser repetido *n* vezes antes da criação da OT afecta ao plano em questão e, portanto, significa poderem ser indicados *n* valores intermédios do parâmetro de controlo, cujo objectivo e existência ajuda a melhorar a previsão efectiva da próxima intervenção planeada. Um dos exemplos de uma situação desta natureza é o caso de um sistema de manutenção interligado com um sistema de produção onde os contadores afectos à produção de cada equipamento são importantes para melhorar a estimativa da intervenção planeada nesses mesmos equipamentos.

## 7.2 Optimização dos percursos entre torres eólicas

O problema da optimização dos trajectos entre os diferentes parques eólicos é importante, embora possa não ser muito relevante no caso de um número baixo de parques. Conhecendo a localização dos parques, ou de cada aero-gerador, pretende-se optimizar as distâncias percorridas, tendo como condicionantes as seguintes variáveis: custos, nomeadamente, as perdas de produção de energia; velocidade do vento estimada para determinado instante; disponibilidade de equipamentos caros de manutenção; tempos de reparação; tempos de percurso; etc. Há ainda um outro limite, que é o número de vias alternativas ao trajecto, o qual é limitado entre os diferentes parques quando estes estão instalados em serras. O oposto verifica-se num parque marítimo onde por via aérea a deslocação pode ser em linha recta ou, no caso de via marítima, dependendo igualmente da navegabilidade dos percursos e das respectivas rotas.

Uma forma actual de resolver um problema semelhante ao precedente é utilizada nos programas de cálculo de rotas de tráfego, nomeadamente os GPS portáteis, *AutoRoute*, *GoogleMaps*, etc, condicionado à função de custo distância e/ou tempo de viagem. O cálculo do melhor trajecto entre dois pontos, ou vários pontos, é desde sempre um dos desafios da área da optimização. Na literatura especializada, este problema aparece descrito como o *Travel Salesman Problem*[1], que consiste em calcular o trajecto óptimo de modo a minimizar a distância percorrida ao visitar todos os clientes, uma única vez, voltando ao ponto inicial.

No caso dos parques eólicos, muitas das vias de comunicação não se encontram registadas nas aplicações de referenciação geográfica supracitadas, logo não existe à partida uma ferramenta de base para determinar as distâncias entre dois pontos e o tempo necessário para efectuar esse

[1] *Travel Salesman Problem*- Problema do caixeiro viajante, [122].

percurso. Um dos primeiros passos poderá passar por fazer este levantamento em cada parque, registando os pontos passíveis de admitirem acções de manutenção.

O método proposto nesta secção visa ultrapassar estas duas limitações e é exposto como um dos caminhos de investigação a desenvolver no futuro.

A proposta passa por referenciar cada ponto a visitar pelas coordenadas GPS, determinando pontos obrigatórios de passagem, coincidentes com intercepções de estradas, até uma infra-estrutura de acesso registada em mapas digitais, por exemplo uma estrada Nacional ou Camarária. Em alternativa, registar o número de quilómetros e o tempo necessário entre dois nós de um trajecto.

Cada um dos pontos recolhidos pode ser catalogado em dois grupos distintos:

$$\begin{gathered} G = \{(latitude_1, longitude_1) \cdots (latitude_i, longitude_i)\} \\ com\ (latitude_i, longitude_i)\ \in\ \{posição\ dos\ aero-geradores\} \end{gathered} \tag{7.4}$$

e o conjunto

$$\begin{gathered} C = \{(latitude_1, longitude_1) \cdots (latitude_i, longitude_i)\} \\ com\ (latitude_i, longitude_i)\ \in\ \{ponto\ de\ um\ caminho\ entre\ dois\ aero-geradores\} \end{gathered} \tag{7.5}$$

onde o conjunto $C$ (caminhos) representa as restrições impostas aos elementos do conjunto $G$ (aero-geradores), para viajar entre dois elementos.

De seguida, a cada aero-gerador associam-se os custos integrados na métrica a minimizar - a função objectivo ou função de custo.

Finalmente, o aspecto mais complexo é a escolha de um algoritmo de optimização para decidir sobre a sequência do trajecto e dos trabalhos de manutenção dos aero-geradores. Muitos dos algoritmos disponíveis na literatura como: *ACO theory* [2], algoritmos genéticos, optimização linear, ..., pretendem resolver o problema da procura de soluções óptimas num elevado conjunto de hipóteses de soluções, com a métrica associada a uma função de custo.

Não sendo o principal objecto desta dissertação a análise de vários métodos e o estudo comparativo do seu desempenho, optou-se por reformular o actual problema à luz dos algoritmos de optimização baseado no comportamento da formiga e nos algoritmos genéticos. Evidentemente, cada um destes métodos usa heurísticas de procura aleatória em espaço de estados de elevada dimensão e, segundo alguns autores, a sua utilização deve ser concretizada quando mais nenhum outro método resolve o problema em tempo útil.

Importa agora salientar o grau de dificuldade do problema, pois ainda não foram introduzidas as restrições, nomeadamente, o número de equipas de técnicos, o número de dias de reparação por equipa, o número de equipas possíveis em cada uma das reparações, o local onde pernoitam as equipas caso seja necessário, o número de horas de produção estimadas por dia para os próximos

[2] *Ant colony optimization theory* - Teoria associada ao comportamento das colónias de formigas, segundo a qual os caminhos mais curtos para o alimento são encontrados através da consistência de propriedade químicas de *pheromones* marcando o trajecto. As propriedades da substância química evaporam ao longo do tempo de acordo com uma lei exponencial. O seu primeiro proponente foi Marco Dorigo em 1992 [123] na sua tese de doutoramento.

dias, a potência de cada aero-gerador, o custo do trajecto, etc. Estes são apenas alguns dos factores influenciadores da solução óptima, apesar de alguns poderem ser considerados complementares, tais como, por exemplo, o número de dias de reparação e o número de equipas possíveis em cada uma das reparações. É evidente que alguns deles podem ser de magnitude ínfima quando comparados com outros e, portanto, podem ser desprezáveis para obtenção da solução final.

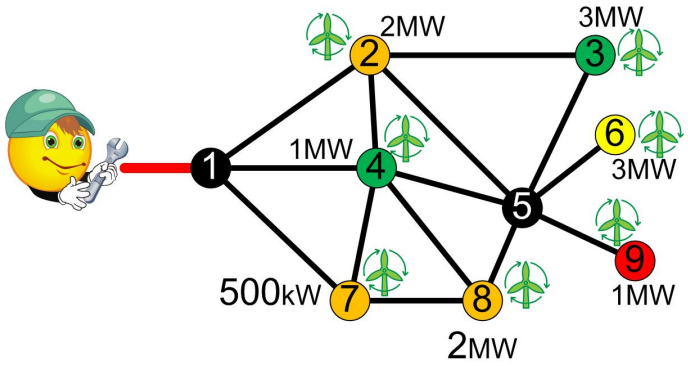



Figura 7.1: Exemplo gráfico de um problema da optimização de percursos. O exemplo é apenas para fins de simulação.

A Fig. 7.1 apresenta um exemplo possível de uma configuração com indicação das respectivas potências nominais de cada aero-gerador, onde se têm representados dois pontos de passagem (preto), 2 aero-geradores em bom estado (verde), um avariado (vermelho), três para avariar em pouco tempo (amarelo escuro) e outro para avariar com um prazo mais longo (amarelo claro). Supondo que se têm duas equipas de manutenção que pernoitam no $\text{Nó}_1$ qual deverá ser a sequência de manutenção dos aero-geradores?

Evidentemente que a solução para o problema da Fig. 7.1 pode necessitar de várias visitas ao mesmo nó, dependendo do número de dias necessários para a reparação. Assim, por exemplo, com a condição de 2 dias de reparação do $\text{Nó}_9$, uma das soluções poderia ter o seguinte trajecto: $\text{Nó}_1$, $\text{Nó}_4$, $\text{Nó}_5$, $\text{Nó}_9$, Trabalhos, $\text{Nó}_5$, $\text{Nó}_4$, $\text{Nó}_1$, Pernoitar, $\text{Nó}_1$, $\text{Nó}_4$, $\text{Nó}_5$, $\text{Nó}_9$, Trabalhos, $\text{Nó}_5$, $\text{Nó}_4$, $\text{Nó}_1$, Pernoitar, $\text{Nó}_1$, $\text{Nó}_7$, Trabalhos,...

Este aspecto realça a diferença entre este problema e o problema do caixeiro-viajante, onde cada nó é visitado apenas uma vez, enquanto neste problema um nó pode ser visitado várias vezes ou nenhuma.

Na literatura da programação linear [124], este tipo de problemas são designados como problemas de afectação de pessoal a tarefas, podendo ser entendido como um problema de optimização combinatória. A sua resolução apresenta-se no entanto por vezes difícil segundo este ponto de vista.

### 7.2.1 Formulação da função de custo

Como não poderia deixar de ser, a função de custo é expressa em euros ou, mais genericamente, em Unidades Monetárias (UM) e representa o custo financeiro associado à perda de produção

de energia eléctrica, ao custo da mão-de-obra e da sua logística, nomeadamente deslocações e pernoites.

Assim, o custo financeiro associado a um nó será então dado por:

$$custoNó = perdaEurosPorAvaria + perdaEurosReparar + custoEurosDePernoitar \tag{7.6}$$

O custo das deslocações está associado a um trajecto e, portanto, a um nó origem e a um destino. Assim, genericamente, o custo associado a um trajecto entre o nó $i$ e o nó $j$ é dado por:

$$custoDoTrajecto_{i,j} = custoViagemNó_{i,j} + custoTécnicosNó_{i,j} \tag{7.7}$$

**Parâmetros associados ao custo do nó**

Para calcular a Eq. 7.6, o potencial financeiro diário de um determinado nó, concretamente a sua contribuição no caso de se encontrar avariado e não produzir, é dado por:

$$perdaEurosPorAvaria = potencialFinanceiro = Pgerador \times nHProdPorDia \times eurosPorMW \tag{7.8}$$

Pode-se, de seguida, calcular a perda no caso de determinado nó estar em manutenção:

$$perdaEurosReparar = \begin{cases} potencialFinanceiro \times taxaPerdaPorReparar, \text{ se nó a reparar} \\ 0, \text{ noutros casos} \end{cases} \tag{7.9}$$

O nó só se encontra no entanto em reparação se tiver sido alocada uma equipa de técnicos para realizar os trabalhos, e esta só pode ser alocada caso se verifique a condição seguinte: $nDiasPraAvaria \leq nDiasFiaveis$ ou $nDiasReparacao > 0$.

Associado a um nó existe ainda o custo financeiro de pernoitar nesse ponto,

$$custoEurosDePernoitar = \begin{cases} nósAPernoitar \times eurosPorNoite, \text{ se a equipa pernoita} \\ 0, \text{ noutros casos} \end{cases} \tag{7.10}$$

Para estimar estes factores tem que se entrar em consideração com os seguintes parâmetros (deverão ser fornecidos à entrada do algoritmo):

- *Pgerador* - **Potência máxima disponível num aero-gerador** - pode ser a potência nominal do aero-gerador ou o máximo disponível em caso de este se encontrar numa situação de corte de potência;
- *nHProdPorDia* - **Número de horas produção por dia** - é uma estimativa baseada na antevisão da velocidade do vento. Neste item considera-se a existência de dados de previsão

para os cinco dias consecutivos; a partir do sexto dia é usado um valor fixo, podendo ser, por exemplo, a média verificada em anos anteriores para a época do ano em questão;

- *eurosPorMW* - **Preço do MW produzido** - custo do MW produzido. Neste item são indicados dois valores, o preço estimado para o dia seguinte e a média do valor pago utilizada nos restantes dias;
- *nDiasFiaveis* - **Janela, em dias de previsão, para avaria** - número máximo de dias para o qual é possível prever com confiança o aparecimento de uma avaria (discutido noutros capítulos desta dissertação, em concreto no 6);
- *nDiasPraAvaria* - **Número de dias para avaria** - estimação do número de dias que o aero-gerador pode produzir sem avariar. Findo este número de dias considera-se o aero-gerador avariado e, portanto, sem capacidade de produção. Não é incluída a hipótese de, depois de avariado, ainda poder produzir residualmente. Em caso de avaria $nDiasPraAvaria = 0$. Quando não subsistir avaria prevista ou a estimativa for superior a *nDiasFiaveis*, será indicado o valor $nDiasPraAvaria = nDiasFiaveis * 2$ ou $nDiasPraAvaria = Infinito$ caso seja representável (para diferenciar entre os valores do intervalo [0;*nDiasFiaveis*]);
- *nDiasReparacao* - **Número de dias de reparação** - para a avaria estimada, quantos dias são necessários para efectuar a reparação, por exemplo, por implicar paragem do aero-gerador;
- *taxaPerdaPorReparar* - **Factor de perda quando em reparação** - percentagem de perda de produção durante os trabalhos de reparação;
- *nTecnicos* - **Número de técnicos/equipas disponíveis** - número de equipas disponíveis em simultâneo para realizar os trabalhos. Não se considera possível duas ou mais equipas poderem trabalhar em simultâneo no mesmo aero-gerador. Cada equipa tem transporte autónomo;
- *nósAPernoitar* - **Nós onde se pernoita** - locais onde se pode pernoitar sempre que a duração dos trabalhos é superior a um dia. Caso se possa pernoitar, é atribuído o valor um, caso contrário o valor zero. Este factor pode obrigar a deslocações das equipas;
- *eurosPorNoite* - **Custo por noite** - preço por noite por equipa num determinado nó.

**Parâmetros associados ao custo das deslocações**

Para calcular a Eq. 7.7, define-se o custo de viagem e o custo da mão-de-obra dos técnicos como:

$$\begin{aligned} custoViagemNó_{i,j} &= eurosPorKm_{i,j} \times nKmViagem_{i,j} \\ custoTécnicosNó_{i,j} &= nDiasViagem_{i,j} \times eurosPagoDiaViagem_{i,j} \end{aligned} \tag{7.11}$$

Nas deslocações optou-se por considerar que cada equipa tem um meio de transporte autónomo. Para estimar os custos das deslocações entra-se em consideração com os seguintes itens:

- *eurosPorKm* - **Preço por quilómetro** - preço por quilómetro de deslocação num determinado troço (entre dois nós adjacentes);
- *nKmViagem* - **Número de quilómetros** - indica o número de quilómetros num troço (entre dois nós adjacentes);

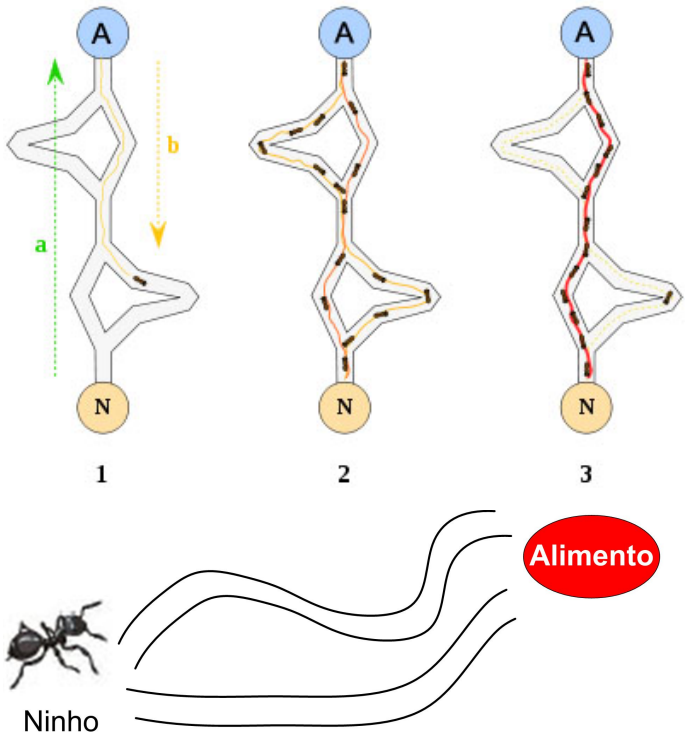



Figura 7.2: Funcionamento do algoritmo de pesquisa da Formiga (adaptado de [19]).

- *nDiasViagem* - **Número de dias de viagem** - indica o número de dias de viagem entre dois nós adjacentes;
- *eurosPagoDiaViagem* - **Custo pago às equipas técnicas pela viagem** - entre dois nós adjacentes de um troço, indica quanto custa a viagem por dia.

### 7.2.2 Inspiração no algoritmo das formigas

A Fig. 7.2 apresenta os passos básicos observados na natureza com esta espécie e mostra que, após algum tempo, as formigas conseguem encontrar o caminho mais curto. Assim, num primeiro passo, uma formiga especial, aleatoriamente, procura alimento, achando um trajecto entre o ninho e a comida (situação *a*). Quando encontram o alimento, voltam ao ninho pelo caminho mais curto, largando produtos químicos para marcar o trajecto (situação *b*). Caso existam dois caminhos entre o ninho e a comida (passo 2), como o produto químico tem um tempo de vida curto, no caminho mais longo o seu rasto terá tendência a desaparecer. A quantidade de formigas que faz o trajecto também influencia na longevidade do efeito do produto químico deixado no trajecto sempre que voltam ao ninho (passo 3). Numa colónia de formigas há pelo menos dois tipos de tarefas, que são o de recolher comida e o de procurar comida. Este sistema baseia-se no *feedback* positivo (o depósito do produto químico atrai outras formigas que reforçarão o trilho) e negativo (dissipação do trilho por evaporação que impede o sistema de ficar caótico).

Existe uma variedade de problemas onde este algoritmo pode ser aplicado, nomeadamente na análise de imagens através da correspondência de contornos de silhuetas de objectos, pessoas, etc. A sua aplicação estende-se à análise de imagens médicas [125].

No sector da optimização da topologia da rede eléctrica vários autores têm utilizado este género de algoritmos, procurando configurações minimizadoras da perda de energia nos vários troços e de melhoria de qualidade da energia fornecida [126].

A vantagem da optimização pelo algoritmo das formigas relativamente aos outros métodos apontados, advém do facto deste lidar com maior facilidade com as alterações dinâmicas do grafo representativo dos caminhos [19], como acontece, por exemplo, na modelização de problemas de tráfego e, em concreto, no tráfego urbano.

Do ponto de vista do formalismo matemático, o algoritmo estabelece uma probabilidade de uma formiga viajar do nó $i$ para o nó $j$ dada por:

$$p_{i,j} = \frac{[\tau_{i,j}(t)]^{\alpha} . [\eta_{i,j}]^{\beta}}{\sum \left( [\tau_{i,j}(t)]^{\alpha} . [\eta_{i,j}]^{\beta} \right)} \tag{7.12}$$

onde $\tau_{i,j}(t)$ é a quantidade de produto químico existente no trajecto, $\eta_{i,j}$ representa a atractividade do troço (tipicamente o inverso da distância) e $\alpha$ *e* $\beta$ são parâmetros de controlo da influência de cada um dos termos.

**Actualização do produto químico no troço em cada iteração**

Representando $\tau_{i,j}(t) = (1-\rho)\tau_{i,j}(t) + \Delta\tau_{i,j}(t)$, a quantidade de produto químico existente no trajecto e $\rho$ a taxa de evaporação do químico, o termo $\Delta\tau_{i,j}(t)$ dá a quantidade do novo químico depositado, cuja expressão é representada por:

$$\Delta\tau_{i,j}(t) = \begin{cases} \frac{1}{L}, \textit{ se a formiga circula no trajecto} \\ 0, \textit{ noutro caso} \end{cases} \tag{7.13}$$

onde, neste caso, $L$ representa o custo do trajecto.

Existem, no entanto, algumas variantes à formulação exposta anteriormente, das quais se referem três, designadamente:

- **Primazia** - a melhor solução global deposita químico em todos os troços do trajecto;
- **Mínimo - Máximo** - existe um valor mínimo e um valor máximo admissível de depósito do químico;
- **Lista de classificação** - todas as soluções são classificadas de acordo com a sua aptidão (valor do custo). A quantidade de químico depositada é então ponderada para cada solução, de forma a que as soluções com melhor aptidão depositem mais químico do que as soluções com pior aptidão.

### 7.2.3 Algoritmo de pesquisa baseado no comportamento das formigas

Do ponto de vista do algoritmo, pode ser feita a analogia entre o alimento e a necessidade de manutenção[3] de um aero-gerador e entre o caminho mais curto e o caminho com menor custo associado.

O problema em questão pode ser decomposto em dois sub-problemas, o primeiro para determinar os percursos mais favoráveis entre um nó de origem e um nó de destino e o segundo, para descobrir a sequência das acções de manutenção para cada um dos geradores susceptíveis de sofrer manutenção.

O segundo aspecto é importantíssimo e, apesar da melhor adaptação deste algoritmo, onde a dinâmica do sistema muda frequentemente, a verdade neste problema passa pela necessidade de existir uma matriz de depósito do químico para cada percurso, pois, caso contrário, apareceriam vários trilhos ao mesmo nível, tornando aleatório o melhor trajecto em função do troço mais percorrido. Em resumo, no i-ésimo trajecto para as **m** formigas o resultado é guardado no i-ésimo nível da matriz. Esta restrição significa que, para uma urgência de **n** dias de reparação a **nMaxnos** nós distintos, é necessária uma Matriz de $nMaxnos \times nMaxnos \times nTécnicos$, assumindo-se resolvido o problema dos percursos óptimos entre dois pontos e qual o melhor nó para pernoitar.

Relativamente à concretização das equações 7.12, 7.13, a proposta passa por considerar

$$\eta_{i,j} = custoNó + custoDoTrajecto \tag{7.14}$$

A deposição do produto químico poderia ser inversamente proporcional à qualidade da solução, o que pode ser equacionado pela seguinte fórmula:

$$\Delta\tau_{i,j}(t) = \frac{1}{custo\ global} \tag{7.15}$$

Colocadas as ressalvas anteriores, este algoritmo pode ser utilizado para resolver o problema em apreço, de acordo com o seguinte algoritmo:

**Algoritmo 7.1** *Algoritmo da Formiga para optimização de trajectos de manutenção*

***Inicializar***
***Repetir***
    ***a)*** *Calcular* ***m*** *soluções (m formigas)*
    *b) Para cada solução calcular o custo*
    *c) Refrescar o químico de acordo com a qualidade das soluções (uma matriz por cada trajecto)*
    *d) Guardar a melhor solução*
***Até*** *solução estável e/ou máximo iterações atingidas*

o sub-problema **a)** pode ser resolvido e descrito como:

[3] Ponderada pelo custo de perda de produção

**Algoritmo 7.2** *Passo a) do Algoritmo 7.1*

*Reiniciar os pontos de partida e os destinos*
*Para as **m** formigas*
*Para cada destino colocar o máximo de visitas admissíveis, e nos locais a pernoitar um valor referente ao número de camas (ou um valor elevado)*
*Enquanto houver locais a visitar*
*Coloca todos os destinos elegíveis como desocupados*
*Para cada Técnico*
*Seleccionar um **destino1** com base em p(,)*
*Se não houver **destino1** terminar este ciclo*
*Calcular o melhor caminho entre **origem1** e **destino1** (alg. da formiga autónomo)*
*Diminuir o número de visitas em **Destino1** e considerá-lo ocupado*
*Fim Para*
*Para cada Técnico*
*Seleccionar **destino2** =local de **dormida** ou a **saída** caso não hajam mais destinos*
*Calcular o melhor caminho entre **PosiçãoActual** e **destino2** (alg. da formiga autónomo)*
*Diminuir o número de camas disponíveis*
*Fim Para*
*Fim Enquanto*
*Fim Para*

O algoritmo precedente é um contributo para a perspectiva holística da tese, não sendo, contudo, um elemento basilar da mesma, pelo que não foi alvo de desenvolvimentos adicionais.

### 7.2.4 Algoritmo de pesquisa baseado em algoritmos genéticos

No anexo C, nas pág. 198 e pág. 206 faz-se uma breve introdução aos algoritmos genéticos, cujas técnicas foram inspiradas na evolução natural das espécies. Em síntese, o algoritmo segue os seguintes passos: aleatoriamente gera-se uma população de **n** indivíduos solução para o problema em causa, avalia-se o seu desempenho, escolhendo-se os melhores para produzir a geração seguinte através de vários operadores genéticos, tais como a descendência, a mutação, a roleta aleatória, entre outros.

Mais uma vez, nesta solução, o problema divide-se em dois sub-problemas: definir qual o melhor caminho entre dois nós e determinar a sequência de visitas a cada nó.

O problema de encontrar o melhor caminho entre dois pontos é extremamente estudado na teoria de grafos onde se encontram alguns algoritmos propostos. Uma das soluções, quando os custos associados aos trajectos são positivos, é o algoritmo de *Dijkstra* [127]. No caso dos custos associados aos trajectos apresentarem valores negativos, o algoritmo de *Dijkstra* não funciona e uma das soluções da área da teoria de grafos é dada pelo algoritmo de *Bellman Ford* [128]. Este problema pode ainda ser resolvido com base nos algoritmos genéticos, tendo como representação

| Dia nº 1 | | | Dia nº 2 ... | |
|---|---|---|---|---|
| Técnico 1<br>nó trabalho\|nó dormida | ...<br>... | Técnico n<br>nó trabalho\|nó dormida | Técnico 1<br>nó trabalho\|nó dormida | ...<br>... |

Tabela 7.1: Formato genético de um indivíduo.

um vector constituído por cada nó ordenado aleatoriamente. Para um indivíduo ser solução do problema, apenas é necessário que o primeiro elemento do vector seja o nó de origem. Os operadores genéticos utilizados foram os descritos na Fig. 7.3, a), b), c) e d).

Quanto ao problema de base, isto é, determinar a sequência de visitas das equipas, iniciou-se a sua resolução pela codificação de uma solução, ou seja, um indivíduo é representado pela Tab. 7.1[4] sendo o seu comprimento **n** dado por

$$n = nE \times nTecnicos \times 2 + nTecnicos = nTecnicos.(nE \times 2 + 1)$$
$$nE = Max(nTotalDiasDeReparacao, tamY_matriz_nHProdPorDia, tamY_matriz_euroPorMW) \quad (7.16)$$

onde se considera que cada equipa de manutenção inicia o seu "caminho" num nó de entrada pré estabelecido (na Fig. 7.1 o nó 1), nó este que não está incluído na representação genética indicada na Tab. 7.1, por ser idêntico em todas as soluções.

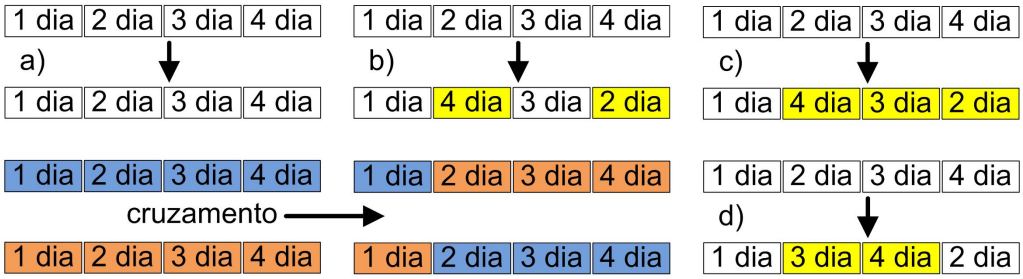



Figura 7.3: Operadores genéticos utilizados.

**Operadores genéticos**

Para toda a população, calculam-se grupos aleatórios de seis elementos e, para o melhor indivíduo, desses seis elementos aleatórios geram-se quatro novos indivíduos:

a) Mantém-se o próprio indivíduo;

b) Trocam-se de lugar dois elementos aleatoriamente. Considera-se elemento o conjunto de todas as visitas do mesmo dia (a ordem das visitas num dia de todos os técnicos não sofre nenhuma alteração);

c) Inverte-se a ordem a um conjunto aleatório de *n* dias;

[4]Outra alternativa seria considerar uma representação matricial, onde nas linhas se caracteriza as diferentes equipas de técnicos e nas colunas os vários dias, sendo o formato do seu conteúdo idêntico ao proposto.

d) Move-se um bloco de *n* dias uma casa à frente.

Para os restantes dois elementos, aleatoriamente, geram-se dois novos indivíduos, ou faz-se o cruzamento entre os dois melhores dos seis, verificando-se e corrigindo-se os dois novos indivíduos para garantir que são solução do problema (Fig. 7.3).

O algoritmo 7.3 apresenta a solução descrita anteriormente, tendo sido implementado com sucesso, sendo os resultados apresentados no capítulo 8.

**Algoritmo 7.3** *Algoritmo genético*

```
Reiniciar vectores de métricas
Para as nIter iterações
    população=gera_individuos(x)
    Calcular o desempenho
    Se o melhor(população) é melhor que o actual
        guarda o melhor resultado
    Aplicar aleatoriamente a grupos de 6 indivíduos os operadores genéticos
Fim Para

gera_individuos(nelementos)
Para nelementos
    Enquanto não acabou
        Para cada técnico
            Escolhe um nó para reparar
            Escolhe o melhor nó para pernoitar
        Fim Para
    Fim Enquanto
Fim Para
```

Os operadores apresentados na Fig. 7.3 não são suficientes para permitir uma procura em todo o espaço de soluções (apesar de a cada iteração, ser criado um novo indivíduo), porque trabalham ao nível do dia. Estão excluídas nomeadamente as permutações num dia para os *n* técnicos, pois o interessante é melhorar as características da melhor solução até ao momento. Dito de outra forma, suponha que num dia o técnico 1 visitou o nó A, o técnico 2 visitou o nó B; no dia seguinte o técnico 1 visita o nó C e o técnico 2 visita o nó D. A sequência é, portanto, |(A,dormir), (B,Dormir)|, |(C,dormir), (D,dormir)|, mas, pode ser mais barato, por questões de custos de trajecto, fazer-se a troca no segundo dia entre os técnicos, dando origem à sequência: |(A,dormir), (B,Dormir)|, |(D,dormir), (C,dormir)|. Este foi o operador re-introduzido após algumas experiências onde, numa população de 100 indivíduos com 1800 iterações, apenas se conseguiam cerca de 20 soluções óptimas parciais. Foram então introduzidos os mesmos operadores b) c) e d) a nível dos trabalhos dos técnicos associados a um intervalo de dias aleatórios (neste caso escolhe-se um grupo de oito indivíduos).

Indubitavelmente verifica-se ainda o facto dos operadores genéticos, ao cruzarem indivíduos, poderem gerar novas populações que não são soluções do problema. Este é um aspecto importante que condiciona a utilização de certos operadores genéticos ou, ainda, obriga à "reparação" dos indivíduos da nova geração, de modo a serem solução do problema.

Listagem 7.1: Programa para calcular uma solução específica.

```
function  individuo= gerasolucaomelhor(n,nMaxNos,noDeSaida,nTecnicos)
global perdaEurosPorAvaria perdaEurosReparar
global nDiasdeReparacao nDiasPraAvariar

  individuo=ones(1,n)*noDeSaida;
  [llsy2,llsx2]=size(perdaEurosReparar);
  ciclo=1;
  ndias=1;
  lv_ndiasreparacao=nDiasdeReparacao;
  lv_ndiasPraAvariar=nDiasPraAvariar;
  indices=find(lv_ndiasreparacao); % obtem nós que ainda faltam reparar (indices nonzero)
  [ignore,sxy]=size(indices);
  while sxy>0
      lv_backup_diasReparacao   = lv_ndiasreparacao;
      lv_backup_ndiasPraAvariar = lv_ndiasPraAvariar;
      for ntc=0:nTecnicos-1
                % Valor mínimo onde reparar.
                diasPraFimProducao=lv_backup_ndiasPraAvariar-lv_backup_diasReparacao;
                [min_dist,index] = min(diasPraFimProducao(indices)); % Minimo
                if (isempty(min_dist))
                    continue;
                end
                index=indices(index);

                indices=find(diasPraFimProducao<=0);
                lv_pp=perdaEurosPorAvaria(min(ndias,llsy2),indices)-perdaEurosReparar(min(ndias,llsy2),indices);
                [max_perda,indperda]=max(lv_pp);
                if isempty(indperda)
                    indperda=index;
                else
                    indperda=indices(indperda);
                end
                individuo(ciclo+ntc*2)=indperda;
                lv_backup_ndiasPraAvariar(indperda)=Inf; % Não deixa este nó ser escolhido por outro técnico
                lv_backup_diasReparacao(indperda)=0;     %

                lv_ndiasreparacao(indperda)=lv_ndiasreparacao(indperda)-1; % "
                if lv_ndiasreparacao(indperda)<=0
                    lv_ndiasPraAvariar(indperda)=Inf;  % Indica que já está reparado...
                end

                lv_dorms=dameNoParaDormir(indperda);
                individuo(ciclo+ntc*2+1)=lv_dorms; % nó de dormida

                indices=find(lv_backup_diasReparacao); % obtem nós que ainda faltam reparar (indices nonzero)
                [ignore,sxy]=size(indices);
      end
      ciclo = ciclo + 2*nTecnicos;
      ndias=ndias+1;
      % Falta menos um dia para avariar
      indices=find(lv_ndiasPraAvariar>0);
      lv_ndiasPraAvariar(indices)=lv_ndiasPraAvariar(indices)-1;
      % Recalcula os indices
      indices=find(lv_ndiasreparacao); % obtem nós que ainda faltam reparar (indices nonzero)
      [ignore,sxy]=size(indices);
  end
```

### 7.2.5 Algoritmo de pesquisa baseado em algoritmo específico

Este mesmo problema pode ser resolvido com base num algoritmo específico escolhendo os nós com os custos mais elevados, após subtrair os dias previstos para eventuais avarias e os respectivos dias de reparação, bem como os dias de manutenção planeada, dando a antevisão do número

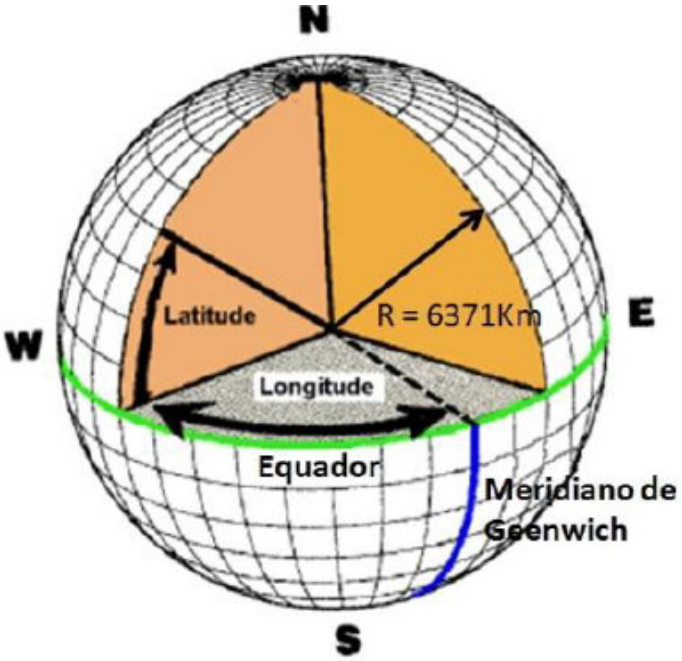



Figura 7.4: Modelo esférico do globo terrestre [20].

de dias até os custos crescerem - $DiasAteCustosDeAvaria = nDiasPraAvaria - nDiasReparacao$. Para os valores $DiasAteCustosDeAvaria \leq 0$ haverá perdas financeiras em caso de produção e, portanto, em princípio deve-se alocar uma equipa a estes nós em primeira mão. A função para calcular esta solução está apresentada na Listagem 7.1 e é fornecida como uma das soluções para o algoritmo genético - fornece um indivíduo para a população, em princípio "mais apto" do que um indivíduo gerado aleatoriamente. Deixa-se a ressalva que o algoritmo da Listagem 7.1 não contempla todas as possibilidades de optimização. Serve para gerar uma solução mais eficaz dando assim um ponto de partida ao Algoritmo genético 7.3.

### 7.2.6 Determinação de distâncias entre dois pontos terrestres

Resolver o problema de cálculo da distância entre dois pontos terrestres é, actualmente, relativamente acessível, em virtude do sistema de coordenadas GPS e do modelo geométrico terrestre que assenta em três aproximações possíveis:

- **Modelo esférico do globo terrestre** - neste modelo o globo terrestre é apresentado como uma esfera perfeita, constituindo uma aproximação à realidade. Para calcular a distância entre dois pontos recorre-se à trigonometria esférica;
- **Modelo elíptico do globo terrestre** - neste modelo o globo terrestre é apresentado como um elipsóide perfeito. A sua aproximação à realidade apresentada é melhorada relativamente ao modelo anterior;
- **Modelo geodésico do globo terrestre** - trata-se da forma verdadeira do globo terrestre, cuja precisão é maximizada à custa de uma maior complexidade dos cálculos. No entanto, dada

a dinâmica da superfície terrestre se encontrar em movimento, esta forma vai-se alterando ao longo do tempo, sendo a representação geodésica uma das áreas de conhecimento que lidam com este aspecto e com a representação do campo gravitacional terrestre.

Actualmente os equipamentos GPS utilizam modelos analíticos normalizados pela norma WGS84[5]; para além deste há outros modelos como o GRS80 ou o WGRS 80/84. Os predecessores do modelo WGS84 foram respectivamente o GRS80 e o WGS72, este último inadequado para as aplicações actualmente em perspectiva.

O GPS disponibiliza os ângulos longitudinais e latitudinais relativamente ao equador e ao meridiano de *Greenwich* (Fig. 7.4). No modelo esférico, de acordo com a Fig. 7.4 concluí-se facilmente que:

$$\begin{gathered} D = R.cos(\alpha).cos(\beta) \\ onde \\ \alpha \ - \ diferença\ das\ latitutes \\ \beta - diferença\ das\ longitudes \\ R \ - \ raio\ da\ terra \\ D \ - \ distáncia\ entre\ os\ pontos \end{gathered} \tag{7.17}$$

Existe outro formulário mais preciso para a determinação de distâncias entre dois pontos referenciados pelas coordenadas do sistema GPS; no entanto, as da Eq. 7.17 são suficientes para o presente problema. A referência [20] dá informação adicional sobre esta temática.

## 7.3 Conclusões

Este capítulo apresentou pormenorizadamente a forma de implementação do planeamento da manutenção em parques eólicos. Trata ainda da gestão global deste tipo de equipamentos, nomeadamente de vários aspectos operacionais relacionados com a actividade da manutenção de parques de aero-geradores, quer terrestres quer marítimos. Neste caso, houve alguns problemas com a determinação da melhor solução à luz das restrições reais, tendo sido ultrapassados após algumas modificações. É apresentada uma técnica de optimização de percursos entre aero-geradores que pode ainda ser utilizada em problemas similares onde a distância entre equipamentos seja uma questão relevante e, por consequência, a determinação da ordem de realização das acções de manutenção planeada, isto é, qual deverá ser a sequência, em função das datas, das tolerâncias para a realização de cada um dos trabalhos e das disponibilidades das equipas alocadas a cada uma das intervenções.

No próximo capítulo apresentam-se os resultados da dissertação, que corresponde ao culminar do presente trabalho, mostrando um pouco do *puzzle* da tese na sua globalidade.

[5]WGS-World Geodetic System, sendo o WGS-84 válido entre 1984 e 2010, ao fim do qual será feita a próxima revisão e validação do modelo

# Capítulo 8

# Integração e validação dos modelos

Neste capítulo apresentam-se os resultados referentes aos modelos propostos nos capítulos anteriores, com os respectivos comentários e indicação de futuros caminhos de desenvolvimento.

## 8.1 Resultados práticos

### 8.1.1 Sistema de aquisição

Conforme descrito no capítulo 5 foi desenvolvida uma metodologia para sincronizar o RTC do módulo *gateway Ethernet-CAN*, com o servidor *Linux* do SMIT. O objectivo principal é permitir efectuar a aquisição de dados de forma sincronizada em diferentes pontos geográficos. De acordo com o exposto no capítulo 5, os sistemas de aquisição de dados encontram-se interligados com o módulo *gateway Ethernet* através de uma rede de campo *CAN*. Esta secção mostra a forma de sincronizar a recolha de dados nos diferentes nós de aquisição, a partir do módulo *gateway Ethernet-CAN*.

Os valores apresentados referem-se ao controlo do processador ARM LM3S8962, a trabalhar a uma frequência de relógio de 50Mhz. Esta frequência significa um ciclo de instrução de 20 ns, ou seja, 50 ciclos correspondem a 1 $\mu$s. A Fig. 8.1 apresenta o valor teórico e o valor medido em termos de tempo de propagação na rede *CAN*, para uma velocidade de operação de 250 kbps (tempo medido de acordo com a Fig. 5.9 da pág. 77), quando se transmite uma mensagem *CAN Extended* da *gateway Ethernet-CAN* para o nó mais longínquo da rede. Como se constata, os resultados teórico e prático não coincidem, sendo os motivos mais evidentes para esta discrepância os seguintes:

a) O valor teórico não contabiliza o tempo de processamento em cada um dos módulos pela camada de *software*, quer no emissor quer no receptor;

b) Não foi considerado o tempo de propagação no meio físico em função da velocidade de propagação e do comprimento do cabo (20cm).

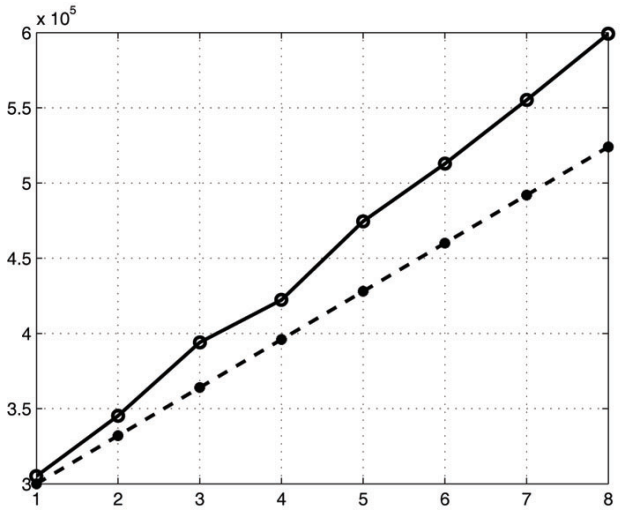



Figura 8.1: Medida do tempo de atraso na rede *CAN* (ns). A tracejado - valor teórico; a cheio - valor medido, para mensagens de 1 a 8 *bytes*.

Portanto, levando em conta a distância do cabo, introduzem-se atrasos não contabilizados no valor teórico. Para cabos curtos ($l<\lambda/10$)[1] a impedância do cabo é desprezável quando comparada com as impedâncias de adaptação. O tempo de propagação é dado por:

$$tempo = \frac{distância_{cabo}}{c_{propagacão}} \tag{8.1}$$

sendo, neste caso, desprezável, tendo em consideração os valores reais. A discrepância justifica-se assim devido ao tempo de tratamento de recepção e reenvio das mensagens nas PICs (dividido por dois). Os valores reais do tempo, medidos de acordo com este gráfico, são utilizados para enviar previamente o pedido de aquisição.

A Fig. 8.2 apresenta a diferença entre o instante imediatamente depois de programar a mensagem *CAN* e o instante da aquisição. De acordo com a Fig. 8.1, cada mensagem com 1 *byte* de dados deveria ser enviada 310 $\mu$s antes. Por opção, decidiu-se enviar ligeiramente mais cedo, 2 $\mu$s. O histograma presente neste gráfico mostra um erro de cerca de 2.5 $\mu$s (largura do histograma), sendo um erro percentual de 2.5/310=0,8% para uma velocidade de transmissão de 250 Kbps. Esta percentagem tem tendência a piorar quando se aumenta a velocidade do *CAN*, em virtude de ser expectável que o valor de 2.5 $\mu$s se mantenha. Como o tempo global diminui, dado que a velocidade de transmissão no *CAN* é mais elevada, percentualmente, os 2.5 $\mu$s terão mais influência. Ainda assim, 2.5 $\mu$s equivale a atrasar uma amostra com uma taxa de amostragem de 400 Khz. Sabendo que o valor máximo de aquisição será, no caso de sinais de vibração, de 20Khz, significa um erro de (2.5/50).T= 5%.T e, no caso de 10Khz, um erro de 2.5%.T. Para os sinais em questão é um desempenho relativamente bom. No actual processador 2.5 $\mu$s equivale a 125 ciclos de relógio.

[1] l: comprimento do cabo; $\lambda$: comprimento de onda.

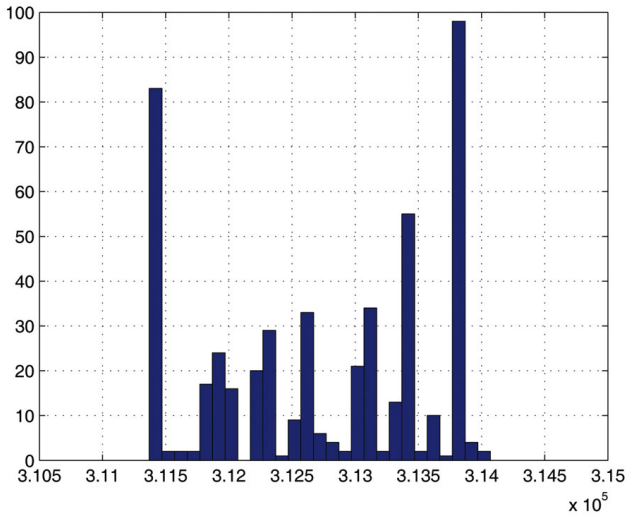



Figura 8.2: Medida do tempo para aquisição (ns), depois de ter programado o *CAN* para enviar uma mensagem com 1 *byte* de dados.

Reportando ao fluxograma da Fig. 5.11 da pág. 79 este foi totalmente implementado em C e, caso o grau de exigência fosse maior, por exemplo, para se conseguir responder a taxas de aquisição de 100 Khz, como poderia ser diminuído este tempo? As hipóteses de melhoria passariam pelas seguintes alterações:

a) Previamente programar os registos de *CAN*. De seguida ligar um temporizador para activar uma interrupção e nesta activar o registo de envio da mensagem *CAN* para início da transmissão;

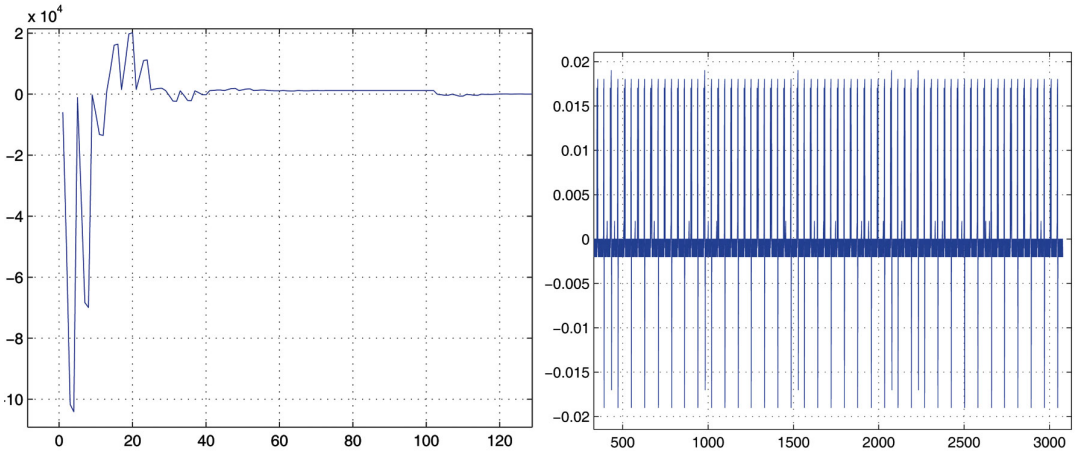
b) Programar em *assembly* de modo a conhecer com rigor as instruções e o número de ciclos de relógio que cada uma demora, para melhor definir o tempo de desvio. Inclui-se também a opção a) nesta linguagem.

### 8.1.2 Sistema de sincronização de relógios via IP

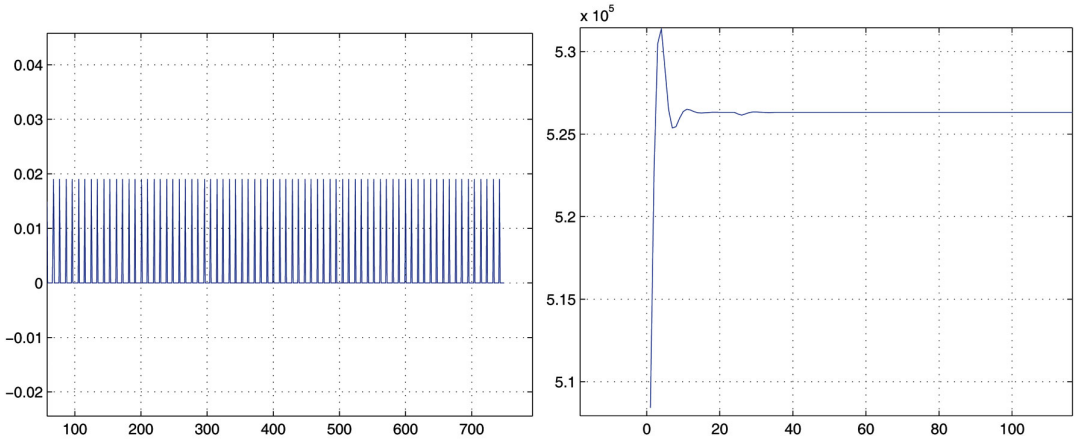
Conforme descrito no capítulo 5 foi desenvolvida uma metodologia para sincronizar o RTC do módulo *gateway Ethernet-CAN*, com o servidor *Linux* do SMIT. O objectivo principal é permitir efectuar a aquisição de dados de forma sincronizada em diferentes pontos geográficos. De seguida apresentam-se os resultados relativos ao acerto dos relógios através da rede TCP/IP entre dois sistemas, referentes à metodologia exposta no capítulo 5.

Para estudar o comportamento do sistema, isto é, do filtro e do ganho exposto nas Figs. 5.6 e 5.7 da pág. 74 e seguintes, desenvolveu-se um simulador em Matlab, que permite estudar o comportamento dos dois sistemas referidos. A diferença com a realidade reside nos seguintes aspectos: a) As mensagens na rede levam 45ms a ser transmitidas (valor alterável); b) Os tempos de envio e recepção são registados correctamente, logo, sem erros; c) A duração dos cálculos no controlo é considerada nula, não influenciando o resultado, sendo, portanto, equivalente a parar o

tempo durante esses instantes; d) O passo da simulação é de 1s, alterando-se para 45ms sempre que há mensagens para enviar ou para receber entre o nó mestre e o nó escravo.




(a) Diferença entre o relógio do Mestre e do Escravo ($\mu$s). **Esquerda**: instantes iniciais. **Direita**: instantes finais;

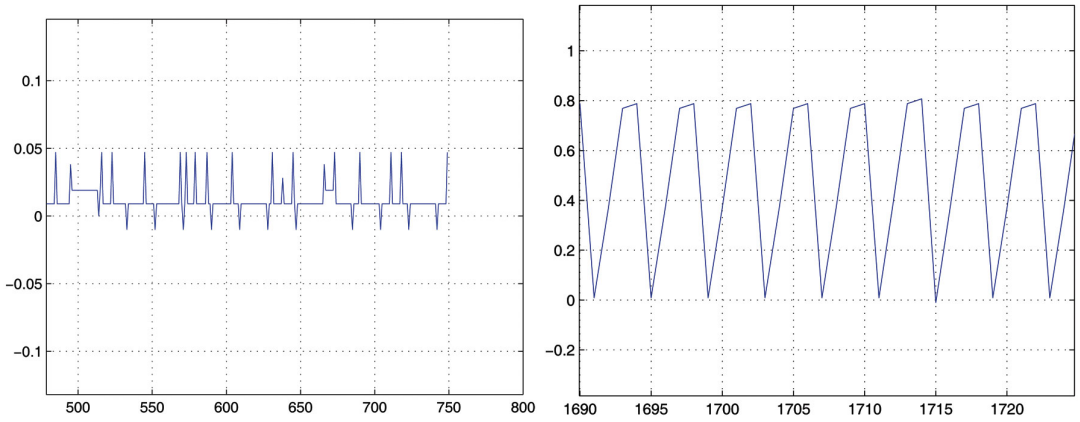



(b) **Esquerda**: Erro $\Delta t$ no momento de aplicar o controlo ($\mu$s). **Direita**: Evolução do valor de CNT_TICK_MAX;

Figura 8.3: Experiência nº 1. Valores calculados e recolhidos no momento da recepção das mensagens *Follow UP* e *Delay Response*.

Os dados considerados para a simulação são os seguintes: no mestre, o período do relógio é abaixo de 1ns; nos nós escravos, a frequência do relógio é de $freq_s = 50MHz$ (período 20ns), podendo-se apenas alterar as variáveis $N_s : N_{ns}$ e o valor de $CNT_TICK_MAX$. O aspecto mais importante da simulação foi ter-se considerado um desvio de 5.263% na frequência de 50MHz, o que introduziu a necessidade da PLL variar o parâmetro $CNT_TICK_MAX$ (ver pág. 74).

A Fig. 8.3 apresenta os resultados da simulação para o protocolo PTP a funcionar normalmente ao longo de 1500s. Pode-se observar a convergência dos relógios, ao fim de 40 mensagens de

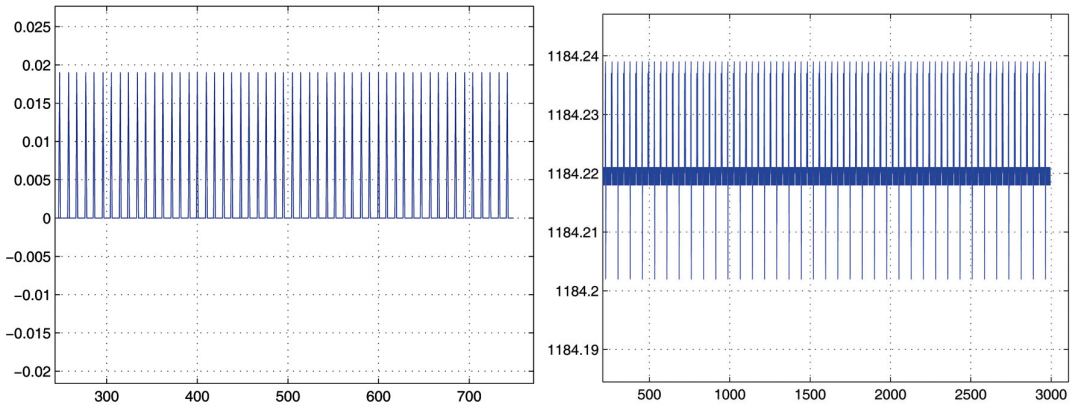



(a) **Esquerda**: erro $\Delta t$ ao controlar. **Direita**: diferença entre o relógio do Mestre e do Escravo ($\mu$s);

Figura 8.4: Experiência nº 2: Precisão de zero casas decimais no parâmetro CNT_TICK_MAX.

sincronismo e, em regime final, um desvio máximo de 40 ns. Nesta experiência considerou-se uma precisão de duas casas decimais no parâmetro CNT_TICK_MAX, tendo o seu valor final sido estimado em 526315.79 para um valor real de 526315.789474. No desfecho, o relógio do escravo estava adiantado em 2 ns. As mensagens *Delay Request* foram aleatoriamente enviadas entre 2 e 30 TSync.

A Fig. 8.4 apresenta a mesma simulação alterando a precisão no parâmetro CNT_TICK_MAX para zero casas decimais, tendo, no final, sido estimado o valor de 526316.0. O relógio do escravo encontrava-se, após a simulação, atrasado de 769 ns. O erro em regime final foi de cerca de 800ns.




(a) **Esquerda**: Erro $\Delta t$ ao controlar. **Direita**: Diferença entre o relógio Mestre e Escravo ($\mu$s);

Figura 8.5: Experiência nº 3: Precisão de duas casas decimais no parâmetro CNT_TICK_MAX. Apenas envia uma mensagem *Delay Request*.

A Fig. 8.5 apresenta a mesma simulação alterando a precisão no parâmetro CNT_TICK_MAX para duas casas decimais, tendo, no final, sido estimado o valor de 526315.79. No entanto, neste

exemplo, apenas se envia a primeira mensagem *Delay Request*, não havendo lugar ao envio de mais nenhuma. O resultado aparece evidenciado na figura através do valor em regime final, que representa um erro grosseiro devido à má estimação do tempo de propagação na rede. No entanto, em termos de variação, o resultado mantém a precisão de 40 ns (gráfico à direita na Fig. 8.5).

A Fig. 8.6 apresenta o resultado de acordo com a alteração proposta ao protocolo PTP, nomeadamente a nível das mensagens de *Sync* e *Follow Up*, tal como indicado na Fig. 5.7 (pág. 75). A diferença, quando comparada com os resultados da Fig. 8.3, reside no erro inicial, com melhorias significativas nesta última experiência. Na experiência nº 1, os erros iniciais foram na casa das centenas de milissegundos e, na última passaram para as décimas de $\mu s$. A nível da estabilidade final, o resultado é idêntico, sendo, portanto, a vantagem conseguida na rapidez para obter estabilidade após uma mudança drástica do tempo. As especificações para esta norma, a nível do *hardware*, nomeadamente no suporte de registo do tempo de recepção e envio das mensagens *Ethernet*, corresponde à abordagem para se obterem precisões na casa dos nanossegundos. Torna-se perfeitamente praticável a alteração proposta, obviamente com *hardware* que registe no pacote o tempo de envio das mensagens (o sistema implementado não tem esta funcionalidade).

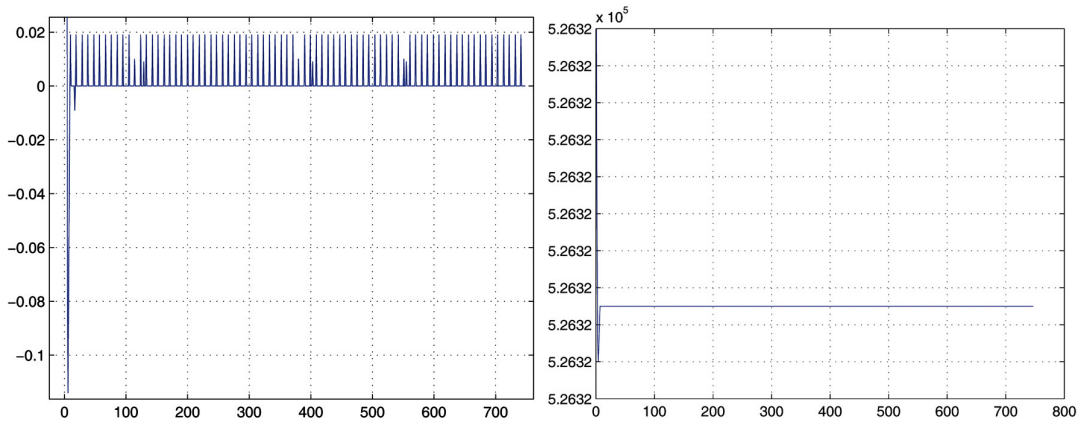
Na implementação real no micro controlador, o registo de CNT_TICK_MAX é inteiro. A pergunta é a seguinte: como se obtém precisão de duas casas decimais? Recorde-se que este valor representa o número necessário de contagens do relógio base do micro controlador para se conseguirem intervalos de 10ms. Num segundo, existem 100 intervalos de 10ms. Supondo um valor de 456.78, ter-se-ia de colocar neste registo em 100 intervalos, 78 com o valor 457 e 22 com o valor 456.

Nas simulações verificou-se ainda que, para determinados desvios de frequência, o sistema consegue eliminar completamente o erro, levando muito pouco tempo a consegui-lo. Depende da relação do desvio e da influência no contador CNT_TICK_MAX. Se o resultado der um valor inteiro, o sistema elimina o erro, caso contrário é necessário sempre aplicar o controlo para não permitir o desvio dos relógios (o que é obrigatório num sistema real).

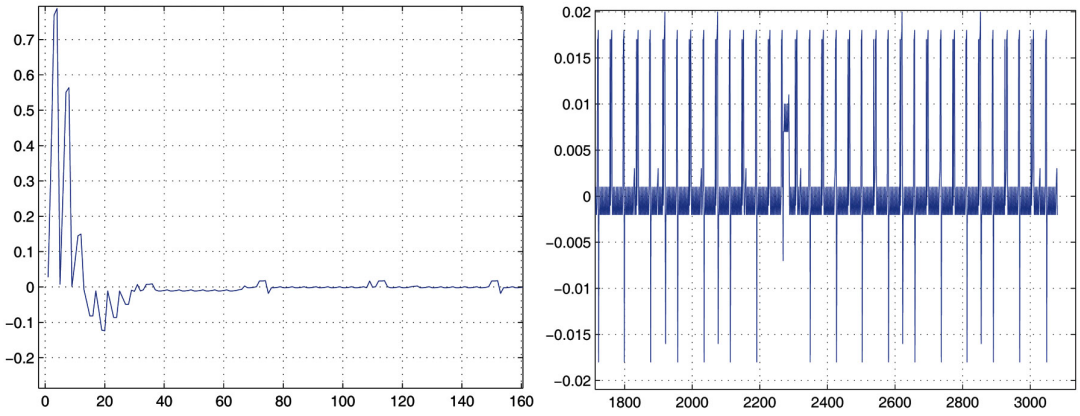
A Fig. 8.7 apresenta os resultados num sistema real. Na Fig. 8.7(c) mostra-se a utilização de um programa para detectar vários protocolos num *interface* real de *Ethernet*. Neste caso apenas mostra os pacotes do protocolo em questão, com o nó escravo a enviar sempre uma mensagem *Delay Request* ao mestre após a recepção da mensagem *Follow UP* (o protocolo assume valores aleatórios para não sobrecarregar o processador do relógio mestre). Nos resultados reais confirmou-se a pouca influência em enviar sempre um pedido *Delay Request* após a recepção da mensagem de sincronismo, quando comparado com o facto do mesmo pedido ser aleatório entre valores de [2;30].TSync.

A Fig. 8.7(a) apresenta, à esquerda, o erro total. A experiência iniciou-se com o relógio do nó mestre num PC real. Em determinado momento desligou-se o servidor de relógio no PC e passou-se o relógio mestre para um PC virtual em *VMWare*[2]. O PC normal apenas tinha instalado o *Linux Slackware* e o servidor de relógio; no PC virtual foi instalado o mesmo sistema, acrescido da base de dados SMIT que se encontrava em funcionamento. Este gráfico pretende mostrar a

[2] http://www.vmware.com




(a) **Esquerda**: Erro $\Delta t$ no momento de aplicar o controlo ($\mu$s). **Direita**: Evolução do valor de CNT_TICK_MAX;

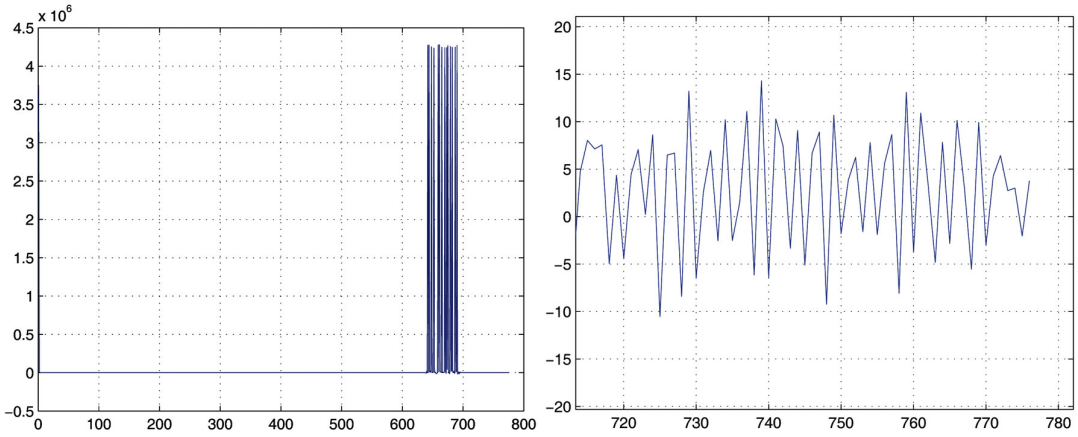



(b) Diferença entre o relógio do Mestre e do Escravo ($\mu$s);

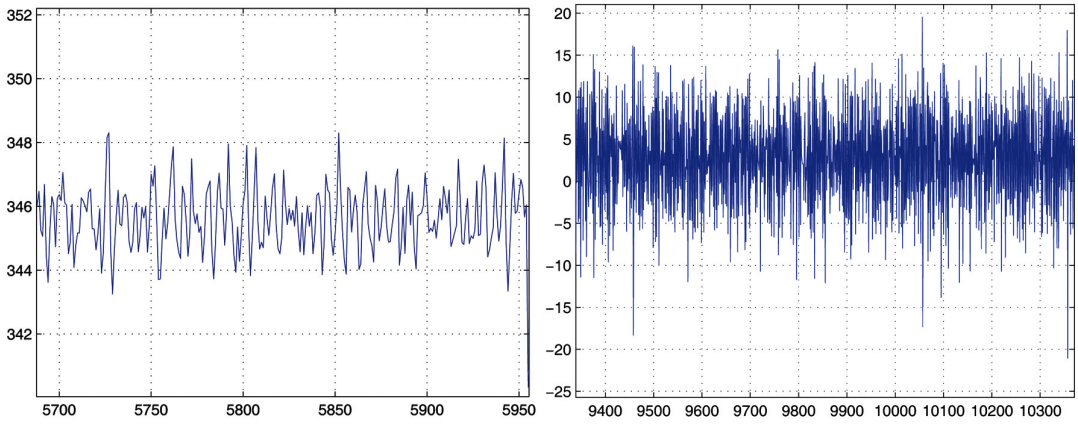
Figura 8.6: Experiência nº 4. Novo método com condições análogas à experiência nº1 (Fig. 8.3).

obrigatoriedade da estabilidade do relógio mestre que, como era previsível num PC virtual, não é assegurada. A consequência desta instabilidade fica patente no gráfico ilustrado nos instantes 600 e 700, quando o mestre do relógio era o PC da máquina virtual. O gráfico da direita apresenta o erro em regime final, na casa dos 30 $\mu s$, sendo mais credível considerar um erro em regime final de 40 $\mu s$ (Fig. 8.7(a), direita), em função das várias experiências realizadas.

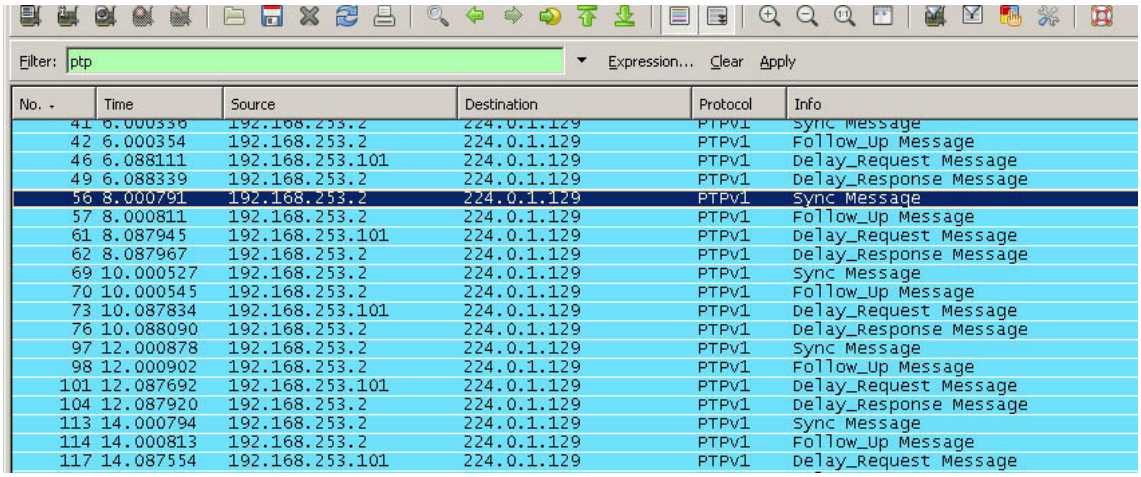
A Fig. 8.7(b) expõe o resultado no gráfico da esquerda da experiência na qual apenas se envia uma mensagem *Delay Request* ao início. Conforme se verifica, o sistema estabiliza mas com um erro grande em regime final, cerca de 345 $\mu$s, tal como previsto na simulação equivalente indicada na Fig. 8.5(a), à direita. Tal influência pode ainda ser verificada na Fig. 5.5 da pág. 73 advindo o erro de estimação do tempo de propagação na rede, que sofre influências do desvio de frequência do processador. Nas experiências reais realizadas utilizou-se um *switch LinkSys WRT54GC* para interligar os diferentes equipamentos, nomeadamente, um portátil com o sistema




(a) **Esquerda**: Erro total $\Delta t$ ($\mu$s). **Direita**: Erro em regime final;




(b) Erro $\Delta t$ ($\mu$s). Duas experiências com 1 mensagem *Delay Request* (esquerda) e 10 mensagens *Delay Request* (direita);




(c) Detecção dos pacotes PTP com um filtro de pacotes *Ethernet*;

Figura 8.7: Experiência nº 5. Resultados com o sistema real.

operativo *windows* e uma máquina virtual *Linux VMWare*, um computador com o sistema operativo *Linux* e uma placa com o processador ARM.

Listagem 8.1: Pseudo-código para controlo do acerto no relógio local

```
unsigned long AjustaControlo(Integer32 Delta_T)
{
    static Integer32 gv_last_DeltaT=0;
    static unsigned long gv_last_controlo=500000*TICKNS;
    unsigned long controlo,CNT_TICK_MAX_TIMES_TICKNS;
    double mm;

    mm=(Delta_T+gv_last_DeltaT)/2;
    gv_last_DeltaT=Delta_T;
    mm = mm / SYSTICKHZ;  // 100  (nanosegundos deslizados em 10ms)

    // Conta agora o tempo de 10ms de acordo com o CNT_TICK_MAX * 20ns.
    // será a medida dele para 10ms em nanosegundos.
    if (g_ulSystemTickReload>=100) controlo = (g_ulSystemTickReload) * TICKNS;
    else controlo = (SysCtlClockGet() / SYSTICKHZ) * TICKNS;

    if (mm>90000) mm =90000; // limite devido ao erro inicial não provocar instabilidade
    if (mm<-90000) mm =-90000;
    // Actualiza o CNT_TICK_MAX (=controlo / TICKNS)
    controlo=controlo - mm*0.6;  //(medida em nanosegundos)

    CNT_TICK_MAX_TIMES_TICKNS=(gv_last_controlo+controlo)/2;
    gv_last_controlo=controlo;

    return CNT_TICK_MAX_TIMES_TICKNS;
}
```

Quanto ao controlador, as mensagens recebidas do PTP foram utilizadas de acordo com as equações indicadas no final da secção 5.2.2, em concreto através do cálculo de $dm2e$, $de2m$, $t_{prop}$ e $\Delta t$. Dividindo o $\Delta t$ em unidades de 10ms sabe-se quanto é preciso adaptar CNT_TICK_MAX. Primeiro utiliza-se a média das duas últimas amostras, de seguida limita-se o valor obtido a $\pm 90000$, aplicando-se um factor de 0.6 para estabilizar. É feita ainda a média do valor de controlo para as duas últimas amostras. A média é equivalente a um filtro digital passa-baixo. O pseudo-código encontra-se na Listagem 8.1. Quer o limite, quer o factor 0.6 encontram-se de acordo com o esquema da Fig. 5.6 (pág. 74), nomeadamente a média das duas últimas amostras equivale a um filtro passa-baixo FIR.

### 8.1.3 Classificação

No capítulo 6 foi exposto um método para, do ponto de vista macroscópico, determinar o estado de funcionamento de um aero-gerador. A metodologia utiliza valores das várias variáveis, sendo estes sujeitos a uma classificação de estado (bom/mau funcionamento). Esta secção apresenta os resultados dessa metodologia, tendo sido validada com dados reais de um aero-gerador do norte do país. Os dados foram obtidos através do INEGI, sediado no *campus* da FEUP.

As Figs. 8.8 e 8.9 apresentam a classificação da curva de potência de acordo com as equações 6.6 e 6.7 (ver pág. 93).

No primeiro caso, Fig. 8.8, utiliza-se a relação entre a velocidade do vento e a potência produzida, a qual não leva em consideração as limitações impostas por uma activação dos travões do aero-gerador ou do ângulo de *pitch* das pás do rotor. Apesar de não ser uma avaria, a relação

normalmente utilizada não se manterá tendo, como consequência, a classificação errada de pontos bons de funcionamento como sendo instantes de avaria.

No segundo caso, Fig. 8.9, utiliza-se a relação velocidade do eixo do rotor *versus* potência activa produzida. Esta relação contempla as influências, quer do ângulo *pitch* das pás do rotor, quer do travão, quer de limitações de potência indicadas pelo explorador. Conforme pode ser observado, existem muitos pontos que violavam a relação da curva de potência, mas que, neste caso, são classificados como pontos de boa operação devido a não violarem a relação pela qual foram classificados (este é o método mais indicado para os dados disponibilizados).

No caso do aero-gerador nº 8, pode ainda verificar-se que a curva de potência aparenta operações constantes com ordenada de 1100, 800, 550 e 220 kW. Embora não exista nenhuma indicação, podem ser instantes em que a potência de produção esteve limitada por algum motivo. Ainda assim, o segundo método classificou de forma idêntica ao primeiro - avaria. Para ultrapassar este problema seria necessário "relaxar" as restrições no gráfico da direita da Fig. 8.9(b). Os dados, conforme indicado, pertencem a um parque eólico e não existe informação relevante naqueles para determinar com segurança qual a situação aplicável.

### 8.1.4 Análise de Séries Temporais

De acordo com o exposto no capítulo 6, apresentam-se nesta secção os resultados obtidos com os diferentes métodos de previsão com base em análise de séries temporais. Os métodos foram validados com séries temporais simuladas e com uma série Mackey-Glass (uma das referências no estudo deste problema) [129].

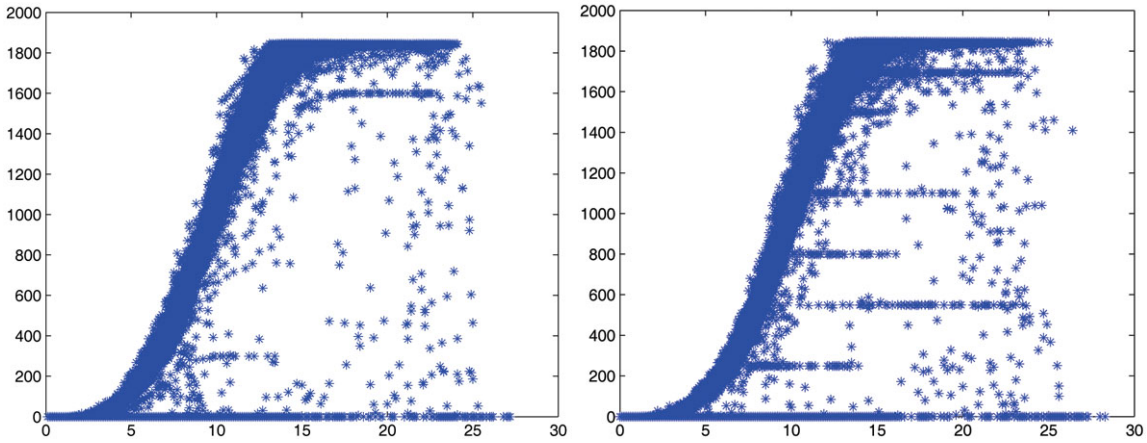
A Fig. 8.10 apresenta os resultados da simulação para os métodos descritos no anexo C. O sinal escolhido para elaborar os testes apresenta ruído gaussiano. Assim, o sinal é descrito matematicamente por:

$$S(t) = \begin{cases} t*0.2 + ruído_gaussiano*0.5, \ se\ t \leq 200 \\ 0 + ruído_gaussiano*0.8, \ se\ 200 < t \leq 400 \\ t*0.2 + ruído_gaussiano*0.8, \ se\ 400 < t \leq 500 \\ t*0.6 + ruído_gaussiano*0.6, \ se\ 500 < t \leq 600 \\ t*0.4 + ruído_gaussiano*0.5, \ se\ 600 < t \leq 800 \\ t^2*0.004 + ruído_gaussiano*0.5, \ se\ 800 < t \leq 1000 \end{cases} \tag{8.2}$$

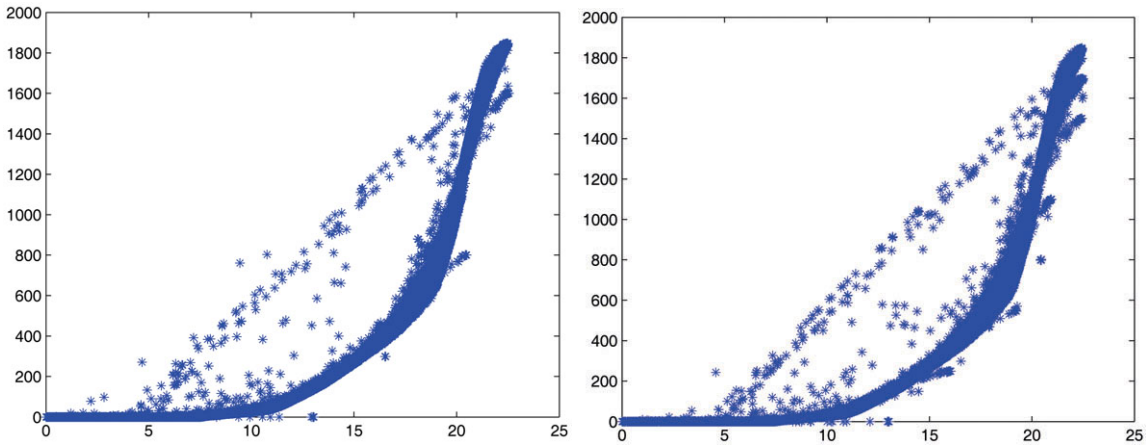
A finalidade do sinal é simular a divergência de determinado parâmetro para fora de valores de tolerância indicados, verificando como se comportam os diferentes algoritmos.

Os parâmetros utilizados em cada algoritmo foram os seguintes: ARRSE(0.5,0.2); ES(0.5); HW(0.5,0.2); HWSAS(0.5,0.2, 0.2,200); ESMSE(0.5,0); MAS(2); ARMA(2,2); SVR-RBF(30,10) e SVR-LIN(30,10) (consultar anexo C).

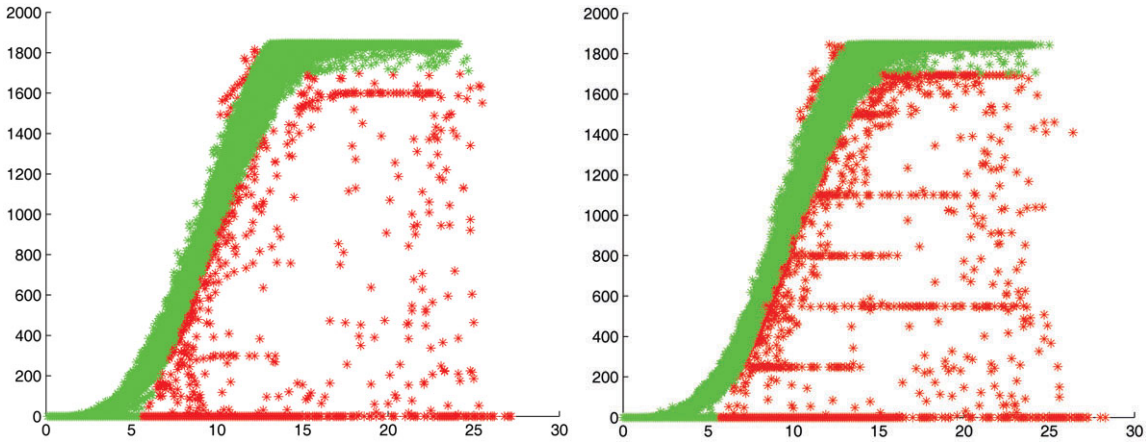
Por análise da Tab. 8.1, constata-se que o método ESMSE apresenta o melhor resultado, na maioria das métricas consideradas, para a previsão dos valores do instante seguinte. No entanto, o método não mantém o melhor desempenho no caso da predição de valores temporalmente mais




(a) Curva de Potência. Esquerda: aero-gerador 3; direita: aero-gerador 8;

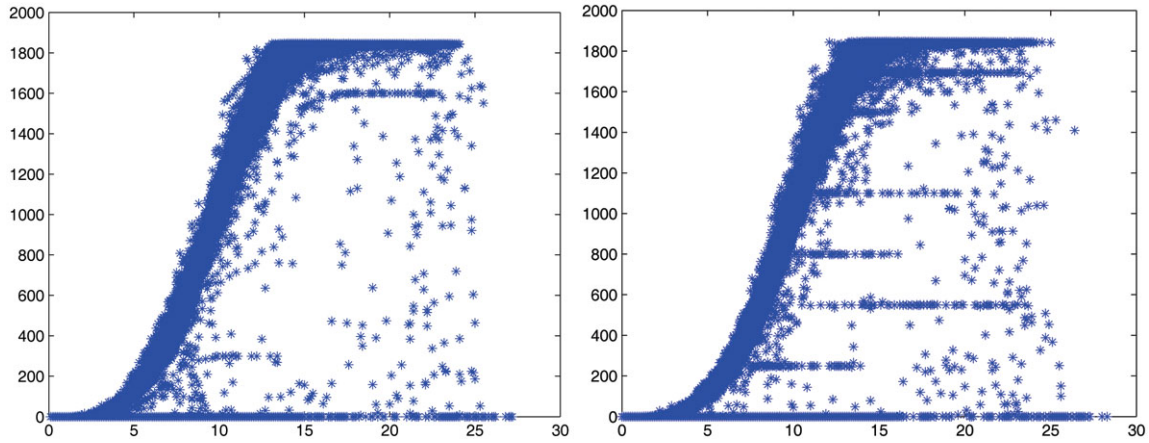



(b) Curva de Potência *versus* velocidade do eixo do rotor;

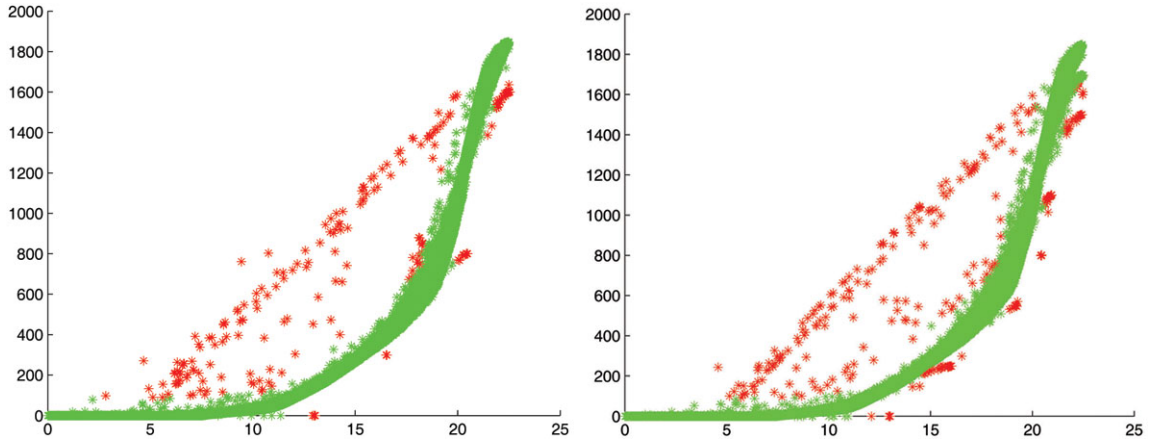



(c) Classificação por SVM da curva de Potência;

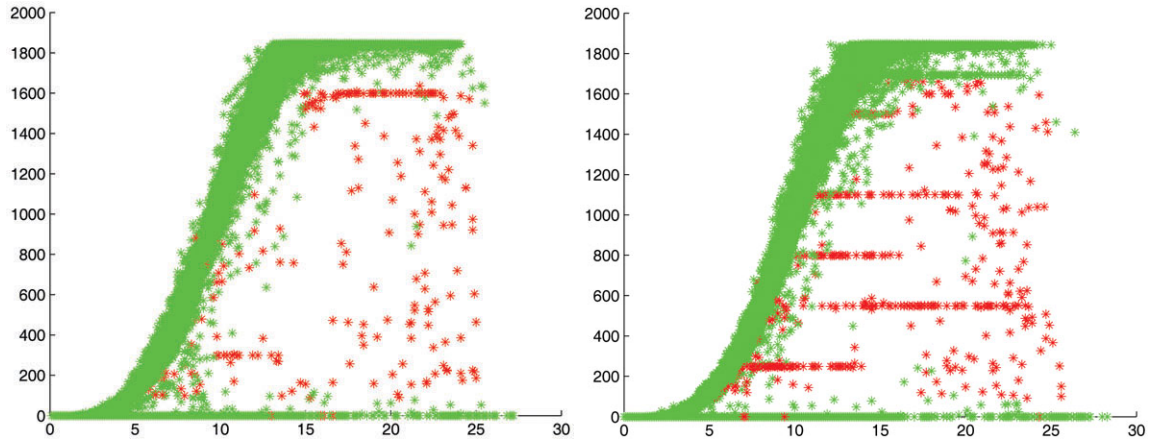
Figura 8.8: Classificação SVM dos dados de Parque Eólico, de acordo com a Eq. 6.6 da pág. 93. Vermelho: avaria; verde: bom funcionamento.



(a) Curva de Potência. Esquerda: aero-gerador 3; direita: aero-gerador 8;

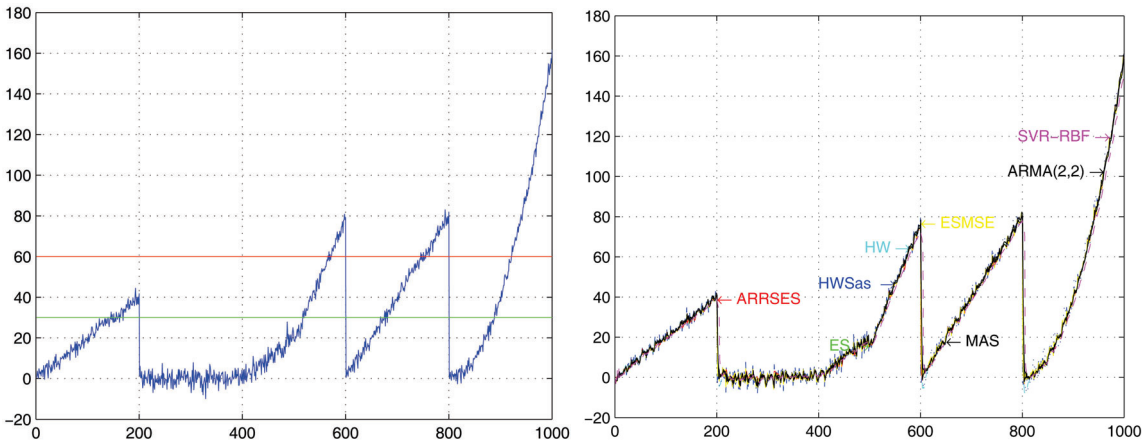


(b) Classificação por SVM da curva de Potência *versus* velocidade do eixo do rotor;

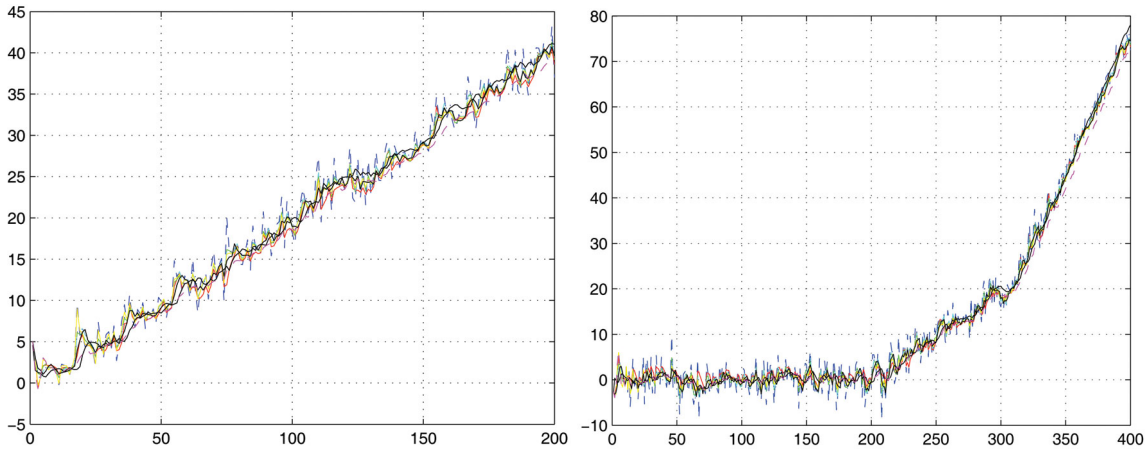


(c) Classificação SVM da Fig. 8.9(b) transposto para a curva de potência;

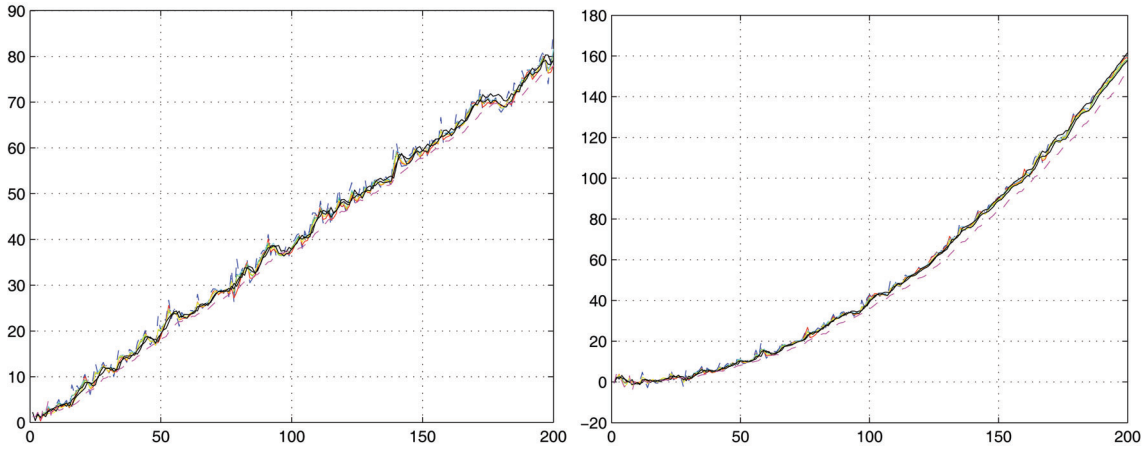
Figura 8.9: Classificação SVM dos dados de Parque Eólico, de acordo com a Eq. 6.7 da pág. 93. Vermelho: avaria; verde: bom funcionamento.




(a) Esquerda: sinal original. Direita: gráfico do desempenho dos vários métodos;



(b) Desempenho no troço 0-200 (esquerda) e 200-600 (direita);



(c) Desempenho no troço 600-800 (esquerda) e 800-1000 (direita);

Figura 8.10: Estudo comparativo dos vários métodos apresentados no anexo C.

| Alg. \ Métr. | MSE | TIC | ME | STD | MAE |
|---|---|---|---|---|---|
| ARRSE | 17.2324 | 0.0486 | 0.2495 | 4.1416 | 1.0677 |
| ES(0.5) | 20.4305 | 0.0531 | 0.3110 | 4.5070 | 1.3335 |
| HW | 20.3014 | 0.0527 | 0.1644 | 4.5005 | 1.1765 |
| HWSAS | 23.2613 | 0.0564 | 0.1804 | 4.8172 | 1.9004 |
| ESMSE(0.5,0) | **15.6385** | **0.0466** | 0.1432 | **3.9500** | **0.9850** |
| ESMSE(0.5,5) | 18.9677 | 0.0514 | 0.2290 | 4.3471 | 1.1795 |
| MAS-2 | 28.7002 | 0.0631 | 0.3884 | 5.3405 | 1.5840 |
| ARMA(2,2) | 30.4376 | 0.0644 | **-0.0087** | 5.5143 | 1.2391 |
| SVR-RBF | 83.9986 | 0.1094 | 1.0846 | 9.0960 | 3.5345 |
| SVR-LIN | 61.0381 | 0.0905 | -0.5677 | 7.7881 | 1.5086 |

| Alg. \ Métr. | MSE | TIC | ME | STD | MAE |
|---|---|---|---|---|---|
| ARRSE | 1.3644 | 0.0087 | 0.8515 | 0.8202 | 0.9596 |
| ES(0.5) | 3.7066 | 0.0186 | 1.3460 | 2.1118 | 1.7807 |
| HW | 3.0305 | 0.0155 | 1.0226 | 1.8419 | 1.2670 |
| HWSAS | 12.1969 | 0.0278 | 0.9774 | 3.6379 | 3.2466 |
| ESMSE(0.5,0) | **1.2975** | **0.0084** | 0.7932 | 0.8116 | **0.9121** |
| ESMSE(0.5,5) | 1.9957 | 0.0131 | 1.0038 | 1.4572 | 1.2820 |
| MAS-2 | 5.3347 | 0.0295 | 1.6295 | 3.6113 | 2.2742 |
| ARMA(2,2) | 1.6631 | 0.0095 | 1.0032 | **0.8083** | 1.0895 |
| SVR-RBF | 86.6412 | 0.0913 | 3.2680 | 11.5840 | 6.5191 |
| SVR-LIN | 2.6590 | 0.0612 | **-0.2522** | 8.3743 | 1.5896 |

Tabela 8.1: Resultados da previsão para a série da Fig. 8.10. À esquerda: resultados globais, à direita: resultados do intervalo 800-1000.

à frente no futuro, sendo igualmente acompanhado pela generalidade dos métodos. A Fig. 8.11 mostra a variação do $\alpha$, à esquerda, quando se utilizam todas as amostras passadas para recalcular o $\alpha$ de acordo com a Eq. C.23 da pág. 212. À direita tem-se a mesma variação do $\alpha$, mas apenas considerando o erro das últimas cinco amostras. A interpretação dos resultados é simples: no primeiro caso o $\alpha$ reflecte o histórico total dos erros e, quando há um erro elevado de previsão, sofre uma alteração grande, cuja influência é decisiva para o seu valor; no outro caso, com um "histórico" de cinco amostras, o $\alpha$ varia de forma mais dinâmica para compensar os erros mais recentes.

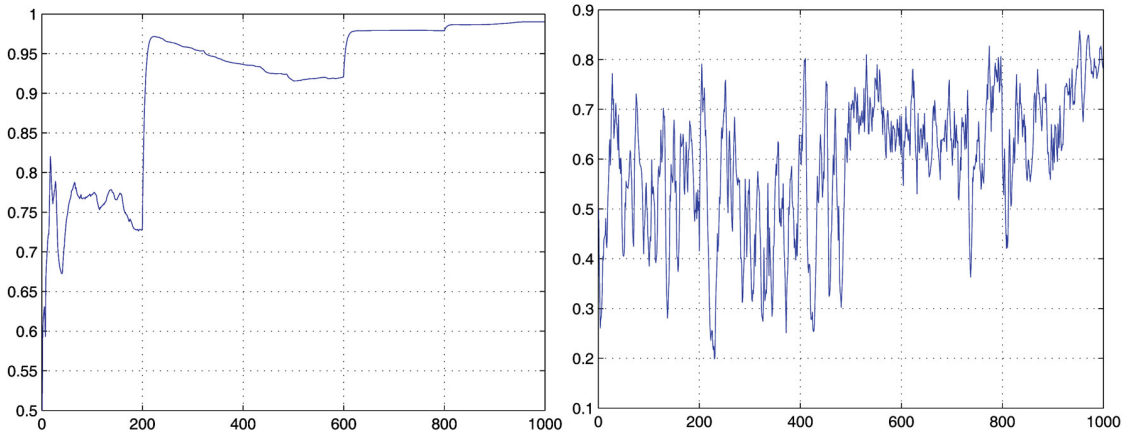



Figura 8.11: Esquerda: ESMSE(0.5,0), direita: ESMSE(0.5,5). Variação de $\alpha$ para a previsão de $\hat{y}[k] \approx y[k+1]$ da série da Fig. 8.10 cujo desempenho se encontra na Tab. 8.1.

Numa segunda simulação, a partir de uma série de referência, utiliza-se um sinal denominado de equação diferencial de Mackey-Glass[3], cuja série temporal é obtida após a integração da equação diferencial:

$$\frac{dx(t)}{dt} = \frac{A.x(t-\tau)}{1+x^C(t-\tau)} - B.x(t) \tag{8.3}$$

Nas experiências descritas na literatura [129], utiliza-se $A = 0.2$, $B = 0.1$, $C = 10$, $\tau = 17$ e, como condições iniciais, $x(0) = 1.2$ e $x(-\tau) = 0$ para $0 \leq t < \tau$ em conjunto com a aplicação do

[3]Descreve um sistema de controlo fisiológico e é normalmente utilizada como teste de comparação na análise do desempenho de algoritmos de séries temporais

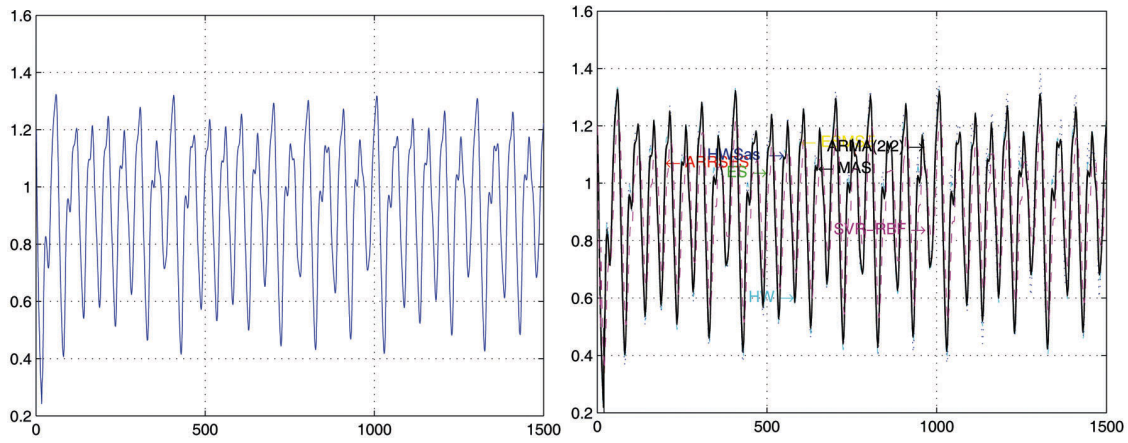



Figura 8.12: Resultados dos algoritmos de séries temporais aplicados à série *Mackey-Glass*: previsão de $\hat{y}[k] \approx y[k+1]$.

método de Runge-Kutta de quarta ordem com passo unitário, para cálculo dos valores da série. Esta equação diferencial foi, em primeira mão, utilizada para análise da concentração sanguínea e análise de doentes com leucemia [129].

As Figs. 8.12 e 8.13 e as Tabs. 8.2 e 8.3 apresentam os resultados obtidos pelos vários métodos indicados. Neste caso, fizeram-se quatro experiências para $\hat{y}[k]$, $\hat{y}[k+2]$, $\hat{y}[k+4]$ e $\hat{y}[k+6]$. Constata-se a deterioração da qualidade das previsões com o incremento dos passos no futuro. No entanto, neste caso, contrariamente à série anterior, o método ESMSE apresenta melhor desempenho para valores superiores de previsão no futuro. A Fig. 8.13 apresenta a variação de $\alpha$ de modo semelhante ao descrito para a série anterior sendo o comportamento idêntico, excepto pelo facto da série de entrada não apresentar descontinuidades.

### 8.1.5 Análise da optimização de trajectos

Esta secção apresenta os resultados relativos à análise da optimização do trajecto dos técnicos em função dos custos associados à perda de produção, tal como descrito no capítulo 7. Os exemplos apresentados reportam-se à Fig. 7.1 da pág. 111, para melhor visualização gráfica. Considerou-se assim um conjunto de nove aero-geradores, duas equipas de técnicos e o nó um

| Alg. \ Métr. | MSE | TIC | ME | STD | MAE |
|---|---|---|---|---|---|
| ARRSE | 0.0025 | 0.0261 | 0.0023 | 0.0496 | 0.0376 |
| ES(0.5) | 0.0041 | 0.0339 | **0.0001** | 0.0646 | 0.0543 |
| HW | 0.0029 | 0.0285 | 0.0003 | 0.0544 | 0.0434 |
| HWSAS | 0.0037 | 0.0321 | 0.0003 | 0.0612 | 0.0496 |
| ESMSE(0.5,0) | 0.0015 | 0.0202 | **0.0001** | 0.0385 | 0.0320 |
| ESMSE(0.5,5) | 0.0014 | 0.0198 | 0.0001 | 0.0378 | 0.0315 |
| MAS-2 | 0.0067 | 0.0432 | 0.0002 | 0.0821 | 0.0688 |
| ARMA(2,2) | **0.0012** | **0.0185** | 0.0006 | **0.0353** | **0.0290** |
| SVR-RBF | 0.0300 | 0.0922 | 0.0154 | 0.1724 | 0.1537 |
| SVR-LIN | 0.0104 | 0.0540 | 0.0074 | 0.1017 | 0.0928 |

| Alg. \ Métr. | MSE | TIC | ME | STD | MAE |
|---|---|---|---|---|---|
| ARRSE | 0.0049 | 0.0368 | 0.0025 | 0.0701 | 0.0482 |
| ES(0.5) | 0.0055 | 0.0392 | **0.0004** | 0.0747 | 0.0614 |
| HW | 0.0052 | 0.0378 | 0.0006 | 0.0721 | 0.0564 |
| HWSAS | 0.0151 | 0.0645 | **0.0004** | 0.1234 | 0.0996 |
| ESMSE(0.5,0) | **0.0026** | **0.0272** | **0.0002** | **0.0519** | **0.0412** |
| ESMSE(0.5,5) | 0.0026 | 0.0270 | 0.0003 | 0.0516 | 0.0407 |
| MAS-2 | 0.0086 | 0.0488 | 0.0009 | 0.0929 | 0.0763 |
| ARMA(2,2) | 0.0035 | 0.0313 | 0.0044 | 0.0594 | 0.0433 |
| SVR-RBF | 0.0470 | 0.1155 | 0.0165 | 0.2161 | 0.1903 |
| SVR-LIN | 0.0183 | 0.0713 | 0.0016 | 0.1353 | 0.1137 |

Tabela 8.2: Resultados da previsão para a série *Mackey-Glass*. À esquerda: previsão para $\hat{y}[k] \approx y[k+1]$; à direita: $\hat{y}[k+2] \approx y[k+3]$.

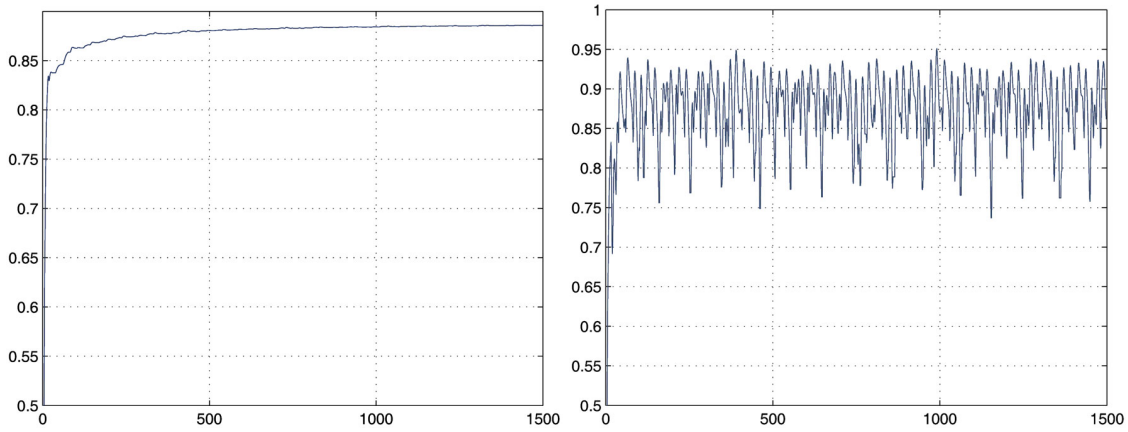



Figura 8.13: À esquerda: ESMSE(0.5,0), direita: ESMSE(0.5,5). Variação de $\alpha$ para a previsão de $\hat{y}[k+2] \approx y[k+3]$ referente à Tab. 8.2.

como entrada, como um caso hipotético, sem correspondência a nenhuma situação real. A Tab. 8.4 apresenta os dados considerados para a experiência, em conjunto com as Tabs. 8.5, 8.6, 8.7 e 8.8.

Antes da apresentação dos resultados, salienta-se que a resolução exposta no capítulo 7, encontra-se simplificada por não contabilizar os dias de viagem, nomeadamente se a viagem entre dois nós durar 1 dia ou mais. Os custos directos são imputados, mas os indirectos, tal como a indisponibilidade da equipa por 1 mês de viagem de barco para chegar ao aero-gerador não foram contabilizados. Porém, em Portugal Continental todas as viagens demoram menos de 8 horas e, portanto, a solução é realista para o nosso país. Esta é a razão pela qual, na matriz apresentada na Tab. 8.7, se considerou a duração da viagem nos caminhos directos como 0.04 dias, ou seja, cerca de 57,6 minutos.

A Tab. 8.9 apresenta as soluções encontradas pelo algoritmo genético secundário na procura da melhor solução para ir de qualquer nó da rede a qualquer nó da rede, com indicação do custo e do caminho a seguir. Encontram-se também incluídos os tempos de execução do algoritmo da formiga e do algoritmo *Dijkstra* para a resolução deste sub-problema.

No cálculo destes custos de trajecto entre dois nós, considerou-se que a melhor solução poderia não ser o caminho directo. Para isso "obrigou-se" o algoritmo genético a procurar uma

| Métr. / Alg. | MSE | TIC | ME | STD | MAE |
|---|---|---|---|---|---|
| ARRSE | 0.0145 | 0.0631 | 0.0033 | 0.1209 | 0.0917 |
| ES(0.5) | 0.0126 | 0.0589 | 0.0012 | 0.1127 | 0.0889 |
| HW | 0.0145 | 0.0629 | 0.0014 | 0.1206 | 0.0940 |
| HWSAS | 0.0309 | 0.0920 | **0.0010** | 0.1765 | 0.1428 |
| ESMSE(0.5,0) | **0.0093** | **0.0504** | 0.0009 | **0.0965** | **0.0746** |
| MAS-2 | 0.0172 | 0.0687 | 0.0021 | 0.1313 | 0.1041 |
| ARMA(2,2) | 0.0120 | 0.0575 | 0.0125 | 0.1086 | 0.0778 |
| SVR-RBF | 0.0658 | 0.1366 | 0.0173 | 0.2557 | 0.2255 |
| SVR-LIN | 0.0323 | 0.0940 | -0.0042 | 0.1795 | 0.1473 |

| Métr. / Alg. | MSE | TIC | ME | STD | MAE |
|---|---|---|---|---|---|
| ARRSE | 0.0360 | 0.0986 | 0.0044 | 0.1902 | 0.1532 |
| ES(0.5) | 0.0298 | 0.0897 | 0.0023 | 0.1726 | 0.1348 |
| HW | 0.0355 | 0.0975 | 0.0025 | 0.1885 | 0.1501 |
| HWSAS | 0.0533 | 0.1204 | **0.0018** | 0.2316 | 0.1878 |
| ESMSE(0.5,0) | **0.0262** | **0.0840** | 0.0019 | **0.1619** | 0.1261 |
| MAS-2 | 0.0364 | 0.0993 | 0.0036 | 0.1910 | 0.1497 |
| ARMA(2,2) | 0.0295 | 0.0903 | 0.0243 | 0.1699 | **0.1250** |
| SVR-RBF | 0.0845 | 0.1548 | 0.0181 | 0.2900 | 0.2568 |
| SVR-LIN | 0.0531 | 0.1198 | -0.0100 | 0.2301 | 0.1904 |

Tabela 8.3: Resultados da previsão para a série *Mackey-Glass*. À esquerda: previsão para $\hat{y}[k+4] \approx y[k+5]$, à direita: $\hat{y}[k+6] \approx y[k+7]$.

| Nó / Dados | Nó 1 | Nó 2 | Nó 3 | Nó 4 | Nó 5 | Nó 6 | Nó 7 | Nó 8 | Nó 9 |
|---|---|---|---|---|---|---|---|---|---|
| Potência(MW) | 0 | 2 | 3 | 1 | 0 | 3 | 0.5 | 2 | 1 |
| nDiasPraAvaria | Inf | 8 | Inf | Inf | Inf | 15 | 6 | 4 | 0 |
| nDiasReparação | 0 | 3 | 0 | 0 | 0 | 4 | 3 | 4 | 6 |
| taxaPerdaPorReparar | 0 | 0 | 0 | 0 | 0 | 0 | 0.4 | 0 | 0 |
| nósAPernoitar | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| eurosPorNoite | 60 | 0 | 0 | 0 | 50 | 0 | 0 | 0 | 0 |
| eurosPorMW (h) | | | | | | | | | |
| 1º dia | 0 | 76 | 76 | 94 | 0 | 92 | 99 | 79 | 89 |
| 2º dia | 0 | 76 | 76 | 94 | 0 | 92 | 99 | 79 | 89 |
| Outros | 0 | 76 | 76 | 76 | 0 | 85 | 90 | 76 | 87 |
| Nº de Horas de Produção | | | | | | | | | |
| 1º Dia | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |
| 2º Dia | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 | 12 |

Tabela 8.4: Dados iniciais do problema de optimização.

| Origem / Destino | Nó 1 | Nó 2 | Nó 3 | Nó 4 | Nó 5 | Nó 6 | Nó 7 | Nó 8 | Nó 9 |
|---|---|---|---|---|---|---|---|---|---|
| Nó 1 | 0 | 20 | Inf | 40 | Inf | Inf | 60 | Inf | Inf |
| Nó 2 | 20 | 0 | 60 | 30 | 40 | Inf | Inf | Inf | Inf |
| Nó 3 | Inf | 60 | 0 | Inf | 20 | Inf | Inf | Inf | Inf |
| Nó 4 | 40 | 30 | Inf | 0 | 40 | Inf | 30 | 80 | Inf |
| Nó 5 | Inf | 40 | 20 | 40 | 0 | 30 | Inf | 60 | 20 |
| Nó 6 | Inf | Inf | Inf | Inf | 30 | 0 | Inf | Inf | Inf |
| Nó 7 | 60 | Inf | Inf | 30 | Inf | Inf | 0 | 50 | Inf |
| Nó 8 | Inf | Inf | Inf | 80 | 60 | Inf | 50 | 0 | Inf |
| Nó 9 | Inf | Inf | Inf | Inf | 20 | Inf | Inf | Inf | 0 |

Tabela 8.5: Dados: distância em quilómetros entre cada Nó (Ligação directa).

| Origem / Destino | Nó 1 | Nó 2 | Nó 3 | Nó 4 | Nó 5 | Nó 6 | Nó 7 | Nó 8 | Nó 9 |
|---|---|---|---|---|---|---|---|---|---|
| Nó 1 | 0 | 1 | Inf | 1 | Inf | Inf | 1 | Inf | Inf |
| Nó 2 | 1 | 0 | 1 | 1 | 1 | Inf | Inf | Inf | Inf |
| Nó 3 | Inf | 1 | 0 | Inf | 1 | Inf | Inf | Inf | Inf |
| Nó 4 | 1 | 1 | Inf | 0 | 1 | Inf | 1 | 1 | Inf |
| Nó 5 | Inf | 1 | 1 | 1 | 0 | 1 | Inf | 1 | 1 |
| Nó 6 | Inf | Inf | Inf | Inf | 1 | 0 | Inf | Inf | Inf |
| Nó 7 | 1 | Inf | Inf | 1 | Inf | Inf | 0 | 1 | Inf |
| Nó 8 | Inf | Inf | Inf | 1 | 1 | Inf | 1 | 0 | Inf |
| Nó 9 | Inf | Inf | Inf | Inf | 1 | Inf | Inf | Inf | 0 |

Tabela 8.6: Dados: montante pago por quilómetro entre cada Nó (Ligação directa).

| Origem \ Destino | Nó 1 | Nó 2 | Nó 3 | Nó 4 | Nó 5 | Nó 6 | Nó 7 | Nó 8 | Nó 9 |
|---|---|---|---|---|---|---|---|---|---|
| Nó 1 | 0 | 0.04 | 0 | 0.04 | 0 | 0 | 0.04 | 0 | 0 |
| Nó 2 | 0.04 | 0 | 0.04 | 0.04 | 0.04 | 0 | 0 | 0 | 0 |
| Nó 3 | 0 | 0.04 | 0 | 0 | 0.04 | 0 | 0 | 0 | 0 |
| Nó 4 | 0.04 | 0.04 | 0 | 0 | 0.04 | 0 | 0.04 | 0.04 | 0 |
| Nó 5 | 0 | 0.04 | 0.04 | 0.04 | 0 | 0.04 | 0 | 0.04 | 0.04 |
| Nó 6 | 0 | 0 | 0 | 0 | 0.04 | 0 | 0 | 0 | 0 |
| Nó 7 | 0.04 | 0 | 0 | 0.04 | 0 | 0 | 0 | 0.04 | 0 |
| Nó 8 | 0 | 0 | 0 | 0.04 | 0.04 | 0 | 0.04 | 0 | 0 |
| Nó 9 | 0 | 0 | 0 | 0 | 0.04 | 0 | 0 | 0 | 0 |

Tabela 8.7: Dados: número de Dias de viagem (Ligação directa).

| Origem \ Destino | Nó 1 | Nó 2 | Nó 3 | Nó 4 | Nó 5 | Nó 6 | Nó 7 | Nó 8 | Nó 9 |
|---|---|---|---|---|---|---|---|---|---|
| Nó 1 | 0 | 2.5 | 0 | 2.5 | 0 | 0 | 2.5 | 0 | 0 |
| Nó 2 | 2.5 | 0 | 2.5 | 2.5 | 2.5 | 0 | 0 | 0 | 0 |
| Nó 3 | 0 | 2.5 | 0 | 0 | 2.5 | 0 | 0 | 0 | 0 |
| Nó 4 | 2.5 | 2.5 | 0 | 0 | 2.5 | 0 | 2.5 | 2.5 | 0 |
| Nó 5 | 0 | 2.5 | 2.5 | 2.5 | 0 | 2.5 | 0 | 2.5 | 2.5 |
| Nó 6 | 0 | 0 | 0 | 0 | 2.5 | 0 | 0 | 0 | 0 |
| Nó 7 | 2.5 | 0 | 0 | 2.5 | 0 | 0 | 0 | 2.5 | 0 |
| Nó 8 | 0 | 0 | 0 | 2.5 | 2.5 | 0 | 2.5 | 0 | 0 |
| Nó 9 | 0 | 0 | 0 | 0 | 2.5 | 0 | 0 | 0 | 0 |

Tabela 8.8: Dados: montante pago por cada dia de viagem (Ligação directa).

| Custo | Genético | Formiga | *Dijkstra* | Nó Origem | Nó Destino | 1º Nó | 2º Nó | 3º Nó | 4º Nó |
|---|---|---|---|---|---|---|---|---|---|
| 80.3 | 0.1101584 | 0.7310512 | 0.0000001 | 1 | 9 | 1 | 2 | 5 | 9 |
| 80.3 | 0.1101584 | 0.4506480 | 0.0000001 | 9 | 1 | 9 | 5 | 2 | 1 |
| 20.1 | 0.1001440 | 0.6309072 | 0.0000001 | 5 | 9 | 5 | 9 | - | - |
| 20.1 | 0.1101584 | 0.1301872 | 0.0000001 | 9 | 5 | 9 | 5 | - | - |
| 90.3 | 0.1402016 | 0.6609504 | 0.0000001 | 1 | 6 | 1 | 2 | 5 | 6 |
| 90.3 | 0.1101584 | 0.4506480 | 0.0000001 | 6 | 1 | 6 | 5 | 2 | 1 |
| 30.1 | 0.1101584 | 0.5507920 | 0.0000001 | 5 | 6 | 5 | 6 | - | - |
| 30.1 | 0.1101584 | 0.1301872 | 0.0000001 | 6 | 5 | 6 | 5 | - | - |
| 20.1 | 0.1101584 | 0.2002880 | 0.0000001 | 1 | 2 | 1 | 2 | - | - |
| 20.1 | 0.1101584 | 0.2503600 | 0.0000001 | 2 | 1 | 2 | 1 | - | - |
| 40.1 | 0.1101584 | 0.3004320 | 0.0000001 | 5 | 2 | 5 | 2 | - | - |
| 40.1 | 0.1101584 | 0.3605184 | 0.0000001 | 2 | 5 | 2 | 5 | - | - |
| 60.1 | 0.1101584 | 0.4005760 | 0.0000001 | 1 | 7 | 1 | 7 | - | - |
| 60.1 | 0.1101584 | 0.3905616 | 0.0000001 | 7 | 1 | 7 | 1 | - | - |
| 70.2 | 0.1101584 | 0.4806912 | 0.0000001 | 5 | 7 | 5 | 4 | 7 | - |
| 70.2 | 0.1001440 | 0.3505040 | 0.0000001 | 7 | 5 | 7 | 4 | 5 | - |
| 110.2 | 0.1101584 | 0.4806912 | 0.0000001 | 1 | 8 | 1 | 7 | 8 | - |
| 110.2 | 0.1101584 | 0.5207488 | 0.0000001 | 8 | 1 | 8 | 7 | 1 | - |
| 60.1 | 0.1101584 | 0.5708208 | 0.0000001 | 5 | 8 | 5 | 8 | - | - |
| 60.1 | 0.1101584 | 0.3304752 | 0.0000001 | 8 | 5 | 8 | 5 | - | - |
| 60.2 | 0.1101584 | 0.3605184 | 0.0000001 | 5 | 1 | 5 | 2 | 1 | - |

Tabela 8.9: Solução encontrada pelo algoritmo genético secundário, para o melhor caminho entre um nó origem e um nó destino. Tempos de execução em segundos para os três algoritmos.

| Ind. | Custo | Nº Solu. | Nº Itera. | Nº Indiv. | Usa 1º | Solução |
|---|---|---|---|---|---|---|
| 1 | 10,443.10 | 1 | 1 | 1 | 1 | 1 1 8 5 9 5 8 5 9 5 8 5 9 5 8 5 9 5 9 5 7 1 2 1 9 5 2 1 7 1 2 1 7 1 6 5 1 1 6 5 1 1 6 5 1 1 6 5 1 1 1 1 |
| 2 | 10,362.90 | 2 | 350 | 100 | 1 | 1 1 8 5 9 5 8 5 9 5 8 5 9 5 8 5 9 5 7 1 9 5 2 1 9 5 2 1 7 1 7 1 2 1 6 5 1 1 6 5 1 1 6 5 1 1 6 5 1 1 1 1 |
| 3 | 10,362.90 | 2 | 500 | 150 | 1 | 1 1 8 5 9 5 8 5 9 5 8 5 9 5 8 5 9 5 7 1 9 5 2 1 9 5 2 1 7 1 2 1 7 1 6 5 1 1 6 5 1 1 6 5 1 1 6 5 1 1 1 1 |
| 4 | 10,362.90 | 3 | 200 | 100 | 1 | 1 1 8 5 9 5 8 5 9 5 8 5 9 5 9 5 8 5 7 1 9 5 2 1 9 5 2 1 7 1 7 1 2 1 6 5 1 1 6 5 1 1 6 5 1 1 6 5 1 1 1 1 |
| 5 | 10,362.90 | 10 | 12500 | 2056 | 0 | 1 1 8 5 9 5 9 5 8 5 9 5 8 5 8 5 9 5 7 1 9 5 2 1 9 5 2 1 7 1 7 1 2 1 1 1 6 5 1 1 6 5 1 1 6 5 1 1 6 5 1 1 1 1 |
| 6 | 10,696.80 | 17 | 400 | 100 | 0 | 1 1 9 5 8 5 8 5 9 5 9 5 8 5 9 5 8 5 2 1 9 5 2 1 9 5 7 1 6 5 2 1 7 1 6 5 7 1 6 5 1 1 6 5 1 1 1 1 1 1 1 1 |
| 7 | 10,696.80 | 15 | 5000 | 150 | 0 | 1 1 8 5 9 5 8 5 9 5 8 5 9 5 8 5 9 5 2 1 9 5 2 1 9 5 2 1 7 1 6 5 7 1 6 5 7 1 6 5 1 1 6 5 1 1 1 1 1 1 1 1 |
| 8 | 10,706.80 | 11 | 350 | 100 | 0 | 1 1 8 5 9 5 8 5 9 5 9 5 8 5 9 5 8 5 6 5 9 5 2 1 9 5 2 1 7 1 7 1 2 1 7 1 6 5 1 1 6 5 1 1 6 1 1 1 1 1 1 1 |
| 9 | 11,396.80 | 16 | 400 | 100 | 0 | 1 1 2 1 8 5 8 5 9 5 8 5 9 5 9 5 8 5 9 5 7 1 9 5 2 1 9 5 7 1 7 1 2 1 6 5 1 1 6 5 1 1 6 5 1 1 6 5 1 1 1 1 |
| 10 | 11,722.50 | 24 | 400 | 100 | 0 | 1 1 8 5 2 1 9 5 8 5 9 5 8 5 8 5 7 1 9 5 7 1 9 5 7 1 9 5 2 1 9 5 2 1 6 5 1 1 6 5 1 1 6 5 1 1 6 5 1 1 1 1 |

Tabela 8.10: Soluções encontradas para várias simulações, com diferentes parâmetros para: número de gerações, número de indivíduos e inclusão de uma solução inicial pseudo-óptima.

solução, independentemente de existir um caminho directo, razão pela qual na Tab. 8.9 constam, por exemplo, caminhos entre os nós 2-1, 5-2, etc. Face aos dados iniciais do problema chegou-se à conclusão de que o melhor caminho entre esses nós era o caminho directo mas, caso não fosse, seria procurado o caminho alternativo mais eficiente e descartado o caminho directo entre os nós.

A Tab. 8.10 apresenta-nos os resultados para várias simulações em função dos dados iniciais do problema, apresentados nas Tabs. 8.4, 8.5, 8.6, 8.7 e 8.8.

A solução com índice nº 1, foi obtida a partir do algoritmo 7.1 descrito na pág. 120. De acordo com o valor do custo da tabela, ainda não se encontra no ponto óptimo. A coluna "Usa 1º" indica sempre que esta solução é incluída na primeira geração de indivíduos do algoritmo genético. Assim, de acordo com esta coluna, as soluções nº 2, 3 e 4, foram obtidas com a inclusão do indivíduo descrito na solução nº1. Significa que o algoritmo genético "pegou" nesta solução e tentou evoluí-la para encontrar melhores soluções. A coluna "NºSolu." indica exactamente quantas soluções óptimas foram encontradas até se chegar à que se encontra apresentada (inclusive).

As restantes soluções foram obtidas apenas com o algoritmo genético, sem inclusão de nenhuma solução específica inicial.

Observando as dez simulações, pode-se concluir que o algoritmo genético por si só consegue encontrar a solução óptima, mas necessita de algum esforço computacional. Isso é claro através dos resultados obtidos nas soluções nº 6, 7, 8, 9 e 10, onde, com um número de gerações e indivíduos diferentes se obtiveram resultados diferentes. Não esquecendo o facto de ser uma solução de pesquisa aleatória, direccionada, só haverá 100% de certeza de chegar ao ponto óptimo com infinitas iterações. A solução nº 5 confirma parcialmente este aspecto, apesar de não ser conclusivo, devido ao número de soluções parciais óptimas ser mais baixo do que no caso das soluções expostas abaixo na tabela. Uma conclusão óbvia é o facto de o algoritmo genético ter conseguido evoluir a solução nº 1, mas só depois de introduzidas as modificações indicadas na secção 7.2.4, pág. 120, com um esforço computacional muito reduzido.

## 8.2 Conclusões

Este capítulo apresentou alguns dos principais resultados da dissertação. Sendo o *software* SMIT um elemento basilar da dissertação é, por isso mesmo, muito extenso e complexo, pelo que é apresentado em anexo. A integração dos vários componentes de *hardware* e de *software* e entre

estes e o SMIT mostraram constituir-se como um sistema coerente e adequado a responder aos problemas referentes à manutenção condicionada de parques eólicos. Estes aspectos são ainda reforçados pelas características particulares do sistema de *hardware* inerentes ao seu baixo consumo, o que lhe dá autonomia do ponto de vista eléctrico, através da inclusão de painéis fotovoltaicos e baterias.

Outro aspecto relevante é o do sistema de sincronização de relógios entre dois processadores o qual apresentou um bom desempenho e para o tipo de sinais que é necessário adquirir num aerogerador mostra-se uma solução promissora e competitiva. Em resumo, existe uma diferença de 42,5 $\mu$s entre o relógio central e o relógio dos nós escravos, garantindo-se, portanto, uma aquisição praticamente síncrona. O erro equivale a um desfasamento de uma amostra numa aquisição de 23,529 kHz. Quer a simulação quer os resultados reais, coincidem nos principais aspectos apontados ao longo do texto.

Relativamente ao problema da optimização da sequência das acções de manutenção nos diferentes aero-geradores pelas várias equipas, o algoritmo apresentado deverá, em desenvolvimentos futuros, ser alvo de novas abordagens conducentes a uma mais ampla aplicação do mesmo. Fica ainda demonstrada a utilidade dos algoritmos genéticos em problemas de optimização estocástica.

Do ponto de vista dos resultados gerais, a globalidade do sistema encontra-se bem conjugado e de forma coerente. Face ao tipo de desenvolvimento efectuado nos planos da algoritmia, do *software*, bem como do *hardware* e à simplicidade da sua implementação, a evolução futura dos mesmos torna-se relativamente acessível.

# Capítulo 9

# Conclusões e Trabalho Futuro

Este capítulo apresenta as conclusões da presente dissertação e um conjunto de propostas para desenvolvimentos futuros.

## 9.1 Conclusões

Os objectivos da presente dissertação foram integralmente cumpridos, quer ao nível do *hardware* quer do *software* e, em particular, do sistema de informação e dos algoritmos de suporte desenvolvidos.

No que respeita ao sistema de informação, SMIT, este foi re-implementado em plataforma *open-source* tendo-lhe sido adicionadas novas funcionalidades, nomeadamente no âmbito da manutenção preditiva, incluindo *drivers* adicionais conducentes à compatibilização das novas funcionalidades, revelando-se, no final, uma ferramenta ainda mais atractiva para unir mais investigadores nesta área. Destacam-se ainda algumas inovações, designadamente, a possibilidade de relacionamento dos objectos de Manutenção numa configuração próxima da matricial e a forma como os algoritmos de condição de funcionamento estão integrados neste módulo.

A nível de relação com os utilizadores do SMIT, foram criados vários módulos para instalação no sistema operativo *Windows*, destacando-se os módulos de Objectos de Manutenção, Técnicos, Peças de Reserva, Clientes de Objectos de Manutenção, Fornecedores, Ferramentas, Ordens de Trabalho, Pedidos de Intervenção, Diagnóstico de Avarias, Planos de Manutenção, Módulo de *Gantt* e Importador/Exportador de Dados.

Para além destes módulos, foi implementado um módulo adicional de recepção de dados, executado no servidor SMIT, para onde todos os módulos de aquisição enviam os dados, recebidos via TCP/IP da *gateway Ethernet-CAN*, oriundos da rede de campo *CAN*.

O sistema de aquisição de dados *low-cost* implementado revelou-se muito eficaz. Esta opção foi assumida como adequada e inovadora desde o início do desenvolvimento da presente dissertação. Esta direcção é actualmente alvo de investigação intensa na área de redes de sensores sem

fios. Os módulos são fiáveis, de baixo custo e de baixo consumo, podendo, inclusive, ser alimentados por sistemas de painéis solares. A importância desta funcionalidade relaciona-se com uma das taxas de avarias mais sistemáticas nos aero-geradores que é devida à falha metódica de alguns dos sistemas de controlo, designadamente dos processadores de autómatos.

Foi previsto readaptar os PCBs para integração em caixas de automação, à semelhança das caixas *Phoenix* modelo ME 22,5 OT-MSTB0. Porém, como o objectivo de uma dissertação desta natureza não é desenvolver soluções finais, mas sim protótipos que antecipem soluções utilizáveis no futuro pela indústria, essa será uma tarefa a ser concretizada numa possível implementação de campo.

Outros contributos incluídos nesta dissertação situam-se ao nível dos algoritmos implementados, que se mostraram eficazes e, que em todos, foi acrescentada mais-valia relativamente ao estado-da-arte, designadamente os seguintes:

- Conjunção de modelos, baseados em séries temporais, ARRSE, ES, HW, HWSAS, ESM-SE, MAS, ARMA, ARIMA, SVM, SVR-RBF e SVM-LIN, utilizando dados dos aero-geradores, aos quais são aplicados critérios de minimização do erro de predição, tais como, o MSE, RMSE, MAE, MAPE, CORRE, e TIC para predição da próxima intervenção;

- Adequação de um modelo apoiado no algoritmo de optimização baseado no comportamento da formiga e em algoritmos genéticos para racionalização da programação das intervenções de manutenção nos parques eólicos, designadamente no que concerne às deslocações das equipas de manutenção.

A sincronização dos instantes de aquisição de dados nos diferentes equipamentos distribuídos geograficamente, através de protocolos de distribuição de RTC via rede IP, permite relacionar precisamente os dados obtidos nos diferentes pontos da rede de campo. Este aspecto é concretizado através da utilização do protocolo PTP.

Os resultados apresentados no capítulo 8 utilizaram dados obtidos através do INEGI, sediado no *campus* da FEUP, referentes a um parque eólico do norte do país. Estes dados necessitaram de pré-processamento para serem utilizados em alguns dos algoritmos cujos resultados são indicados no capítulo 8.

O conjunto de peças do *puzzle* que representa a presente dissertação mostrou ser o desenho adequado para abordar a diversidade e complexidade de variáveis que envolve a manutenção condicionada de parques eólicos.

## 9.2 Contributos

Dos vários aspectos descritos na presente dissertação, destacam-se como principais contributos os seguintes pontos:

- A globalidade do sistema e a sua arquitectura;
- A utilização de séries temporais para detecção da curva de degradação;
- A alteração do algoritmo do Alisamento Exponencial;

- O método de optimização aplicado à determinação da sequência de visitas de manutenção, não existe na literatura.

Do ponto de vista de contributos científicos, a presente dissertação contribuíu com:

- 5 artigos em conferências Internacionais indexados no ISI;
- 1 artigo em revista Internacional indexado no ISI;
- 1 artigo em revista de Manutenção Internacional;
- 2 artigos em conferências de Manutenção Internacionais;
- 2 artigos em conferências da APMI;
- 1 capítulo num Livro Internacional;
- 1 artigo indexado no *IEEE Explorer*;
- 1 artigo em conferência Nacional;
- 1 artigo em conferência Internacional;

aos quais se adiciona os contributos em parceria com a restante equipa de investigação, nomeadamente:

- 6 artigos em conferências Internacionais indexados no ISI;
- 1 artigo em revista Internacional indexado no ISI;
- 3 artigos em conferências de Manutenção Internacionais;
- 3 artigos em conferências da APMI.

## 9.3 Desenvolvimentos Futuros

Um dos primeiros desenvolvimentos futuros deste trabalho, será refazer todos os programas do SMIT da plataforma *Windows* para uma plataforma via *Browser*. Neste momento a base de dados do SMIT encontra-se suficientemente consolidada para permitir desenvolver em PHP todos os *interfaces* dos módulos do SMIT em ambiente *Windows* e, assim, disponibilizá-los via *Web*.

Outro desenvolvimento é re-projectar os módulos de *hardware* de modo a adaptar as suas dimensões a caixas próprias, para facilitar a sua instalação em quadros de automação. Neste ponto seria interessante alterar o módulo *gateway* para incluir o controlador ENC424J600/ENC624/600 da *Microchip* (recentemente lançado pelo fabricante). A proposta justifica-se pelas dificuldades e algumas limitações encontradas no controlador de rede ENC28J60, facto identificado pelo próprio fabricante. A nova versão, o ENC424J600, colmatou as lacunas do predecessor e introduziu um *interface* para 100Mbps com negociação *full-duplex* e *half-duplex* com os *routers/switchs*.

Ainda neste âmbito, como desenvolvimentos futuros, sugerem-se os seguintes:

- Implementação, em FPGA, do *hardware* de recepção de pacotes *Ethernet* com inclusão de registo temporal;

- Inclusão do circuito integrado de alto desempenho - DP83640 *Precision Phyter* - desenvolvido pela *National* - que funciona como *transceiver* de meio físico da *Ethernet*, incorporando as facilidades de *timestamping*;

- Adopção generalizada dos vários contributos que vários fabricantes começam a integrar nos seus componentes (Setembro, 2009), associados ao protocolo PTP, designadamente a *Arasan* com circuitos integrados para velocidades de 100Mbps e 1Gbps, para o segmento dos processadores rápidos (*Intel*, *Mips*, etc).

Ao nível da arquitectura do SMIT, foi desenvolvido e adaptado um sistema de *routers wireless* (com ou sem ADSL e/ou 3G) para instalar em cada aero-gerador, com o *software* mestre/escravo do SNTP e PTP. A relevância para esta dissertação é relativa porque, normalmente, nos parques eólicos todos os aero-geradores estão interligados por uma rede IP através de fibra óptica. No entanto, nas situações de estudo do regime de ventos (ver Fig. 1.4, pág. 6), o sistema pode ser útil mais uma vez apenas com alimentação a partir de painéis solares e com a vantagem de ser possível sincronizar a aquisição das diversas "torres de medida". Nesta situação, será possível, por exemplo com antenas direccionais (custo de cerca de 50 €), estabelecer canais de comunicação em linha de vista a 11Mbps ou 54Mbps.

No que se refere aos algoritmos, salienta-se o aspecto importante do SMIT estar preparado para facilitar a inclusão de novos algoritmos desde que programados nas linguagens disponibilizadas, *R* e *Octave*. Uma das vertentes possíveis de desenvolvimento é a da adaptação de cadeias de *Markov* e de redes neuronais ao problema da detecção do estado e conjugá-las de forma a optimizar a predição do estado seguinte.

Sugere-se o desenvolvimento de novos algoritmos adaptados aos módulos embebidos, de forma a "aliviar" o processamento no servidor SMIT.

No que se refere à optimização dos percursos entre torres e entre parques eólicos, poderá desenvolver-se de forma mais elaborada o algoritmo proposto no capítulo 7 com o objectivo de optimizar os custos, os recursos, as perdas de produção ou a conjugação de várias variáveis, com o objectivo de flexibilizar a adequação do algoritmo às várias situações concretas. Deverá ainda incluir-se no cálculo das trajectórias óptimas informação GPS e outras afins, que possibilitem um incremento da precisão dos algoritmos.

Uma outra área de desenvolvimento será a da modelização 3D, a qual permitirá simular comportamentos, *upgrades*, resolução de falhas, para além da possibilidade de formação em equipamentos virtuais.

Deverão ainda efectuar-se novos desenvolvimentos através da instalação de câmaras *Web* no interior do aero-gerador para possibilitar uma visita remota à cabine, com possibilidade de intervenção remota.

# Anexo A

# Sistema Integrado de Gestão de Manutenção

Neste anexo é descrito o *software* de Gestão de Manutenção nas suas vertentes de Cliente e Servidor. As potencialidades do sistema desenvolvido ultrapassam largamente as necessidades básicas de uma boa gestão da manutenção ligada à área das energias eólicas (SMIT, [68]),que foi actualizado e melhorado, em consequência dos trabalhos desta dissertação, tal como será descrito neste anexo.

O SMIT é constituído por dois pacotes, o pacote servidor, normalmente instalado num sistema operativo *Unix* e o pacote Cliente, instalado num sistema operativo *Windows*. O pacote servidor pode ser também instalado num sistema operativo *Windows*, embora não tenham existido experiências nesse sentido. Relativamente ao servidor será explicado o procedimento para instalação em dois sistemas operativos de referência na área do *open-source*: *FreeBSD* e *Linux-Slackware*.

O *FreeBSD* segue a filosofia da Universidade de *Berkely* e, nesta linha de *software* encontram-se algumas distribuições interessantes, tais como o *OpenBSD*, *NetBSD* e *PC-BSD*. O *PC-BSD*, é uma distribuição com o objectivo de ser apelativa a indivíduos com poucos conhecimentos de *Unix*, nomeadamente a nível da instalação e da utilização.

Apesar do *Slackware* não ser um sistema operativo de fácil instalação/utilização é, contudo, uma das distribuições com maior longevidade e com uma grande comodidade de escolha dos pacotes de *software* para instalação.

Na área dos sistemas operativos *Unix* com uma filosofia de instalação semelhante ao *Windows* destacam-se ainda duas distribuições nas quais o SMIT pode ser instalado sem modificações no *PC-BSD* e com pequenas alterações ao nível dos *scripts* de instalação no *UBuntu*.

De seguida inicia-se a descrição semi-detalhada do programa SMIT, módulo de cliente, apenas nas vertentes de Objectos de Manutenção (OM), Planos de Manutenção, Ordens de Trabalho (OT) e módulo de *Gantt*. Não serão apresentadas descrições dos outros módulos, designadamente, clientes de OM, fornecedores, técnicos, ferramentas, peças-de-reserva, pedidos de intervenção e diagnóstico de avarias.

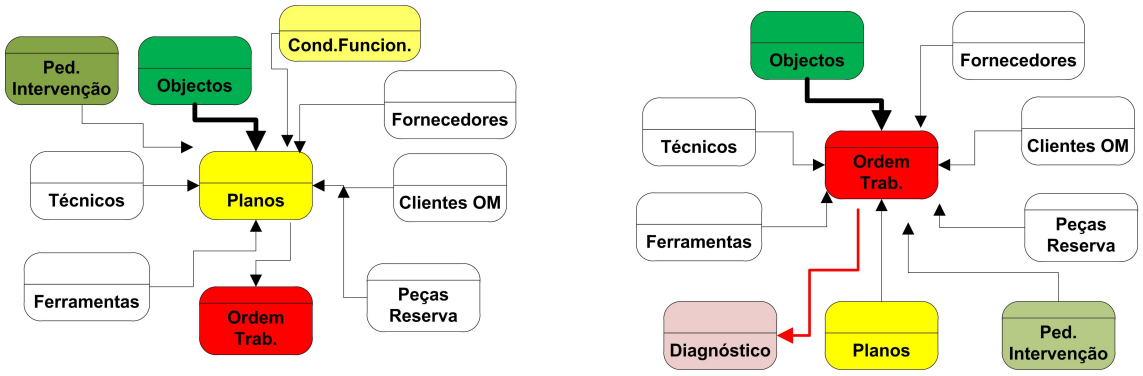



Figura A.1: Planos de Manutenção e OT - Diagrama de interligação dos diferentes módulos.

A Fig. A.1 mostra as principais interligações entre os diferentes módulos do programa ao observar-se o sistema do ponto de vista dos planos de manutenção e/ou das ordens de trabalho.

## A.1 Introdução

Neste secção exemplifica-se o funcionamento do programa SMIT do ponto de vista do utilizador, que se inicia com a abertura da aplicação, onde um utilizador comum poderá usar o programa para efectuar a Gestão da Manutenção de vários Equipamentos/Instalações.

O SMIT V2.0 é um programa cliente servidor, multi-base de dados, permitindo a instalação de vários clientes e a configuração de várias bases de dados na mesma plataforma. O programa é acedido através do ambiente *Windows*.

Como o programa SMIT é multi-utilizador, em determinado instante, pode ocorrer a existência de vários utilizadores a tentarem editar a mesma ficha (ver Fig. A.2). Nesta situação o que abrir primeiro a ficha fica com acesso de alteração. Os outros utilizadores apenas podem consultar a ficha, sem, no entanto, a poderem alterar. Quando um utilizador está a editar uma ficha em modo de leitura, na parte superior da *form* da aplicação é apresentada a cor vermelha onde se encontram os botões, sinalizando assim a impossibilidade de alteração.

O SMIT tem definidos vários níveis de utilizadores, tais como: Administração, Manutenção, Gestão de *Stocks*, Técnicos, Utilizadores, Chefe de Departamento, Chefe de Produção e Chefe de Qualidade. Conforme o nível de acesso do utilizador, assim diferentes funcionalidades estão disponíveis. Compete ao utilizador de nível de Administração, criar utilizadores e indicar o seu nível de acesso. Na presente instalação base está definido o seguinte utilizador:

- **Utilizador:** adm
- ***Password*:** adm

Qualquer utilizador, após entrar no programa pela primeira vez, pode, e deve, alterar a sua *password* através da opção "Alterar *Password*".

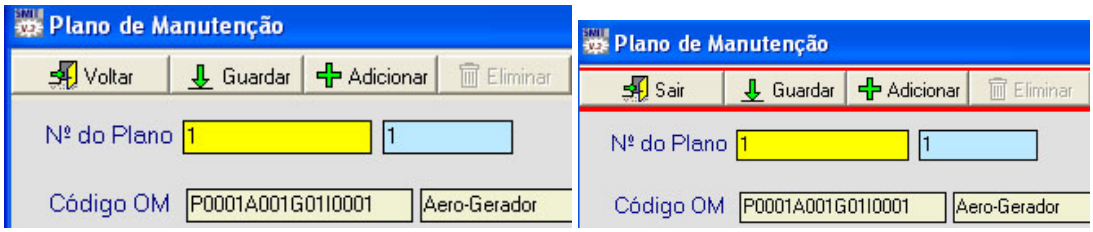



Figura A.2: Edição da mesma ficha por dois utilizadores. À Esquerda: edição da ficha. À direita: apenas acesso de leitura.

## A.2 Diferentes níveis de acesso

Nesta secção descrevem-se os diferentes níveis de acesso ao programa SMIT, indicando-se as funcionalidades existentes em cada um dos níveis nos diversos módulos do programa. Os níveis de acesso são os seguintes:

- **Administração -** tem acesso ao programa com todas as funcionalidades disponíveis. De seguida, o Administrador deverá configurar os utilizadores que vão utilizar o programa, indicando o nível de acesso desses mesmos utilizadores. É boa regra de gestão definir, no mínimo, o gestor de manutenção e um técnico. O Administrador tem ainda uma funcionalidade restrita apenas ao seu nível - a importação de dados;
- **Manutenção -** após iniciar o programa *smit.exe* o utilizador de manutenção pode executar todas as operações excepto as descritas no nível de Administração (criação de utilizadores e importação/exportação de dados);
- **Gestão de *Stocks* -** após iniciar o programa *smit.exe* o utilizador com o nível de "Gestão de *Stocks*", tem apenas autorização para editar Fornecedores e Peças-de-Reserva;
- **Técnicos -** após iniciar o programa *smit.exe* o utilizador com o nível de "Técnico", tem acesso aos pedidos de intervenção e às OTs. No entanto, só pode visualizar as OTs nas quais vai trabalhar ou já trabalhou. Para conseguir listar essas OTs é preciso, em primeiro lugar, que o Responsável de Manutenção associe ao técnico o *Login* do programa SMIT. Outra restrição bastante forte é apenas poder alterar o estado da OT;
- **Utilizadores -** após iniciar o programa *smit.exe* o utilizador com o nível de "Utilizador", tem acesso aos pedidos de intervenção e às OTs. No entanto, só pode visualizar as OTs que foram criadas a partir de pedidos de intervenção feitos por si. O mesmo acontece nos pedidos de intervenção;
- **Chefe de Departamento -** após iniciar o programa *smit.exe* o utilizador com o nível de "Chefe de Departamento" tem acesso aos pedidos de intervenção, OTs e Diagnóstico de Avarias. Só ficam visíveis para o chefe de Departamento os pedidos de intervenção feitos sobre objectos que pertençam ao mesmo departamento. O mesmo para com o módulo das OTs; neste nível o utilizador só tem acesso a criar OTs não planeadas. Como não tem acesso aos planos não os poderá criar, estando esta tarefa apenas acessível ao utilizador com nível de "manutenção". No entanto, se existirem planos para o OM do departamento e forem

geradas OTs, este chefe de Departamento poderá efectuar o seu encerramento e alteração;

- **Chefe de Produção -** após iniciar o programa *smit.exe* o utilizador com o nível de "Chefe de Produção" tem acesso aos pedidos de intervenção, OTs e *Gantt*. Só ficam visíveis para o Chefe de Produção os pedidos de intervenção feitos por si e as ordens de trabalho relacionadas com os pedidos de intervenção associados ao seu utilizador. No módulo de *Gantt*, este utilizador tem acesso ao calendário dos planos das OTs dos pedidos de intervenção e dos planos com OTs planeadas em curso. Pode deslocar graficamente os planos que ainda não tenham OT criada (alterar a data da OT) e bloquear esse plano originando um pedido de intervenção planeado. A vantagem desde processo é adequar o calendário da manutenção planeada ao calendário da produção do equipamento;
- **Chefe de Qualidade -** após iniciar o programa *smit.exe* o utilizador com o nível de "Chefe de Produção" tem acesso aos pedidos de intervenção, OTs e *Gantt*. Só ficam visíveis para o Chefe de Produção os pedidos de intervenção feitos por si e as ordens de trabalho relacionadas com os pedidos de intervenção associados ao seu utilizador. No módulo de *Gantt*, este utilizador tem acesso ao calendário dos planos, das OTs dos pedidos de intervenção e dos planos com OTs planeadas em curso. Dependendo da configuração, pode deslocar a data da próxima OT planeada associada a um plano ainda sem OT;
- **OM em projecto -** o utilizador com este nível de acesso é responsável por cuidar da manutenção dos OMs que estejam em fase de projecto. Isto é, todos os pedidos de manutenção pelos utilizadores normais sobre um OM desta natureza apenas aparecerá aos utilizadores com este nível de acesso. Estão disponíveis os módulos: Clientes de OM, OM, OTs, Pedidos de Intervenção, Diagnóstico de Avarias e Planos de Manutenção. Nesta opção há uma forte restrição, onde o utilizador apenas vê e trabalha com os OMs nesse estado.

O programa tem comportamentos diferentes nos mesmos módulos dependendo do nível de acesso do utilizador. No entanto, face ao âmbito desta dissertação, não se justifica pormenorizar mais detalhes.

## A.3 Módulo de Objectos de Manutenção

Na Fig. A.3 é apresentada a *form* do módulo de OM. Aqui devem ser caracterizados todos os OMs que vão ser alvo de intervenções de manutenção planeada ou não planeada. A caracterização de um OM passa pela indicação das características físicas (dimensões e peso), eléctricas, fornecedor, preço de custo, localização, marca, código de inventário, manuais, foto, etc. O utilizador pode associar aos OMs vários manuais que sejam relevantes, desde ficheiros *Word*, *Excel*, pdf, imagens, texto simples. Também é possível associar uma fotografia ao OM.

Na Fig. A.4 realçam-se as características funcionais e técnicas. Há a salientar nas características funcionais os seguintes campos: valor de aquisição, datas de aquisição, de início de funcionamento e de fim de garantia, as quais podem ser usadas para ajudar a calcular o valor das amortizações e reintegrações do equipamento ao longo dos anos. Na Fig. A.5 estão resumidos

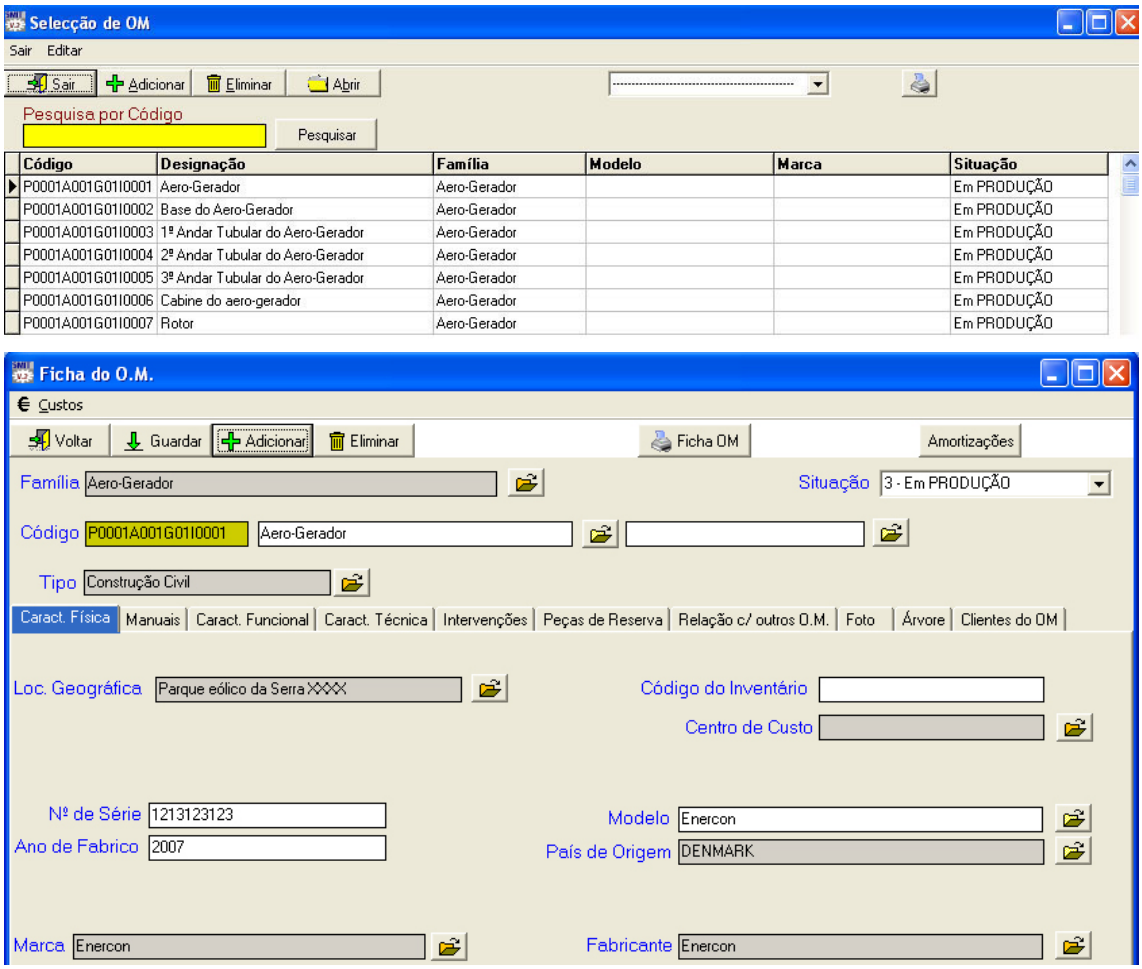



Figura A.3: Em cima: Lista de OM. Em baixo: caracterização de um OM.

os totais das intervenções planeadas e não planeadas, podendo-se assim inferir a eficácia da estratégia de planeamento levada a efeito pelo Departamento de Manutenção. Há ainda a destacar que as peças-de-reserva que ficam associadas a este objecto são associadas ao mesmo, não havendo qualquer cálculo, a não ser a contabilização do total das peças para os OMs (ver módulo de peças-de-reserva). Na Fig. A.6 mostra-se a forma de relacionar os objectos. Nesta versão do SMIT, a relação entre OMs deixou de ser em árvore para ser representada por grafos. O método antigo permitia apenas indicar qual era o OM pai. A vantagem conseguida com esta mudança é permitir relacionar o objecto indicando vários OMs abaixo, vários OMs acima ou relações bi-direccionais (ambos ao mesmo nível), isto é, ligações orientadas ou não orientadas. Dado que com esta abordagem se podem, na prática, efectuar **n** ligações com esta versatilidade, optou-se, na secção "Relação com outro OM", por permitir apenas a edição directa entre o OM em edição e aqueles que lhe estão directamente ligados. O Objecto que está a ser editado é representado pela cor vermelha, sendo todos os outros representados pela cor amarela. As setas de ligação entre dois objectos também têm cores diferentes, tais como, pai-filho com a cor azul, filho-pai com a cor castanha e a cor vermelha para o caso da ligação ser bidireccional.

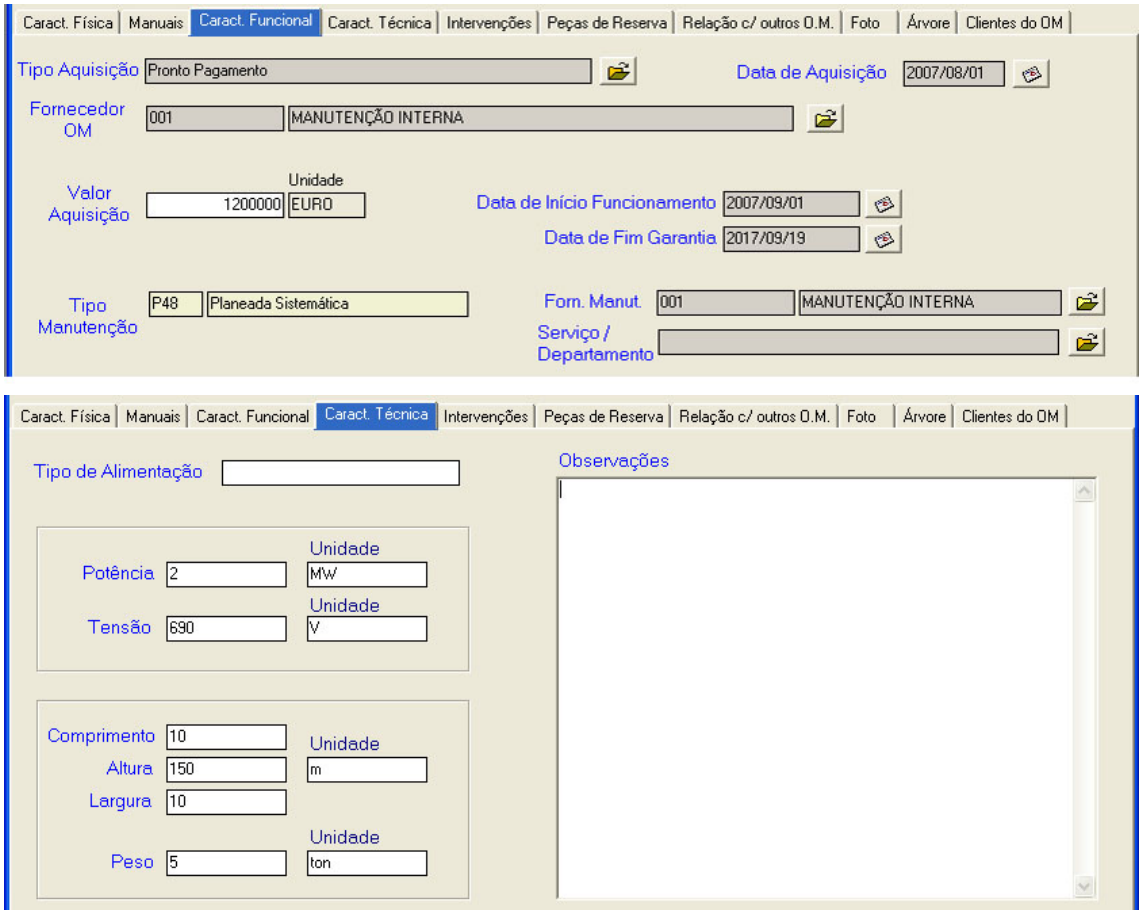



Figura A.4: Caracterização Funcional e Técnica.

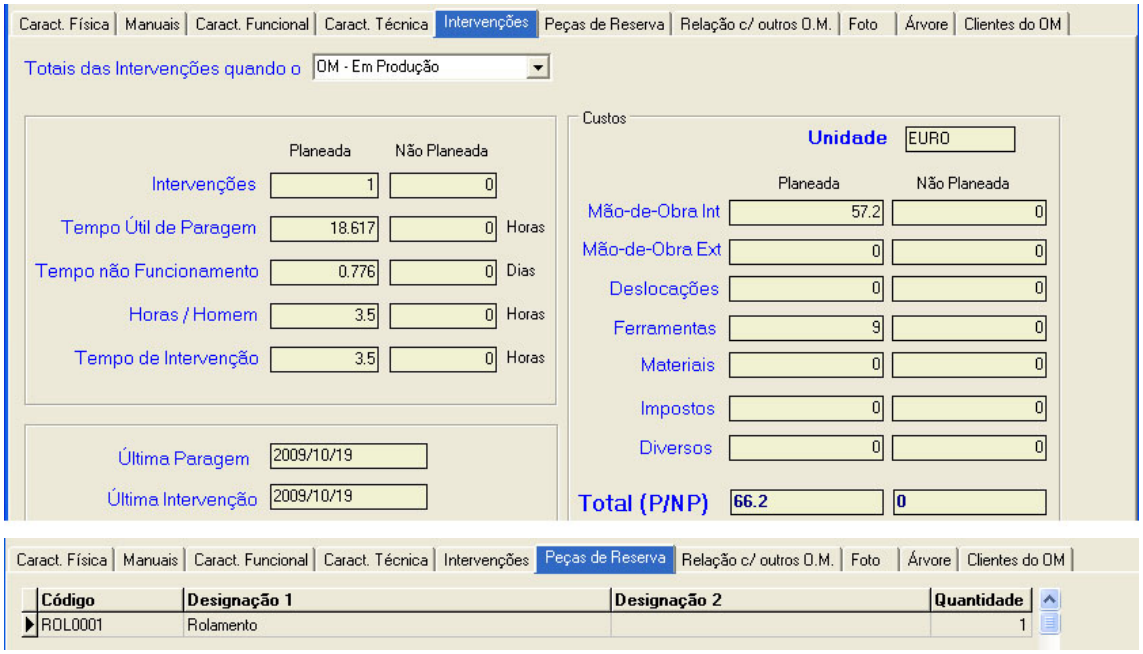



Figura A.5: Em cima: totais das intervenções planeadas e não planeadas; Em baixo: listagem das peças-de-reserva do OM.

Para visualizar as ligações completas a partir do OM em edição, optou-se pela representação em árvore, dado que, do ponto de vista informático, podem-se ter *loops* entre OMs. Assim, sempre que se detecta o início de um *loop*, antes do OM aparece uma seta amarela no formato de circunferência e, quando se detecta o fim do *loop*, é representada uma seta vermelha bi-direccional (normalmente o mesmo objecto aparece duas vezes, uma com a seta amarela e outra com a vermelha). Nas outras relações pai-filho é mostrada uma seta azul direccionada.

Com a estrutura de ligações permitida no programa não existe nenhuma restrição ao número de ligações que se pretendam representar, dando assim uma grande flexibilidade adicional e inovadora neste domínio.

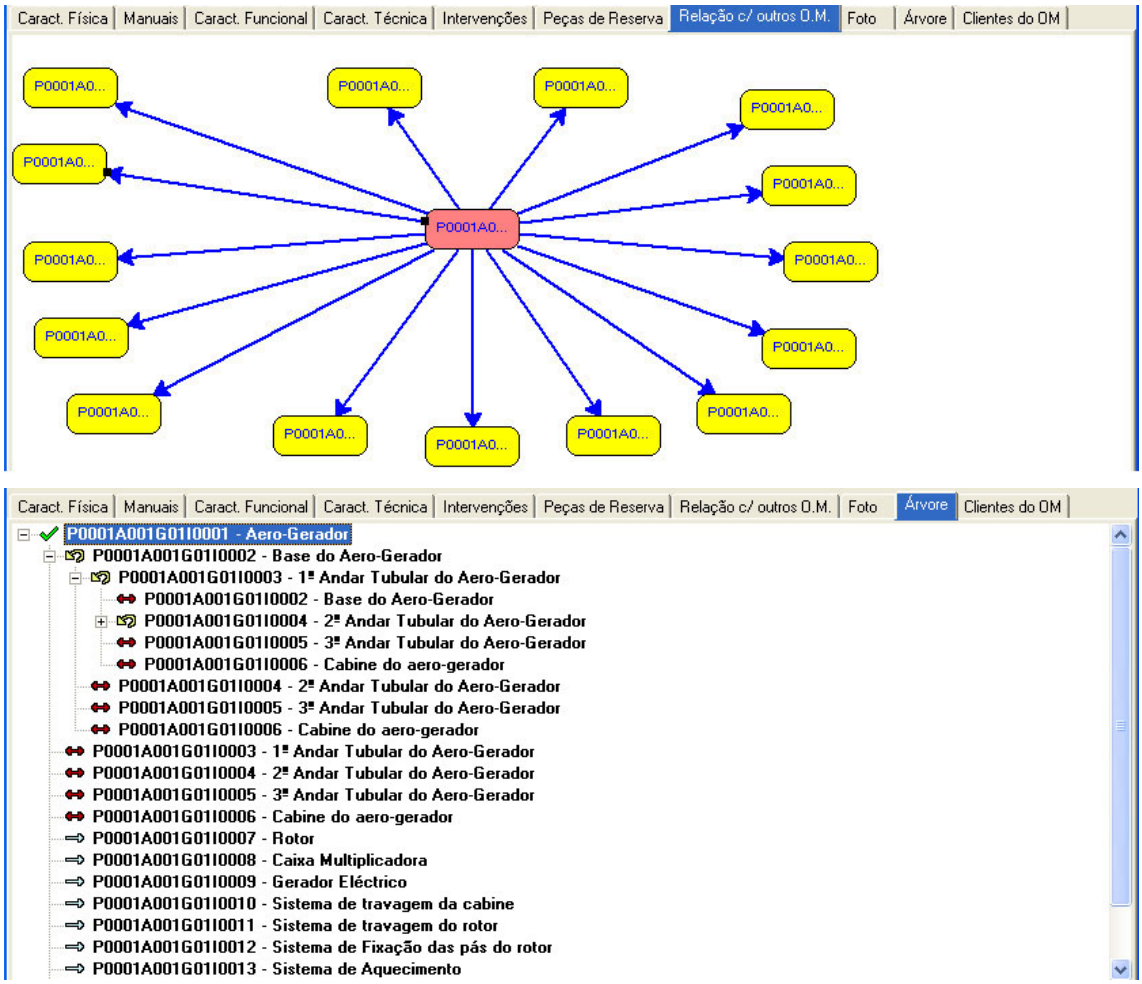



Figura A.6: Em cima: editor de relações entre OMs. Em baixo: árvore de relações do OM.

## A.4 Módulo de Planos de Manutenção

Na Fig. A.7 o utilizador pode escolher o plano que pretende editar/alterar. A simbologia de cores indicada em cada plano tem o seguinte significado:

- **Verde** - indica planos que já têm OT planeada. **Todas as outras cores** indicam planos onde ainda não existe OT planeada. Associada a esta cor está a imagem que tem o mesmo significado;
- **Branco** - a data prevista para a OT é superior a uma semana relativamente à data actual;
- **Amarelo** - a OT que vai ser planeada tem data superior a menos duma semana da data actual;
- **Vermelho** - a OT planeada tem data do dia ou anterior;
- **Cinzento** - planos de manutenção inactivos (ver ficha do plano). Associado a esta cor está a imagem que tem o mesmo significado;
- **Laranja** - não cria OT porque o utilizador indicou no campo "Gerar OT?" a opção "Não". Associado a esta cor está a imagem que tem o mesmo significado. Quando o plano também está inactivo a cor laranja mantêm-se mas a imagem é substituída por .

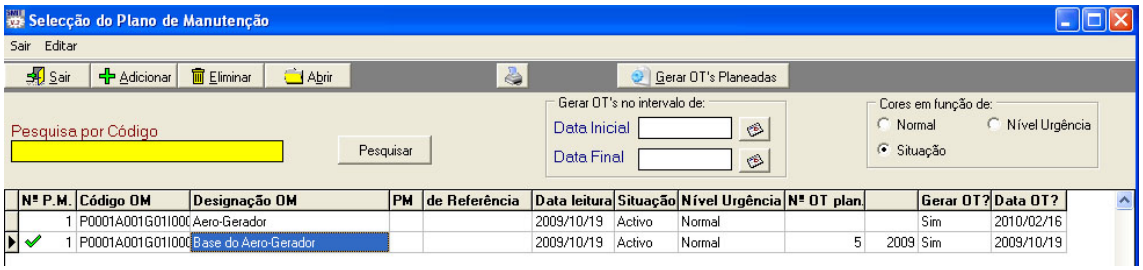



Figura A.7: Selecção do plano de manutenção. As cores podem ser alteradas.

Neste *form* pode ainda efectuar-se o planeamento indicando o intervalo temporal para seleccionar os planos que darão origem à OT. Esse espaço de tempo refere-se à data das OTs. As cores que aparecem na listagem dependem da opção escolhida acima. Salienta-se que, no caso da cor ser dependente do nível de urgência, o utilizador do programa pode escolher a cor do nível de urgência para mais facilmente a identificar.

Na Fig. A.8 encontra-se representada a ficha de um plano de manutenção, onde o utilizador pode escolher o OM para o qual pretende efectuar uma manutenção planeada. Para realizar a escolha nas opções de planeamento pode optar por duas situações: dependente de outro PM ou autónomo:

- **Planeamento dependente de outro PM** - indica no campo "PM de referência", de qual plano se pretende que o actual PM fique dependente; apenas podem ser escolhidos PMs já criados. O planeamento será efectuado com base nas datas de planos do PM de referência;
- **Planeamento autónomo** - neste caso há necessidade de preencher vários campos, nomeadamente:
  - ⇒ Fornecedor de Manutenção - a indicação do fornecedor de manutenção permite associar à OT criada esse mesmo fornecedor, ao qual, mais tarde, serão somados os custos quando a OT for encerrada. Assim, é possível saber-se qual o nível de serviços fornecido por esse fornecedor;

- ⇒ Nível de urgência - todas as OT criadas a partir do plano ficam com este nível de urgência;
- ⇒ Tipo de manutenção - todas as OT criadas a partir do plano ficam com este tipo de manutenção;
- ⇒ Designação de PDF e unidade - Descrição da variável que se vai controlar (tempo, quilómetros,...);
- ⇒ Tipo - Indica o algoritmo de planeamento com as seguintes opções:
  - → ***P-Patamar:*** Próximo valor = "Próximo Valor" antigo + incremento;
  - → ***C-Constante:*** Próximo valor = Valor do PDF indicado pelo utilizador + incremento;
  - → ***A-Temporal-Patamar:*** próxima data = data actual + incremento;
  - → ***T-Temporal-Constante:*** Próxima data = data indicada pelo utilizador + incremento;
- ⇒ Média Prevista / dias - indicação inicial da média prevista para o parâmetro PDF;
- ⇒ Desvio Padrão - desvio padrão da média (indicada apenas inicialmente);
- ⇒ Incremento- indica o intervalo de unidades PDF entre intervenções de manutenção;
- ⇒ Última leitura - valor do PDF na data;
- ⇒ Próximo valor - indica em que valor do PDF deve ser efectuada a intervenção, indexando-a à correspondente data de calendário;
- ⇒ Tolerância em Dias - indica quantos dias podem ser utilizados para efectuar as acções do plano (quando já foi gerada a OT).

Associado a cada plano podem ser indicados os técnicos que vão efectuar os trabalhos de manutenção. Os custos são os indicados na ficha de técnico, não podendo ser alterado a nível dos planos de Manutenção. Na Fig. A.9 indicam-se as ferramentas e as peças-de-reserva, assim como a mão-de-obra. Nas peças-de-reserva pode ser indicado se a peça é externa ou não. Quando as peças são externas, não serão descontadas do *stock*, tudo se passa como se um técnico externo viesse fazer a manutenção trazendo a peça consigo. Na Fig. A.10 encontram-se representados os custos previsíveis da execução planeada das OTs com base no plano definido. São ainda apresentadas as acções que devem ser executadas para efectuar a referida manutenção. Toda a mão-de-obra, peças-de-reserva, ferramentas e acções a efectuar são automaticamente incluídas nas OTs que são criadas a partir dos planos de manutenção. Esta é uma forma de facilitar a automatização procedimental da manutenção, dado que o chefe da manutenção só necessita de verificar as datas e fazer o planeamento de acordo com o seu plano. Este apenas necessita criar o plano uma única vez e, a partir daí, não é necessário efectuar mais alterações. Devido à possibilidade de se poderem indicar se as peças-de-reserva / ferramentas / mão-de-obra são externas / internas, na atribuição desses custos a fornecedores, torna-se necessário indicar quem são os fornecedores internos/externos. O programa considera, por defeito, que o fornecedor interno é o "**001-Manutenção Interna**". Na Fig. A.10, do lado direito, pode-se indicar no PM quais são os fornecedores e onde devem ser contabilizados os custos internos e os custos externos. Assim, quando uma OT for planeada automaticamente ficará com as atribuições descritas no plano que lhe deu origem. A Fig. A.11 apresenta a atribuição de

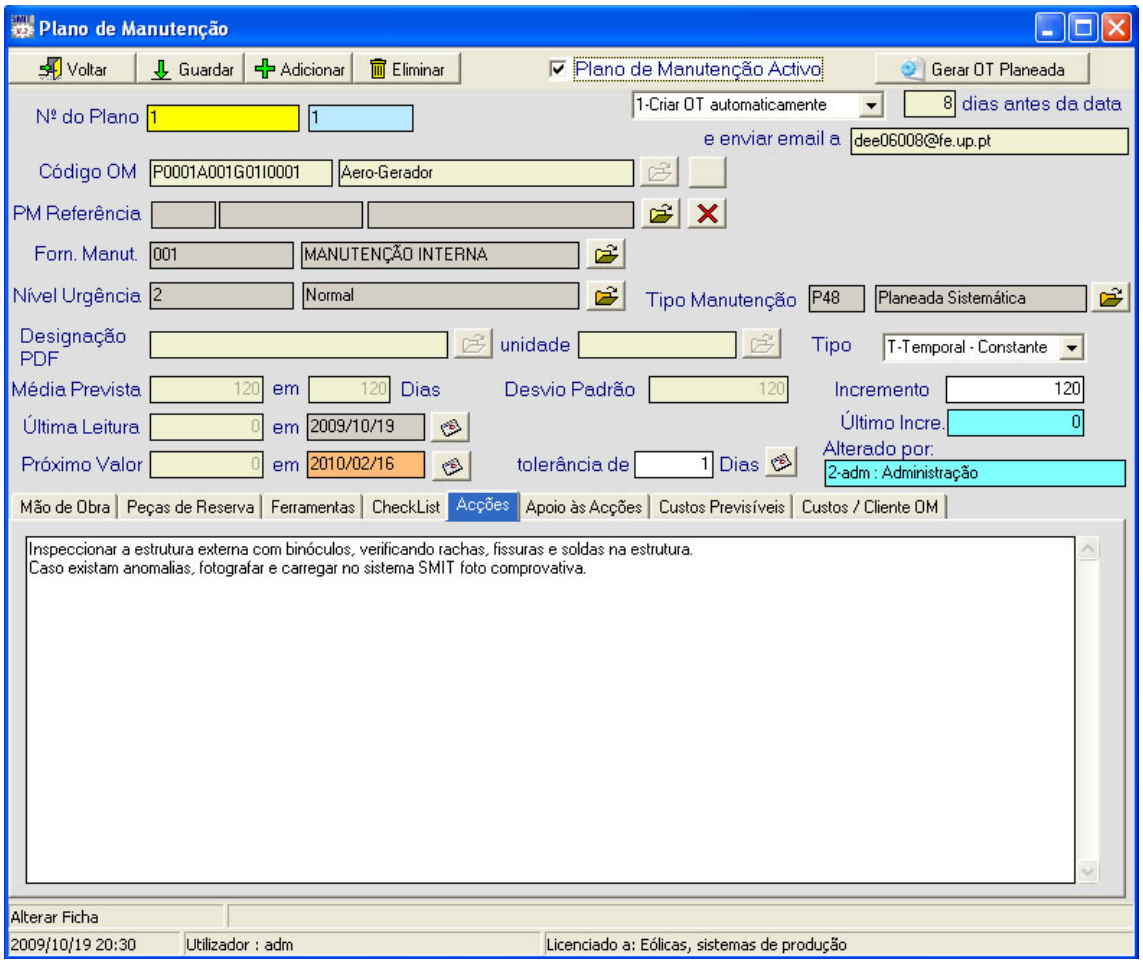



Figura A.8: Ficha de um plano de manutenção e escolha do algoritmo de planeamento.

Custos a vários Clientes de OM. A filosofia de Clientes de OM é a de indicar ao SMIT quais são as entidades que utilizam determinado objecto de manutenção e que, automaticamente, podem ser responsabilizadas pelas acções necessárias de manutenção. Na prática, os custos da manutenção são distribuídos aos clientes para quem o OM "trabalhou".

## A.5 Módulo de Ordens de Trabalho

O módulo de Ordens de Trabalho (OT) permite descrever os trabalhos a serem executados ou já realizados em determinado objecto de manutenção. Existem dois tipos de OTs:

- **Planeadas** - dividem-se em dois grupos:
  - **Planeadas condicionadas** - foram criadas pelo módulo de manutenção de condição (ver capítulo 5);
  - **Planeadas sistemáticas** - foram criadas a partir do módulo de Planos de Manutenção. Salienta-se que as OTs planeadas só podem ser criadas a partir desde módulo, estando




Figura A.9: Recursos do plano e respectivos custos.

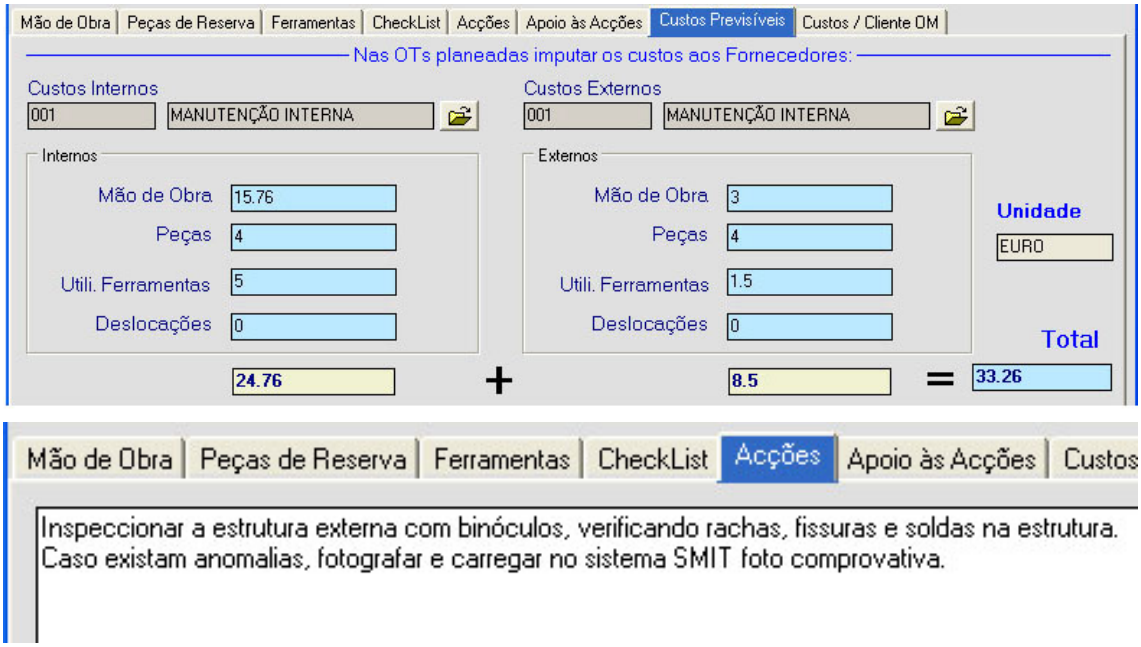



Figura A.10: Em cima: custos previsíveis. Em baixo: indicação das acções a efectuar.

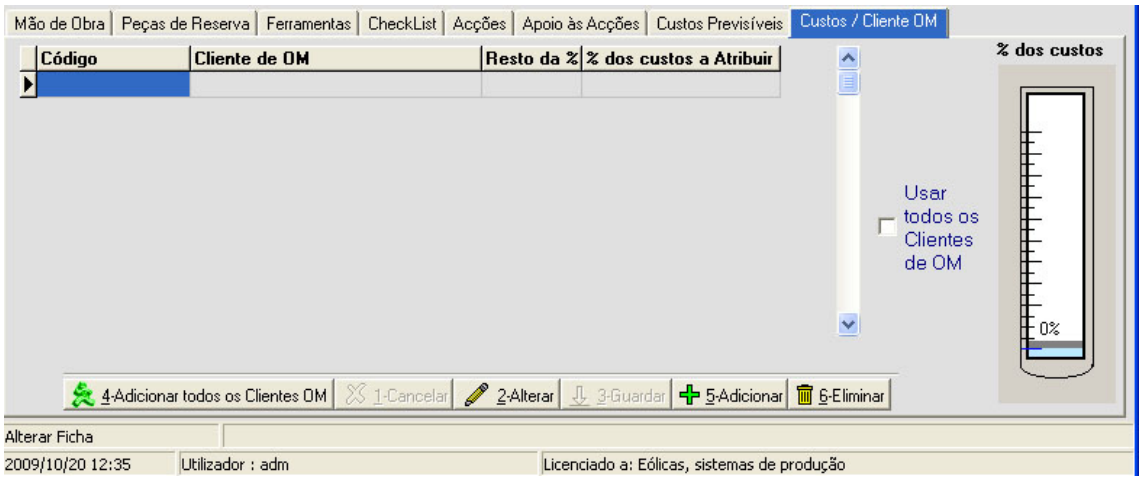



Figura A.11: Inserção dos Clientes de OM para imputação dos custos das intervenções.

vedado ao utilizador criar directamente OTs planeadas. Outro aspecto das OTs planeadas relaciona-se com a necessidade da recolha de um parâmetro - o valor do PDF e respectiva data de leitura na altura do encerramento da OT, na *tablete* "Plano", que é específico das OTs planeadas (Fig. A.13). Exclui-se desta situação o caso do plano que deu origem à OT depender de outro plano - através do campo "PM de referência" da ficha de PM. Nestas OTs não é possível alterar o campo de "Data de Ocorrência" (ver Fig. A.13);

- **Não planeadas** - são criadas pelo utilizador através do botão "Adicionar". Estas OTs quando são encerradas obrigam à recolha de uma palavra chave na *tablete* "Diagnóstico" que é específica das OTs não planeadas.

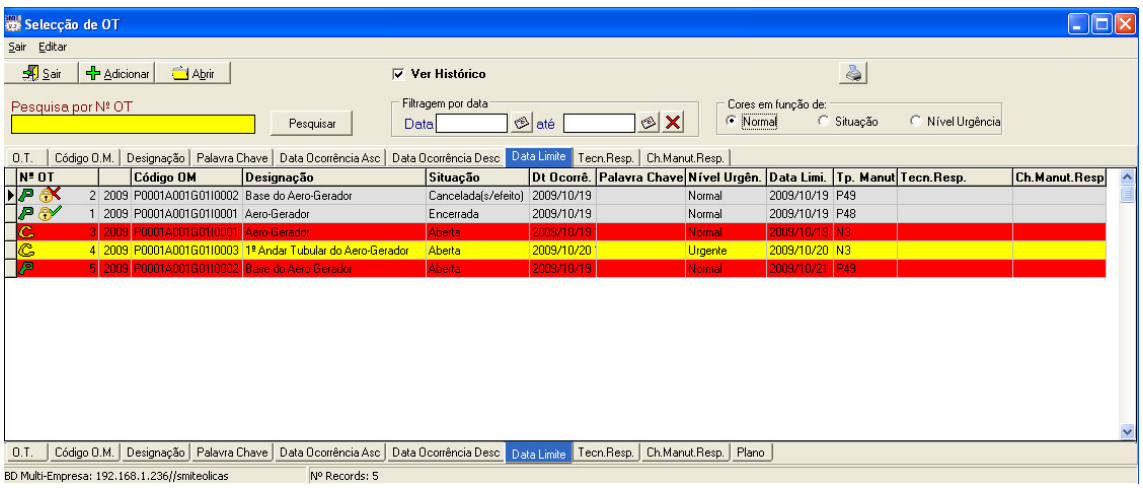



Figura A.12: Selecção de OT.

Na Fig. A.12 estão listadas algumas OTs, onde o C - indica OT correctiva ou não planeada e P - indica OT que teve origem num Plano de Manutenção. Há semelhança com o que acontece nos PMs; também aqui as cores têm um significado associado (opção cores em função de NORMAL):

- **Branco** - a data da OT é superior em uma semana à data actual;
- **Amarelo** - a OT deve ser realizada em menos de uma semana;
- **Vermelho** - a OT já devia ter sido realizada ou deve ser hoje realizada;
- **Cinzento** - OT encerradas. Caso tenha sido encerrada aparece o símbolo . O símbolo  indica que OT foi cancelada não tendo sido executada.

As cores, em função do nível de urgência ou da situação, são especificadas pelo utilizador do programa, já que este lhe permite associar uma cor a uma situação de urgência. Por defeito as OTs encerradas não aparecem; no entanto, o utilizador pode aceder-lhes através do botão "Ver histórico". Na Fig. A.13 estão indicados os campos necessários para o preenchimento de uma OT. Os dados na *tablete* "Paragem / Arranque" são obrigatórios para o encerramento das OTs. Os campos "Tipo de Manutenção" e "Data de Ocorrência" só podem ser alterados se a OT for "não planeada". A mão-de-obra da OT é aqui indicada. No caso de OT planeada este campo é automaticamente preenchido com os dados indicados no PM. Nesta situação, tal como ilustrado na Fig. A.13, o campo "Custo Hora" da mão-de-obra está a amarelo indicando que o custo da mão-de-obra é diferente do declarado na ficha de técnico, significando que foi modificado pelo utilizador.

Na Fig. A.13 pode-se verificar a indicação das peças-de-reserva e das ferramentas necessárias para executar a ordem de trabalho. Nas *tabletes* de mão-de-obra, peças-de-reserva e ferramentas, sempre que o custo de um elemento for diferente do que está descrito na sua ficha, será apresentado o símbolo  em conjunto com a cor amarela nesse campo, sinalizando assim esta situação.

Relativamente às peças-de-reserva existem três situações distintas, quanto às cores com que são representadas:

- **Cor de laranja** - indica que as peças ficaram cativas (peça-de-reserva não é externa);
- **Amarelo** - indica que a peça-de-reserva é externa. Os *stocks* não são alterados;
- **Verde** - a peça foi descontada do *stock*, e existia quando foi associada à OT.

O funcionamento das cativações, descontos de peças, etc, depende da OT ser planeada ou não. As discrepâncias existentes são as seguintes:

- **OTs planeadas** - No caso das peças serem externas nada acontece com a quantidade de *stock*. As peças-de-reserva nunca são descontadas de *stock*, só na altura em que a OT é encerrada. Na realidade, é efectuado um movimento de "Cativação de Peça para OT Planeada". Esta metodologia permite que as OT planeadas para cinco ou seis meses de distância não interfiram noutras manutenções ao descontarem as peças de *stock*;
- **OTs não planeadas** - No caso das peças-de-reserva serem externas, nada acontece com a quantidade em *stock*. Se as peças-de-reserva forem internas, a quantidade é sempre descontada ao *stock* existente, caso contrário, é efectuado um movimento de "Cativação de Peça para OT não Planeada". Nesta situação, o desconto será efectuado aquando do encerramento da OT caso já exista quantidade suficiente para satisfazer o pedido. Dado que uma OT não planeada está associada a situações de avarias, isto implica que sejam imediatamente descontadas as peças, uma vez que a OT está, regra geral, a decorrer no momento.

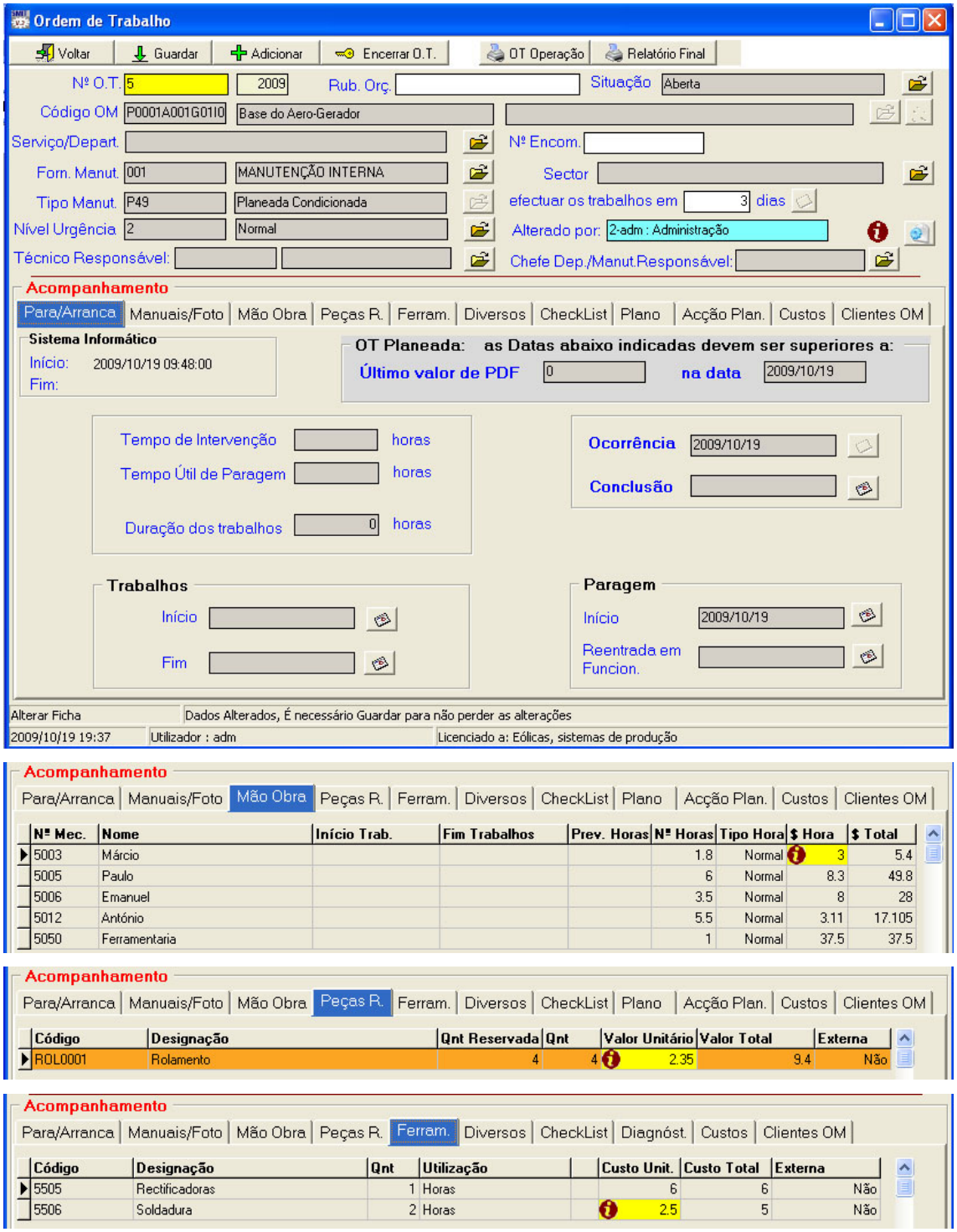



Figura A.13: Em cima: alguns campos da Ficha de OT. Ao meio: mão-de-obra da OT. Em Baixo: Peças-de-Reserva e ferramentas da OT.

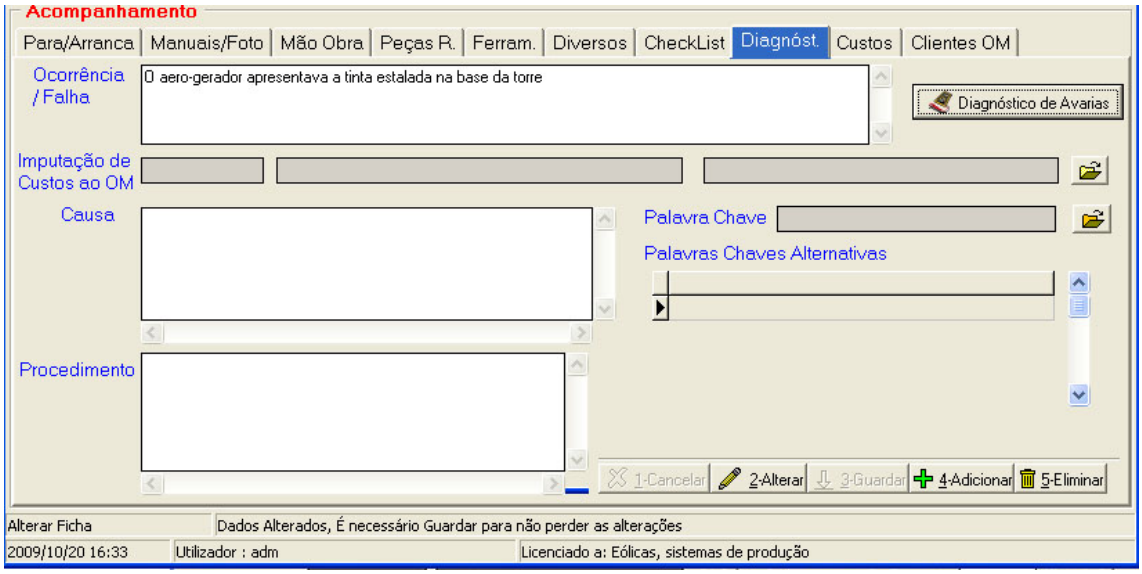



Figura A.14: Caracterização de uma ordem de trabalhos correctiva (não planeada), onde é necessário "catalogar" a avaria através da recolha das causas, do procedimenteo de reparação e de palavras chave, bem como da possibilidade de ocorrência.

Na Fig. A.14 mostra-se a introdução de dados para uma OT não planeada. Neste caso, um tubo de ensaio permite ao utilizador quantificar a sua impressão acerca da avaria (indica uma percentagem relativa à opinião do técnico). Normalmente essa impressão está associada à experiência que o utilizador tem. Com este procedimento simples, o programa recolhe esse conhecimento para mais tarde o utilizar noutro módulo do programa.

É também de enorme relevância a indicação de uma, ou mais, palavras-chave, dado que permitem associar à avaria termos simples de pesquisa, facilitando, desta forma, a pesquisa em situações posteriores. Regra geral, os custos associados a uma OT são contabilizados quer no fornecedor da manutenção quer no OM. Caso o responsável da manutenção pretenda, pode imputar os custos a outro OM através do campo "Imputação de custos ao OM" (*tablete* "Clientes de OM"), se for essa a situação real da intervenção. De facto, em OMs interligados, por vezes o utilizador reporta avarias a determinados equipamentos, quando, na realidade, o OM com falha é outro conectado àquele e, por consequência, é a este que devem ser imputados os dados da intervenção.

## A.6 O Servidor SMIT

Um dos pontos mais importantes do SMIT é a vertente do servidor, no qual é guardada toda a informação relevante.

O servidor SMIT utiliza o *Linux* como sistema operativo, *Slackware* versão 12.1 e 13.0 estando também disponível a sua instalação para o *FreeBSD* versão 6.3 e 7.0. Como é evidente, a sua portabilidade para qualquer outro sistema operativo é facilmente realizável, condicionada à existência de ferramentas de desenvolvimento tais como o compilador de *C*, *make*, *gmake* e outras

aplicações relacionadas. Igualmente, é possível a sua instalação em ambiente *Windows*, embora com maior dificuldade.

### A.6.1 Motor de base de dados

O motor de base de dados utilizado foi o *PostgreSQL* devido à facilidade disponibilizada do ponto de vista de linguagens de programação procedimentais, estando disponíveis, nativamente, as linguagens PL/pgSQL[1], PL/Tcl[2], PL/Perl[3] e PL/Python[4]. No entanto, podem ser instaladas opções de terceiros, como as linguagens PL/PHP, PL/R, etc.

Todas as funções foram programadas na linguagem PL/pgSQL, num total de cerca de 150; para além das funções, o sistema incorpora ainda cerca de 140 tabelas SQL. Na listagem A.1 apresenta-se o exemplo da função associada ao algoritmo de planeamento escrito na linguagem de programação PL/pgSQL.

Listagem A.1: Função para recalcular o planeamento

```
CREATE FUNCTION recalcula_planeamento(integer, text, real, date) RETURNS text
    AS '
declare
 av_cod alias for $1;       -- Codigo do PM
 av_codsmom alias for $2;   -- Codigo do OM
 av_ultleit alias for $3;   -- Última leitura
 av_dtultleit alias for $4;-- Data da Última leitura
 ------------------------------------------
 lv_valorpdf real;  -- valor PDF
 lv_dtpdf date;     -- Data de valor de PDF
 lv_savedata date;
 lv_integer integer;
 lv_a real;
 lv_b real;
 DP2_real_f_n0 real;
 DP2_real_f_n1 real;
 DP2_previ_f_n1 real;
 m_real_f_n0 real;
 m_real_f_n1 real;
 m_previ_f_n1 real;
 df_n1 real;
 dt_n1 real;
 dt_n2 real;
 lv_varProxVal real;
 lv_varProxData date;
 lv_smpm record;
 lv_close integer;
Begin
 select into lv_smpm * from smpm where cod = av_cod and codsmom = av_codsmom;
 if not found then
    RAISE EXCEPTION ''ERROR-PM NOT FOUND'';
 end if;
 lv_valorpdf := av_ultleit;
 lv_dtpdf := av_dtultleit;
 if lv_smpm.otjagerada = 1 then
    insert into smpmleituras values (av_cod,av_codsmom,lv_dtpdf,lv_valorpdf,now(),0,1,1);
    --RAISE EXCEPTION ''THERE IS AN OT FOR THIS PM'';
    return ''OK'';
 end if;
 if lv_smpm.codsmomref is null or lv_smpm.codsmomref = '''' then
    if av_dtultleit <= lv_smpm.dtultleit then
        RAISE EXCEPTION ''DATA DESTA LEITURA INFERIOR A DATA DA ULTIMA LEITURA DO PM'';
    end if;
```

[1] PL/pgSQL - *SQL Procedural Language* idêntica à linguagem do Oracle.
[2] PL/Tcl - *Tcl Procedural Language*
[3] PL/Perl - *Perl Procedural Language*
[4] PL/Python - *Python Procedural Language*

```
45      -- guarda a leitura para saber que foi ca colocada
        insert into smpmleituras values (av_cod,av_codsmom,lv_dtpdf,lv_valorpdf,now(),1,1,1);

        -- nao faz planeamento se a data tiver sido alterada pelo utilizador
        if (lv_smpm.tipo = ''T'') or (lv_smpm.tipo = ''A'') then
50          lv_integer := round(lv_smpm.medprevdias);
            df_n1 := 0;
            lv_valorpdf := 0;
            if lv_smpm.tipo = ''T'' then
                lv_varProxData := lv_dtpdf + lv_integer;
55          else
                lv_varProxData := lv_smpm.dtproxvalo;
                LOOP
                    lv_varProxData := lv_varProxData + lv_integer;
                    if (lv_varProxData > current_date) then
60                      exit;
                    end if;
                END LOOP;
                -- Verifica se o valor data difere do pretendido...
                lv_a := lv_varProxData - current_date;
65              if lv_a >= abs(1.7 * lv_smpm.medprevdias) then
                    lv_varProxData := lv_smpm.dtproxvalo;
                end if;
            end if;
        else
70          lv_a := 0.1;
            lv_b := 0.1;
            -- média prevista / nº de dias (resultado = Média/dia)
            if (lv_smpm.medprevdias<1.0) then
                lv_smpm.medprevdias := 1;
75          end if;
            m_real_f_n0 := lv_smpm.medprevvalo / lv_smpm.medprevdias;

            -- Delta de Funcao
            df_n1 := lv_valorpdf - lv_smpm.ultleit;
80
            -- Delta dos tempos
            dt_n1 := lv_dtpdf - lv_smpm.dtultleit;

            -- BEGIN INACIO COLOQUEI PONDERADO POR INTERVALO
85          lv_smpm.sdeltat := lv_smpm.sdeltat + dt_n1;
            lv_smpm.sdeltaf := lv_smpm.sdeltaf + df_n1;
            df_n1 := lv_smpm.sdeltaf;
            dt_n1 := lv_smpm.sdeltat;
            -- END INACIO
90
            -- media = derivada
            m_real_f_n1 := df_n1 / dt_n1; --media real
            -- media prevista pelo alisamento exponencial
            m_previ_f_n1 := (m_real_f_n1 + (((1 - lv_a) ^ dt_n1) * (m_real_f_n0-m_real_f_n1)));
95
            ------------------------------------------
            -- Modificado relativamente ao Metodo Normal
            ------------------------------------------
            lv_varProxVal := lv_smpm.proxvalo;
100         lv_close := 0;
            if lv_close = 1 then
                if lv_smpm.tipo = ''C'' then
                    lv_varProxVal := lv_valorpdf + lv_smpm.incremen;
                else
105                 lv_varProxVal := lv_smpm.proxvalo;
                    if (lv_smpm.incremen<0) then
                    LOOP
                        lv_varProxVal := lv_varProxVal + lv_smpm.incremen;
                        if (lv_varProxVal < lv_valorpdf) then
110                         exit;
                        end if;
                    END LOOP;
                    else
                        LOOP
115                         lv_varProxVal := lv_varProxVal + lv_smpm.incremen;
                            if (lv_varProxVal > lv_valorpdf) then
                                exit;
                            end if;
                        END LOOP;
120                 end if;
                    -- Verifica se o valor Funcao actual difere do pretendido...
```

```
                lv_a := abs(lv_varProxVal - lv_valorpdf);
                if lv_a >= abs(1.7 * lv_smpm.incremen) then
                    lv_varProxVal := lv_smpm.proxvalo;
                end if;
            end if;
        end if;

        lv_a := lv_varProxVal - lv_valorpdf; --Delta funcao para andar
        if (abs(m_previ_f_n1)<0.001) then -- caso especial de medias pequenas...
            -- temos lv_a >= 1   1 / qnt pequena > 366 dias
            dt_n2 := 366;
        else
            dt_n2 := lv_a / m_previ_f_n1;
            if (dt_n2>366.0) then -- nao deixa > 1 ano
                dt_n2 := 366;
            end if;
        end if;
                lv_integer := round(dt_n2);
                lv_varProxData := lv_dtpdf + lv_integer;

                -- Cálculo do valor do desvio padrao
                DP2_real_f_n0 := lv_smpm.desvpadr * lv_smpm.desvpadr / lv_smpm.medprevdias;
                lv_a := m_real_f_n1 - m_real_f_n0;
                DP2_real_f_n1 := dt_n1 * lv_a * lv_a;
                DP2_previ_f_n1 :=(DP2_real_f_n1 + (((1 - lv_b) ^ dt_n1) * (DP2_real_f_n0 - DP2_real_f_n1)));
                DP2_previ_f_n1 := |/ ( DP2_previ_f_n1*lv_smpm.medprevdias);
                lv_smpm.medprevvalo := m_real_f_n1 * lv_smpm.medprevdias;
                lv_smpm.proxvalo := lv_varProxVal;
                lv_smpm.desvpadr := DP2_previ_f_n1;
    end if;
    lv_savedata := lv_varProxData;
    if (lv_varProxData < current_date) then
            lv_varProxData := current_date + 1;
    end if;
    -- Seo utilizador tiver mudado a data <>dtprevOT, deixa ficar a dele...voila
    if lv_smpm.dtproxvalo <> lv_smpm.dtprevOT then
            lv_varProxData := lv_smpm.dtproxvalo;
    end if;
    update smpm set
        codsmotg = null, codsmotanog = null, codsmpedint = null, codsmpedintano = null,
        otjagerada = 0, dtultleit = lv_dtpdf, dtprevot = lv_savedata, dtproxvalo = lv_varProxData,
        sdeltaf = lv_smpm.sdeltaf, sdeltat = lv_smpm.sdeltat,
        ultincvalo = df_n1, ultleit = lv_valorpdf,
        medprevvalo = lv_smpm.medprevvalo, proxvalo = lv_smpm.proxvalo, desvpadr = lv_smpm.desvpadr
      where cod = av_cod and codsmom = av_codsmom;
 end if;
 return ''OK'';
End;
'
    LANGUAGE plpgsql;
```

Para acesso remoto dos utilizadores, optou-se por instalar um servidor de páginas, o *Apache* com a linguagem *PHP*, versão 2.2.4 e 5.2.1, respectivamente. Alguns componentes do SMIT são disponibilizados em *PHP*. A linguagem de programação *PHP* é ainda utilizada para executar periodicamente funções de gestão da base de dados, nomeadamente, inicialização da base de dados *PostgreSQL* e geração automática de emails, OTs, etc.

Para resguardar a propriedade intelectual presente em todo o desenvolvimento, nomeadamente nos *scripts PHP* e SQL do *PostgreSQL*, foram criadas duas ferramentas de encriptação.

Listagem A.2: Função encriptada, cujo código fonte se encontra na Listagem A.3

```
CREATE FUNCTION fnctrg1_smtipomanu() RETURNS "trigger"
    AS '
dFoElPaFiDcBpAjDoFiBfGlCbPcPbCpApAaGcFpFkOiAiDmAgCfGaGgPoOkDhFbEdBdGnDhDaFgGnEdGdEgCaCmBdB
lElAbDmElClBoBdCeBiAiBhEeCgFaGmAfClBlFgGkClFfDbCgCnBhEkPeAaDhElOjAaBjOiOkDkAhDeFnPmAgCpAjA
lFlBdAgCcAkEfFlEkEmEkBcPeGpFdAbApElBlElFnDpEmCbBhAbFjBgCjAkOhBmOkFiFbGgCaCjBdDoPgBlDmOcAaE
fFlFiDiEiDpBeBaFfBkBcCjOiBiOeFcFbGbDgAfFlDlEgDcGnFoFoDlAfBeAdBoOmEhEfGkAfFjAgEmAfGdAbAnFoA
eGkFdEhBlOcCiBdGlEjAiBnCkEnPaFeFiBiBhEcBdCfAkPjEjPaAfPcBbFnDnFaAeGhFmBiBaPkAlOdGhChEeBcBfA
oEkFcDpFcPbBjCkCpBpEkEcDgAaBgFpBiDfAfDbCkCaFhCbBkBiFlEkAkCoOiAcPoBbGmAhBbDbGmCaGgDgGgGaDbD
iEkBcPeGgEgCnFdDeDbGhFdDpFnDhAiElOcDnDaCfBmCdDoBkEhD
```

```
10
'
    LANGUAGE plpgsql;
```

Relativamente aos *scripts* de SQL, apenas as funções PL/pgSQL são encriptadas; relativamente ao *PHP* todo o conteúdo do programa é encriptado. Obviamente, para realizar esta tarefa houve a necessidade de alterar o código fonte das respectivas distribuições. A ferramenta de encriptação dos ficheiros *PHP* é simples de implementar dado que todo o ficheiro deve ser encriptado. A ferramenta para efectuar esta operação nos *scripts* do *PostgreSQL* é mais complexa, pela necessidade de interpretar o código fonte tal como se de um compilador se tratasse. Para isso foi criado um ficheiro de regras para o interpretador *flex* e *bison*, que são duas das ferramentas incontornáveis na criação de novas linguagens.

O resultado pode ser verificado através da listagem A.2 cujo código fonte se encontra na listagem A.3

**Listagem A.3: Função de *trigger* associada à criação de códigos**

```
CREATE FUNCTION fnctrg1_smtipomanu() RETURNS "trigger"
    AS '
declare
  lv_next integer;
5 Begin
  lv_next := nextval(''S1_smtipomanu'');
  if ( NEW.planeada <> ''1'' and NEW.planeada <> ''0'' ) then
    NEW.planeada := ''0'';
  end if;
10       if ( NEW.planeada = ''1'' ) then
    NEW.cod := ''P'' || lv_next;
  else
    NEW.cod := ''N'' || lv_next;
  end if;
15  return NEW;
End;
'
    LANGUAGE plpgsql;
```

O *flex* é uma ferramenta que permite especificar sequências de caracteres como símbolos (*tokens*). Quando combinado com o *bison* é possível criar tradutores de linguagem de forma rápida e eficiente. A ferramenta *bison* permite indicar regras através da sequência de símbolos (indicados na listagem em *flex*) produzindo saídas de texto (normalmente linguagem de programação C), para gerar a tradução desejada. O exemplo descrito na listagem A.2 foi obtido através de um programa gerado com estas duas ferramentas. As listagens A.4 e A.5 apresentam algumas das principais regras utilizadas quer no *flex* quer no *bison*. Notar, que, neste caso em concreto, é preciso criar duas ferramentas, uma para codificar (encriptar) e outra para descodificar o SQL.

**Listagem A.4: Símbolos descritos no *Flex***

```
%option    yylineno
%option    yy_scan_string
%{
#include <stdio.h>
5 #include <string.h>
#include "pgs.h"
extern int gv_crypt;
extern int func00(unsigned char *data,unsigned int ind[], int check,int translate);
extern int func01(unsigned char *data,unsigned int ind[],unsigned char *out);
10
int gv_debug=0;
int  gv_strlen=0;
```

```
#define TAM_BUF 4000
char gv_str[TAM_BUF+10], gv_str_out[TAM_BUF+10],gv_str_out_crypt[3*TAM_BUF+10];
FILE *fd_out=0;
FILE *fd_out_crypt=0;
char gv_temp_file_name[200];
static int mv_flag=0;
static int mv_last_print_out=0;
void faz_CRYPT(int doit,int gv_test){
...
}
void save_out(){
...
}

void debug_flex(char *msg) {
...
}
// REJECT
// abaixo estão definidos os TOKENS a utilizar como detecção para
// encriptar/desencriptar as funções PLPGSQL
%}
digit [0-9]
word     [a-zA-Z][a-zA-Z0-9_]+
plg [pP][lL][pP][gG][sS][qQ][lL][_][cC][aA][lL][lL][_][hH][aA][nN][dD][lL][eE][rR]
CRLF   [\r][\n]
%%
[Aa][Ll][Tt][Ee][Rr]                    {debug_flex("ALTER"); return ALTER;}
[Tt][Aa][Bb][Ll][Ee]                    {debug_flex("TABLE"); return TABLE;}
[cC][rR][eE][aA][tT][eE]                {debug_flex("CRE"); return CREATE;}
[oO][rR]                                {debug_flex("OR"); return OR;}
[rR][eE][pP][lL][aA][cC][eE]            {debug_flex("REP"); return REPLACE;}
[fF][uU][nN][cC][tT][iI][oO][nN]        {debug_flex("FUNC"); return FUNCTION;}
\(                                      {debug_flex("OFUNC"); return OPENFUNC;}
\)                                      {debug_flex("CFUNC"); return CLOSEFUNC;}
[rR][eE][tT][uU][rR][nN][sS]            {debug_flex("RET"); return RETURNS;}
[aA][sS]                                {debug_flex("AS"); return AS;}
["][tT][rR][iI][gG][gG][eE][rR]["]      {debug_flex("TRG"); return TRIGGER;}
{plg}                                   {debug_flex("LANG"); return LANGPL;}
[Ss][Ee][Tt][Oo][Ff]                    {debug_flex("SETOF"); return SETOF;}
,                                       {debug_flex("VIR"); return VIRGULA;}
''                                      {debug_flex("DPLI"); return DUPLAPLICA;}
'                                       {debug_flex("PLI"); return PLICA;}
;                                       {debug_flex("PVIR"); return PONTOVIRGULA;}
[lL][aA][nN][gG][uU][aA][gG][eE]        {debug_flex("LAN"); return LANGUAGE;}
[pP][lL][pP][gG][sS][qQ][lL]            {debug_flex("PL"); return PLPGSQL;}
{word}*                                 {debug_flex("WRD"); return WORD;}
[ \t]+                                  {debug_flex("ESP"); return ESP;}
{CRLF}                                  {debug_flex("NL"); return ENTER;}
[\n]                                    {debug_flex("NL"); return ENTER;}
.                                       {debug_flex("TCH"); return TCHAR;}
%%
```

Listagem A.5: Regras codificadas em *Bison*

```
%{
#include <stdio.h>
#include <string.h>
#include <string.h>
#include <unistd.h>

extern void save_out(int i);
extern void print_out(int );
extern int faz_CRYPT(int doit,int gv_test);
extern char gv_str[];
extern char *yytext;
extern int yystate;
extern FILE *fd_out;
extern FILE *fd_out_crypt;
extern char gv_temp_file_name[];
extern int gv_debug;

//-------------------
extern char p2[];

int gv_crypt=1;
int gv_flextoken,gv_flextoken;
```

```
int gv_test=0;

yyerror(const char *str) { if (gv_debug!=0) fprintf(stderr,"yyerror: %s\n",str); return 0; }
int yywrap() { return 1; }

int main(int argc,char *argv[])
{
  extern FILE *yyin;
  char *ptr_str,*pass,*ptr_name=0;
  int i,my_type_lib;

  gv_crypt=-1;
  for (i=1;i<argc;i++) {
    if (strcmp(argv[i],"-c")==0) gv_crypt=1;
    else
    if (strcmp(argv[i],"-d")==0) gv_crypt=0;
    else
    if (strcmp(argv[i],"-t")==0) gv_test=1;
    else
    ptr_name=argv[i];
  }
  switch(my_type_lib=type_lib()){
  case 0: printf("Programa sem utilidade - Veja params de Compilacao!!!\n");return 0;
  case 1: gv_test=0; gv_crypt=0; break; // com password, sem func crypt
  case 2: break;
  case 3: gv_test=0; gv_crypt=1; break;
  }
  if ((gv_crypt==-1) || (ptr_name==0)) {printf("Usage: %s <-c/-d> file\n",argv[0]); return 0;}
  strcpy(gv_temp_file_name,ptr_name);

  if (gv_crypt==1) strcat(gv_temp_file_name,".plgz"); else {
    ptr_str=strstr(gv_temp_file_name,".plgz");
    if (ptr_str!=0) ptr_str[0]=0; else strcat(gv_temp_file_name,".out");
  }
  fd_out=fopen(gv_temp_file_name,"w");
  yyin=fopen(ptr_name,"r");
  gv_str[0]=gv_temp_file_name[0]=0;
  strncat(gv_temp_file_name,tmpnam(argv[0]),200);
  //gv_debug=1;
  //gv_test=1;
  if ((gv_test==0) && (my_type_lib==2)) {
     if (gv_crypt==0)
               pass=getpass("Password for DeCrypt: ");
          else
               pass=getpass("Password for Crypt: ");
          strcpy(p2,pass);
          pass=getpass("Retype Password : ");
          if (strcmp(p2,pass)!=0) {
               printf("Passwords are not equal!!!\n");
               exit(0);
          }
  }
  printf("Parsing File\n\n");
        yyparse(); // Faz o parsing do Ficheiro ....
  fclose(fd_out);
  if (fd_out_crypt!=0) fclose(fd_out_crypt);
  fclose(yyin);
  unlink(gv_temp_file_name);
//-------------------------------------------------------------------
}

%}

%token CREATE OR REPLACE FUNCTION ALTER TABLE
%token OPENFUNC VIRGULA CLOSEFUNC
%token RETURNS AS PLICA DUPLAPLICA
%token LANGUAGE PLPGSQL PONTOVIRGULA
%token WORD ESP TCHAR ENTER TRIGGER LANGPL SETOF
%error-verbose
%%
// REGRA INICIAL
input:  /* EMPTY */
  | input commands
  | error ENTER {if (gv_debug&1!=0)
                      printf("\n\n------>ERR INPUT  \n");
                      print_out(9901); yyerrok; yyclearin;
                    }
```

```
100   ;

      commands:
        FUNCTION_PL
        | TABLE_SQL
105     | TCHAR
        | WORD          {if (gv_debug&1!=0) printf("WORD: %s\n",yytext);}
        | OPENFUNC
        | ENTER         { if (gv_debug&1!=0) printf("ENTER\n"); print_out(0);}
        | DUPLAPLICA
110     | PLICA
        | CREATE
        | OR
        | REPLACE
        | FUNCTION
115     | CLOSEFUNC
        | VIRGULA
        | RETURNS
        | LANGUAGE
        | LANGPL
120     | PLPGSQL
        | AS
        | TRIGGER
        | PONTOVIRGULA
        | ESP
125     | error ENTER { if (gv_debug&1!=0) printf("\n\n----------->ERROR-COMMANDS (%s)\n\n",yytext);
                         print_out(9902); yyerrok; yyclearin;
                       }
        ;

130   TABLE_SQL: CREATE ESP TABLE
        | ALTER ESP TABLE
        ;

      FUNCTION_PL:
135     FUNCTION_START BODY_FUNC FUNCTION_END_PL                                                {faz_CRYPT(1,gv_test);}
        | FUNCTION_START BODY_FUNC FUNCTION_END_OTHER                                         {faz_CRYPT(0,gv_test);}
        | CREATE ESP_NL FUNCTION ESP_NL LANGPL OPENFUNC CLOSEFUNC ESP_NL RETURNS ESP_NL WORD  {print_out(8802); }
        | CREATE ESP_NL FUNCTION ESP_NL WORD OPENFUNC FUNC_ARGS_BAD                           {print_out(8801); }
        ;
140
      FUNCTION_START:
        CREATE ESP_NL OR ESP_NL REPLACE ESP_NL FUNCTION ESP_NL WORD OPENFUNC FUNC_ARGS        {print_out(2);}
        | REPLACE ESP_NL OR ESP_NL CREATE ESP_NL FUNCTION ESP_NL WORD OPENFUNC FUNC_ARGS      {print_out(2);}
        | CREATE ESP_NL FUNCTION ESP_NL WORD OPENFUNC FUNC_ARGS                               {print_out(2);}
145     | REPLACE ESP_NL FUNCTION ESP_NL WORD OPENFUNC FUNC_ARGS                              {print_out(2);}
        ;

      FUNC_ARGS_BAD:
          ARGS CLOSEFUNC ESP_NL RETURNS ESP_NL WORD ESP_NL AS ESP_NL PLICA TCHAR            {} // PARA ANULAR FUNCS
150
      FUNC_ARGS:
          ARGS CLOSEFUNC ESP_NL RETURNS ESP_NL WORD ESP_NL AS ESP_NL PLICA ESP_NL           {print_out(2);}
        | ARGS CLOSEFUNC ESP_NL RETURNS ESP_NL TRIGGER ESP_NL AS ESP_NL PLICA ESP_NL       {print_out(2);}
        | ARGS CLOSEFUNC ESP_NL RETURNS ESP_NL SETOF ESP_NL WORD ESP_NL AS ESP_NL PLICA ESP_NL {print_out(2);}
155     ;

      ARGS: /* EMPTY */
        | ESP WORD
        | WORD
160     | ARGS ESP WORD
        | ARGS ESP_NL VIRGULA ESP_NL WORD
        | ARGS VIRGULA WORD
        | ARGS ESP_NL VIRGULA WORD
        | ARGS VIRGULA ESP_NL WORD
165     ;

      FUNCTION_END_PL:
        PLICA ESP_NL LANGUAGE ESP_NL PLPGSQL PONTOVIRGULA                                     {print_out(4);}
        | PLICA ESP_NL LANGUAGE ESP_NL PLPGSQL ESP_NL PONTOVIRGULA                            {print_out(4);}
170     | ESP_NL PLICA ESP_NL LANGUAGE ESP_NL PLPGSQL PONTOVIRGULA                            {print_out(4);}
        | ESP_NL PLICA ESP_NL LANGUAGE ESP_NL PLPGSQL ESP_NL PONTOVIRGULA                     {print_out(4);}
        ;

      FUNCTION_END_OTHER:
175     PLICA ESP_NL LANGUAGE ESP_NL ANULA_FEND PONTOVIRGULA                                  {print_out(4);}
        | ESP_NL PLICA ESP_NL LANGUAGE ESP_NL ANULA_FEND PONTOVIRGULA                         {print_out(4);}
```

```
    ;
ANULA_FEND:
  WORD
  | WORD ESP
  | ANULA_FEND WORD
  | ANULA_FEND WORD ESP
  ;

BODY_FUNC:    /* EMPTY */
  | BODY_FUNC BODY_CMDS
  ;

BODY_CMDS: /* EMPTY */
   TCHAR
  | ENTER                   { print_out(3);}
  | WORD
  | DUPLAPLICA
  | CREATE
  | OR
  | REPLACE
  | FUNCTION
  | OPENFUNC
  | CLOSEFUNC
  | VIRGULA
  | RETURNS
  | TRIGGER
  | AS
  | PONTOVIRGULA
  | ESP
  | error ENTER { if (gv_debug&1!=0) printf("\n\n----------->ERROR-BODY_CMDS (%s)\n\n",yytext);
      print_out(9902);yyerrok;yyclearin;
          }
  ;

ESP_NL:
  ENTER                     { print_out(1);}
  |ESP |
  | ESP_NL ENTER            { print_out(1);}
  | ESP_NL ESP
  ;
%%
```

### A.6.2 Ferramentas para cálculo científico

Para execução dos algoritmos matemáticos foram integrados os pacotes de *software* do *Octave* versão 3.0.3 e *R* versão 2.8.0 (ou 2.9.0). A sua inclusão no SMIT, para o caso do *R*, foi simples dado esta ferramenta incluir, nativamente, suporte para a base de dados *PostgreSQL* através de dois pacotes opcionais - *TSPostgreSQL* ou *RPostgreSQL*. Quanto ao *Octave*, houve necessidade de criar uma livraria para o efeito, com suporte SSL (Listagem A.7). A listagem do programa A.6 apresenta a codificação em *Octave* do algoritmo C.2 (pág. 212).

Listagem A.6: Algoritmo em *Octave*

```
%-------------------------------------------------------------------------%
% Alisamento exponencial
% entradas: cod_can - código da placa que recolhe os dados
%           mac     - código da Gateway
%           tempo   - timestamp da data hora até ns da aquisição
%           valor   - valor medido no terreno
% Utiliza: pg_isOTopen - função escrita em C++ que verifica se existe OT
%                        para o Equipamento em Questão
%          pg_insertOT - função escrita em C++ que insere uma OT no SMIT
%                        sempre que o valor estimado sai fora dos limites
%  function saida=TeseExpsmoo(cod_can, mac, tempo, valor_in)
%-------------------------------------------------------------------------
function saida=Expsmoo(cod_can, mac, tempo, valor)
lv_high_limit=50;
lv_low_limit=-10;
```

```
%pg_insere('smiteolicas',cod_can,mac,tempo,valor,1);
%dados=pg_get('smiteolicas',cod_can,mac,tempo,valor,1);
%dados=pgdata('smiteolicas','smit','smit',
%       'select extract(epoch from dtleitura),valor from smDAQ order by dtleitura desc limit 10;');

nfile=strcat(cod_can,mac,".mat");
try
    load(nfile);
catch
    alpha=0.4;          lv_locked=0;
    vect_in=[valor];    vect_out=[valor];
    save("-text",nfile,"alpha","vect_in","vect_out","lv_locked");
    saida=valor;
    return;
end
if lv_locked=1
    lv_otexiste=pg_isOTopen('smiteolicas',cod_can, mac);
    if lv_otexiste=1
        saida=valor;
        return;
    endif
    alpha=0.4;          lv_locked=0;
    vect_in=[valor];    vect_out=[valor];
    save("-text",nfile,"alpha","vect_in","vect_out","lv_locked");
    saida=valor;
    return;
endif

[sxin,syin]=size(valor);
if(sxin<1)
    exit;
endif;
vect_in=[vect_in valor];
[sxin,syin]=size(vect_in);
for m=2:1:syin
    valor=alpha*vect_in(m)+(1-alpha)*vect_out(m-1);
    vect_out=[vect_out valor];
end
saida=valor;
vect_out=[valor];
if (lv_locked=0)
    if (saida>lv_high_limit)
        lv_locked=pg_insertOT('smiteolicas',cod_can, mac);
    elseif (saida<lv_low_limit)
        lv_locked=pg_insertOT('smiteolicas',cod_can, mac);
    endif
endif
save("-text",nfile,"alpha","vect_in","vect_out","lv_locked");
```

As funções criadas para o *Octave* escritas em C++ foram as seguintes (A listagem A.7 apresenta a função pgdata):

- **pg_isOTopen(basededados,cod_redeCan,cod_MAC)** - verifica se na base de dados existe uma OT de condição de funcionamento para o Objecto de Manutenção associado ao par (*Gateway*,Placa *CAN* de E/S). Retorna 1 se existir 0 caso contrário;
- **pg_insertOT(basededados,cod_redeCan,cod_MAC)** - insere na base de dados uma OT de condição de funcionamento para o OM associado ao par (*Gateway*,Placa *CAN* de E/S). Retorna 1 se inseriu;
- **pgdata(basededados,utilizador,password,SQL)** - executa um comando SQL genérico;
- **pg_insere(basededados, cod_redeCan, cod_MAC,tempo,valor,n)** - insere dados nas tabelas internas de aquisição do SMIT;
- **pg_get(basededados, cod_redeCan, cod_MAC,tempo,valor,n)** - lê dados das tabelas internas de aquisição do SMIT.

Para o caso do *R*, não foram desenvolvidas rotinas específicas, estando apenas disponível o *interface* nativo. Este facto tem como consequência a necessidade do programa em *R* necessitar de executar os comandos SQL, enquanto que, no *Octave*, este aspecto foi escondido, facilitando assim a programação a um utilizador desta linguagem.

Listagem A.7: Livraria para acesso do *Octave* ao *PostgreSQL* escrita em C++, que permite ler dados da base de dados do SMIT para uma variável vector do *Octave*.

```
#include <config.h>             // Configuração do Octave
#include <oct-obj.h>
#include <defun-dld.h>

#include <stdio.h>
#include <stdlib.h>
#include <libpq-fe.h>
//-------------------------------------------------------------------
#define MSG  "D = pgdata( db, user, passwd, select_command)\n\n \
       create function date2num(timestamp) returns double precision \n \
       as ' \n \
         select date_part(''EPOCH'',$1); \n \
       ' language sql; \n\n \
       create function date2num(date) returns double precision \n \
       as ' \n \
         select date_part(''EPOCH'',$1); \n \
       ' language sql; \n\n \
        Example: \n \
     D = pgdata(\"smit\",\"cristal\",\"passwd\", [\"set DateStyle to 'ISO';\", \\\n \
                          \"select date2num(dtleitura), valor from smpmleituras where codsmom='F - 0980'\", \\\n \
                          \"order by dtleitura\"]);\n\n \
\n"
//-------------------------------------------------------------------

DEFUN_DLD (pgdata, args, nargout,MSG)
{
        const char *conninfo,*ptr;
        char str[1500],db[100],user[100],pass[100];
        PGconn     *conn;
        PGresult   *res;
        int                     nFields;
        int                     i,
                                j;

  octave_value_list retval;     // lista dos valores de retorno

  int nargs = args.length ();   // nº de argumentos

  if (nargs != 4) {             // sem argumentos obrigatórios mostra esta msg
    usage("Try 'help pgsql' for info");
    return retval;
  }
  str[1499]=0;
  ptr=args(0).string_value().c_str();
  snprintf(db,100,"dbname=%s ",ptr);
  ptr=args(1).string_value().c_str();
  snprintf(user,100,"user=%s ",ptr);
  ptr=args(2).string_value().c_str();
  snprintf(pass,100,"password=%s ",ptr);
  snprintf(str, 1499, "%s%s%s sslmode=require ",&db[0],&user[0],&pass[0]);
  //snprintf(str, 199, "dbname = %s user = pgsql ",ptr);
  conn = PQconnectdb(str);
  if (PQstatus(conn) != CONNECTION_OK)
  {
    error("Connection to database failed.\n");
    return retval;              // ...
  }
  ptr=args(3).string_value().c_str();
  snprintf(str,1499,"%s",ptr);
  str[args(3).length()]=0;
  res = PQexec(conn, str);
  if (!res || PQresultStatus(res) != PGRES_TUPLES_OK) {
               error("Query failed");
               error(ptr);
               PQclear(res);
```

```
                PQfinish(conn);
                return retval;
  }
  long k = PQnfields(res); // resultado do Postgresql
  long n = PQntuples(res);

  if (n == 0) { // Existem Dados?
    //error("Query returned no data");
    //PQclear(res);
    //PQfinish(conn);
    //return retval;
    Matrix Y(0,0);
    retval(0) = Y;
    PQclear(res);
    PQfinish(conn);
    return retval;              // nada a retornar
  }
  Matrix X(n, k);               // Cria a matriz de retorno

      for (long i=0; i<n; i++)  // converte para double
       for (long j=0; j<k; j++)
          sscanf(PQgetvalue(res, i, j), "%lf", &X(i, j));

  retval(0) = X;                // set the return vector and
    PQclear(res);
    PQfinish(conn);
  return retval;                // give it back to the caller
}
//
// compile-command: "mkoctfile pgsignal.cc -I/usr/include/libpq -I/usr/include/postgresql -L/usr/local/pgsql/lib -lpq"
// install-command: "cp *.oct /usr/libexec/octave/site/oct/i486-slackware-linux-gnu" and
//                  "ldconfig /usr/local/pgsql/lib /lib /usr/lib"
```

## A.7 Instalação do *Software* SMIT

Nesta secção apresentam-se os passos necessários para uma adequada instalação do SMIT.

### A.7.1 Servidor SMIT para *FreeBSD*

O *cd-rom* de instalação do servidor do SMIT possuí suporte para instalar nas versões do *FreeBSD* 6.3 e 7.0. O SMIT pode, no entanto, ser instalado em versões superiores, condicionado ao facto do utilizador resolver os eventuais erros que possam surgir. Para proceder à instalação do SMIT no *FreeBSD*, tendo em atenção as configurações das partições de disco de rede e outras, deve-se iniciar a instalação do sistema operativo escolhendo apenas os seguintes pacotes:

- *base* - ficheiros binários do sistema;
- *kernel* - sistema operativo;
- *man* - suporte para manuais do sistema;
- *ports* - instalar caso deseje usar as opções *make freebsd_build* e *make freebsd_build_crypt*.

Salienta-se que a partição de *swap* deverá ser criada com o dobro da memória disponível no computador. Finda esta instalação básica do *FreeBSD*, insere-se o *cd-rom* de instalação do SMIT e, mudando a directoria corrente para a base do *cd-rom*, na linha de comandos do sistema digita-se:

- *make freebsd_install_crypt* - instalação do servidor SMIT em modo encriptado. Após a conclusão, o sistema está pronto a entrar em funcionamento, sendo necessário efectuar uma inicialização com o comando *reboot*;

- *make freebsd_install* - instalação do servidor SMIT em modo normal. Após a conclusão, o sistema está pronto a entrar em funcionamento, sendo necessário efectuar uma inicialização com o comando *reboot*;
- *make freebsd_build_crypt* - permite criar o pacote de instalação do SMIT para o *FreeBSD* em modo encriptado (Nota: necessita da instalação dos *ports* e este passo não se encontra disponível no CD);
- *make freebsd_build* - permite criar o pacote de instalação do SMIT para o *FreeBSD* instalado (Nota: necessita da instalação dos *ports* e este passo não se encontra disponível no CD).

### A.7.2 Servidor SMIT para *Slackware*

O *cd-rom* de instalação do servidor do SMIT possuí suporte para instalar as versões do *Linux Slackware* 12.1 e 13.0. As opções são exactamente iguais ao do *FreeBSD* alterando o nome do sistema operativo; por exemplo, *make freebsd_build* passa a ser *make slack_build* e, de igual forma, para os outros comandos.

Para instalar um servidor SMIT neste sistema operativo deve-se, de início, instalar o *Linux Slackware*, incorporando uma partição de *swap* (tamanho 2 X memória) e outra de sistema de ficheiros. No momento da selecção das séries de pacotes, podem ser desligadas o *KDE, XWindows, XAP-Xapplication* e o *Y-Games*. De seguida, deve-se escolher a opção *menu* para seleccionar os pacotes dentro de cada série, sendo individualmente desligados os seguintes:

- *Na opção A* - jfsutils, loadlin, mkinitrd, xfsprogs;
- *Na opção AP:Applications* - desligar o alsa-utils, amp, aumix, cdparanoia, enscript, espgs, flac, hpijs, jed, joe, jove, madplay, mpg321, mysql, rexima, sox, vorbis-tools, workbone;
- *Na opção E:GnuEmacs* - desligar todas as opções;
- *Na opção F:Faqs* - desligar todas as opções;
- *Na opção K:Linux Kernel* - desligar todas as opções;
- *Na opção L:Libraries* - desligar o alsa-lib, apr, apr-util, audiofile, desktop-file-utils, esound, libmikmod, libogg, libtheora, libvorbis, mpeg_lib, néon, sdl, taglib, wv2;
- *Na opção N: Network* - desligar o http, irssi, php, wireless-tools.

No final da configuração da rede e de reiniciado o sistema operativo, coloca-se o *cd-rom* e utiliza-se o *make* para colocar o SMIT em funcionamento, tal como descrito para o *FreeBSD*.

### A.7.3 SMIT Cliente para *Windows*

O pacote SMIT para o ambiente *Windows* constituí a parte de Cliente do *software* apresentado nesta dissertação. O método de instalação é bastante simples, estando dividido em três fases:

- **Instalação inicial** - este passo acontece quando o utilizador instala o programa pela primeira vez devendo utilizar o ficheiro *smit_total.exe*;
- **Actualizações** - o SMIT incorpora um esquema de actualizações remotas automáticas do *software* instalado em *Windows*; para tal é disponibilizado um ficheiro executável no servidor SMIT em *Unix*, onde são alojadas as actualizações. Sempre que um utilizador acede ao

programa, aquele verifica a existência de uma nova versão e, caso isto se verifique, o SMIT descarrega o executável e inicia a sua instalação automaticamente.

Para que o esquema indicado anteriormente funcione de forma adequada é necessário que o utilizador tenha as devidas permissões. No momento da primeira instalação do SMIT, assume-se tal procedimento ser realizado por um administrador. No caso das actualizações remotas não é admissível "sobrecarregar" o administrador com tal tarefa, pelo que foi criada uma alternativa, explicada de seguida através das duas hipóteses:

- ***Instalação com a conta de administração*** - nesta opção, o trabalho é simples e resume-se a executar o ficheiro de instalação fornecido;
- ***Instalação sem conta de administração*** - de modo a instalar o programa, nomeadamente as actualizações, sempre que o administrador da rede não se encontra disponível; optou-se por utilizar um programa *open-source* - CPAU - através do qual o Administrador insere as permissões criando um ficheiro de configuração, que deve ser instalado no servidor do SMIT. Sempre que existe uma nova actualização o cliente SMIT do *Windows* descarrega a nova actualização e o respectivo ficheiro CPAU para executar a instalação com as permissões de Administrador. A vantagem deste processo é permitir esconder a senha do administrador e o programa em que é executado.

### A.7.4 Construção dos ficheiros de instalação para *Windows*

Os pacotes de instalação do SMIT foram construídos com base no *InstallShield* do *Delphi* e no *nLite Add-On Maker* e dividem-se em três parcelas:

- **cristall_install.msi** - instalação da parte de *runtime* do *Cristall Report*, para execução dos relatórios disponibilizados por esta plataforma. O ficheiro de instalação foi construído com base no *InstallShield*;
- **psqlodbc.msi** - instalação *odbc* da base de dados do *PostgreSQL*, disponibilizada pelo fabricante;
- **smit.msi** - instalação do *software* SMIT. O ficheiro de instalação foi construído com base no *InstallShield*.

Após a obtenção destes três ficheiros, utilizou-se a versão *open-source* do *nLite Add-On Maker*. Este *software* é utilizado para construir *add-ons* para inserir nos *CD-Rom* de instalação do *Windows*, mas adequa-se perfeitamente à realização da presente tarefa, isto é, "juntar" estes três ficheiros de instalação num só que realize todas as operações de instalação do *software* SMIT.

Para construir este pacote é necessário incluir um *script* no executável final (é um ficheiro zip auto-extraível), de acordo com a Listagem A.8.

Listagem A.8: Ficheiro instalar.cmd

```
@echo off
cls
REM echo This file will install SMIT software in your computer.
REM echo.
REM echo Press Ctrl-C to abort the upgrade or pause
```

```
msiexec /i psqlodbc.msi /passive
msiexec /i cristall\_install.msi /passive
msiexec /i smit.msi /passive

cd \%TEMP\%
echo Windows Registry Editor Version 5.00> smit.reg
echo.>> smit.reg
echo [HKEY\_LOCAL\_MACHINE\textbackslash{}SOFTWARE\textbackslash{}
      ODBC\textbackslash{}ODBC.INI\textbackslash{}ODBC Data Sources]>> smit.reg
echo "smitv2"="PostgreSQL ANSI">> smit.reg
echo.>> smit.reg

echo [HKEY\_LOCAL\_MACHINE\textbackslash{}SOFTWARE\textbackslash{}
      ODBC\textbackslash{}ODBC.INI\textbackslash{}smitv2]>> smit.reg
echo "Driver"="\%PROGRAMFILES\%\textbackslash\textbackslash{}
       psqlODBC\textbackslash\textbackslash{}0803\textbackslash\textbackslash{}
       bin\textbackslash\textbackslash{}psqlodbc30a.dll">> smit.reg
echo "CommLog"="0">> smit.reg
echo "Debug"="0">> smit.reg
echo "Fetch"="100">> smit.reg
echo "Optimizer"="0">> smit.reg
echo "Ksqo"="1">> smit.reg
echo "UniqueIndex"="1">> smit.reg
echo "UseDeclareFetch"="0">> smit.reg
echo "UnknownSizes"="0">> smit.reg
echo "TextAsLongVarchar"="1">> smit.reg
echo "UnknownsAsLongVarchar"="0">> smit.reg
echo "BoolsAsChar"="1">> smit.reg
echo "Parse"="0">> smit.reg
echo "CancelAsFreeStmt"="0">> smit.reg
echo "MaxVarcharSize"="255">> smit.reg
echo "MaxLongVarcharSize"="8190">> smit.reg
echo "ExtraSysTablePrefixes"="dd\_;">> smit.reg
echo "Description"="">> smit.reg
echo "Database"="smit">> smit.reg
echo "Servername"="127.0.0.1">> smit.reg
echo "Port"="5432">> smit.reg
echo "Username"="cristal">> smit.reg
echo "UID"="cristal">> smit.reg
echo "Password"="cristal">> smit.reg
echo "ReadOnly"="0">> smit.reg
echo "ShowOidColumn"="0">> smit.reg
echo "FakeOidIndex"="0">> smit.reg
echo "RowVersioning"="0">> smit.reg
echo "ShowSystemTables"="0">> smit.reg
echo "Protocol"="7.4">> smit.reg
echo "ConnSettings"="">> smit.reg
echo "DisallowPremature"="0">> smit.reg
echo "UpdatableCursors"="1">> smit.reg
echo "LFConversion"="1">> smit.reg
echo "TrueIsMinus1"="0">> smit.reg
echo "BI"="0">> smit.reg
echo "AB"="0">> smit.reg
echo "ByteaAsLongVarBinary"="0">> smit.reg
echo "UseServerSidePrepare"="0">> smit.reg
echo "LowerCaseIdentifier"="0">> smit.reg
echo "SSLmode"="require">> smit.reg

echo "XaOpt"="1">> smit.reg

\%windir\%\textbackslash{}REGEDIT /S smit.reg

del smit.reg
```

O ficheiro de comandos da Listagem A.8 executa as instalações do SMIT, do ODBC e do *crystal report*. De seguida efectua o registo da ligação ODBC, nomeadamente no "DSN de Sistema" para o funcionamento dos relatórios.

Caso fosse necessário também se poderia criar um ficheiro para desinstalar o SMIT com o nome "uninstall.cmd", de acordo com a Listagem A.9.

Listagem A.9: Ficheiro uninstall.cmd

```
@echo off
cls
REM echo This file will uninstall SMIT software from your computer.
REM echo.
REM echo Press Ctrl-C to abort the upgrade or pause

cd \%TEMP\%
echo Windows Registry Editor Version 5.00> smit.reg
echo.>> smit.reg
echo [-HKEY\_LOCAL\_MACHINE\textbackslash{}SOFTWARE\textbackslash{}
      ODBC\textbackslash{}ODBC.INI\textbackslash{}ODBC Data Sources]>> smit.reg
echo.>> smit.reg
echo [-HKEY\_LOCAL\_MACHINE\textbackslash{}SOFTWARE\textbackslash{}
      ODBC\textbackslash{}ODBC.INI\textbackslash{}smitv2]>> smit.reg

msiexec /x psqlodbc.msi /passive
msiexec /x cristall\_install.msi /passive
msiexec /x smit.msi /passive

\%windir\%\textbackslash{}REGEDIT /S smit.reg

del smit.reg
```

**Construção do Ficheiros cristall_msi**

Para construir o ficheiro msi de instalação do *Cristall Report* foi necessário instalar o *Visual Studio versão 6.0* e o *Visual Studio Installer* e, de seguida, efectuar as instruções do documento *rdc9_deployment.pdf*[5].

## A.8 Conclusões

Este anexo apresentou abreviadamente o desenvolvimento do *software* de manutenção SMIT, nomeadamente a gestão de utilizadores, de objectos de manutenção, de planos de manutenção, e de ordens de trabalho.

As considerações ligadas a facilitar a instalação foram igualmente expostas com ênfase para a instalação do servidor e dos clientes *Windows*, aproveitando-se para mostrar aspectos tais como a interligação com os *softwares Octave*, *R* e a encriptação das funções mais sensíveis a nível da base de dados.

[5]Disponível em: http://resources.businessobjects.com/support/communityCS/TechnicalPapers/rdc9_deployment.pdf

# Anexo B

# Descrição do *software*/*hardware* do sistema de Aquisição de Dados

Neste anexo apresenta-se o desenvolvimento do sistema de aquisição de dados descrito parcialmente no capítulo 5. É igualmente exposta uma breve introdução à rede *CAN* e, de seguida, apresentados os principais detalhes. Os sistemas descritos são os módulos PIC e ARM.

No final do anexo apresentam-se algumas fotos do *hardware* desenvolvido bem como os seus diagramas esquemáticos.

## B.1 Sistemas de aquisição de dados de baixo custo

Nesta secção é descrito o *hardware* desenvolvido para aquisição de dados. A solução do "problema" da aquisição de dados deve ser aqui vista como um contributo para a criação de um sistema que solucione de forma adequada o fim em causa ao mais baixo custo, tendo em consideração o actual desenvolvimento na área dos micro-controladores. Nesta perspectiva, deverá ter-se sempre presente a enorme evolução e a relação de preço/poder de cálculo que, actualmente, apresentam vários micro-controladores de 8, 16 ou 32 *bits*. Associado ao seu poder de cálculo deve-se ainda levar em conta a variedade de periféricos de comunicação, tais como a *Ethernet*, o *CAN*, *RS485*, *RS422*, *RS232*, *ProfiBus*, etc.

Com este pressuposto e com a intenção de colocar a hipótese dos sistemas desenvolvidos poderem comunicar com os apresentados nas secções anteriores, optou-se pela escolha dos micro-controladores com periféricos *CAN* e *Ethernet*. A razão da utilização do *CAN* é sobejamente justificada dado ser um dos meios físicos de comunicação usados na indústria automóvel e na automação industrial, mais exactamente nos protocolos *DeviceNet* e *CanOpen*. Estes dois protocolos não são proprietários, sendo denominados protocolos industriais abertos e, por consequência, permitem ao sistema desenvolvido comunicar com variadíssimos equipamentos industriais. De notar que quer o *DeviceNet*, quer o *CanOpen* são protocolos lógicos implementados no meio físico *CAN*, mas também poderiam ser implementados sobre outro meio físico, tal como a *Ethernet*.

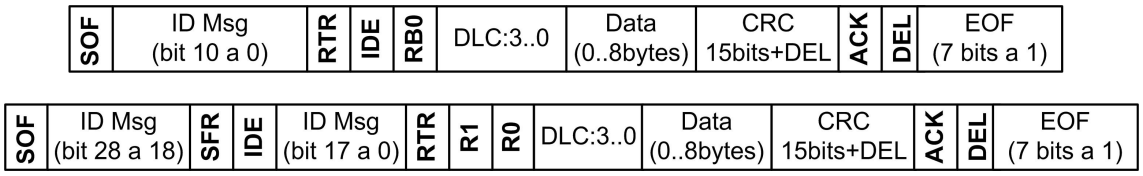



Figura B.1: Formatos da mensagem de dados. Em cima: *Standard*. Em baixo: *Extended*.

### B.1.1 O *CAN*

O *CAN* foi desenvolvido pela *Bosch* nos meados dos anos oitenta, sendo actualmente válidas as revisões 2.0A e 2.0B normalizadas pela norma ISO/DIS 11898.

É um protocolo estruturado em camadas, seguindo o modelo OSI normalizado pela norma ISO 7498 de sete camadas, mas, apenas duas são especificadas pela norma ISO/DIS 11898, em concreto, a camada de ligação de dados e a camada física.

Ao contrário de outros, este protocolo está orientado à mensagem e não ao nó. Ou seja, não são os nós a possuir identificadores mas sim as mensagens. Assim, cada nó pode receber todo o tipo de mensagens desde que convenientemente programado. Os identificadores que caracterizam a mensagem definem a prioridade desta.

As mensagens são enviadas com um formato fixo, para quatro tipos diferentes de mensagens: dados; remota; erro e sobrecarga. A norma 2.0A discrimina o formato *standard* e a 2.0B o formato *Extended* da mensagem (ver Fig. B.1 para o formato da mensagem de dados; os outros não serão detalhados).

Os tipos de mensagens têm o seguinte significado:

- **Dados** - mensagem de dados podendo conter desde zero até oito *bytes*;
- **Remota** - este tipo de mensagens serve para um nó pedir um tipo de mensagens a um segundo nó da rede;
- **Erro** - serve para indicar que uma mensagem enviada para a rede chegou com erros a pelo menos um nó;
- **Sobrecarga** - serve para um nó anunciar que não se encontra preparado para continuar a receber determinada mensagem.

A decisão de qual o nó a fazer a transmissão usa o método CSMA/CD, ou seja, todos tentam transmitir, mas aquele que tiver menor prioridade deixa de transmitir ao detectar que um dos *bits* da sua mensagem difere do *bit* existente na rede. Este processo é feito automaticamente pelo *hardware* de controlo ao nível físico, através de uma técnica de *bits* dominantes (zero) e recessivos (um).

Normalmente, todos os controladores de *CAN* oferecem um método de selecção das mensagens com base no identificador de mensagem (ID Msg da Fig. B.1). Esta tarefa é elaborada através da máscara e do filtro. A máscara (*bit*=0 - indiferente; *bit*=1 - verifica) serve para indicar os *bits* da mensagem recebida que devem coincidir com os *bits* na mesma posição do valor do filtro programado. Uma máscara com todos os *bits* a 0 permite a recepção de todas as mensagens, enquanto




Figura B.2: Indicação ao nível físico do local onde o controlador *CAN* deve amostrar um *bit* numa mensagem.

que uma máscara com todos os *bits* a um apenas recebe uma mensagem cujo identificador seja igual ao valor programado no filtro.

Não há transmissão de relógio nas mensagens *CAN* e, portanto, existe a necessidade de os receptores sincronizarem a mensagem. A velocidade de transmissão define o tempo de cada *bit*, estando este dividido em vários segmentos, tal como apresentado na Fig. B.1. O tempo de *bit* é múltiplo de uma quantidade básica definida - *TQ-Time Quantum*- cujo valor depende do micro-controlador e da frequência do relógio e de registos programados de acordo com uma fórmula específica[1]. A partir do TQ são definidos os parâmetros da Fig. B.2. O local onde cada *bit* é amostrado pode ser controlado através da indicação dos quatro parâmetros seguintes:

- Segmento de sincronização - normalmente dura um TQ e serve para sincronização;
- Segmento de propagação - pode durar entre 1TQ a 8TQ;
- Segmento *Phase Buffer1* - pode durar entre 1TQ e 8TQ;
- Segmento *Phase Buffer2* - ocupa o restante tempo do *bit* em TQs.

O ponto onde o controlador efectua a amostragem é determinante, sendo recomendando na literatura um valor sensivelmente de 75% do tempo de *bit*, em função do comprimento do cabo *CAN* e da velocidade de transmissão[2].

Estes parâmetros são altamente delicados caso seja necessário colocar vários equipamentos com diferentes processadores e frequências de relógio a comunicarem entre si na mesma rede *CAN*. É necessário planear correctamente cada um dos valores aquando da inicialização do controlador *CAN*, caso contrário os equipamentos não conseguem comunicar entre si.

### B.1.2 PIC18F2685 e DSPIC13F4012

Este micro-controlador possuí nativamente o controlador de *CAN*, suportando a norma 2.0A e a norma 2.0B. Face ao seu baixo custo e bom desempenho, tornou-se numa das opções consideradas. A frequência máxima admissível é de 40Mhz e consegue efectuar aquisições a 100 kHz partilhado por cinco entradas analógicas (AN0,...,AN4). Para aceder à rede *CAN* é necessário um *transceiver* que, na prática, converte os níveis de TTL para os níveis da rede *CAN*; neste caso foi utilizado o PCA82C251. A Fig. B.6 (pág. 189) mostra o esquema eléctrico da placa baseada neste controlador.

[1] Consultar o *datasheet* da PIC18F2685, dsPIC13F4012, LM3S8962
[2] Consultar http://www.can-cia.org

A Fig. B.7 (pág. 190) apresenta o mesmo conceito descrito agora para a dsPIC18F4012, cuja diferença, relativamente à PIC18F2685, está na maior velocidade, em ter um processador de 16 *bits* e em poder trabalhar, no máximo, a 120Mhz. Possuí ainda a capacidade de efectuar aquisição simultânea através de quatro entradas analógicas, a uma velocidade máxima de 1Msps.

### B.1.3 ENC28J60 e PIC18F67J60

A Fig. B.8 (pág. 191) apresenta-nos o esquema eléctrico do controlador de rede ENC28J60. Trabalha à velocidade de 10Mbps em *half-duplex* e pode ser interligado a um micro-controlador através de SPI, passando este, desde que devidamente programado, a conseguir aceder a uma rede *Ethernet*. No caso de ser interligado à dsPIC30F4012 ou à PIC18F2685, ficam ambas com duas conectividades, *Ethernet* e *CAN*. Relativamente ao ENC28J60, a maior dificuldade durante os trabalhos ocorreu na sua programação, que depende das revisões de *hardware* e, nalguns casos, torna-se necessário corrigir por *software* falhas ao nível do *hardware*. Entretanto, o fabricante lançou muito recentemente uma nova versão deste circuito integrado para uma velocidade de 100Mbps.

A Fig. B.9 (pág. 192) apresenta uma segunda solução com o micro-controlador mais compacto PIC18F67J60, mas este apenas possuí conectividade *Ethernet*. Seria necessário incluir um controlador de *CAN* externo para o efeito.

O fabricante disponibiliza, em *open-source* para ambiente *Windows*, as ferramentas de desenvolvimento, embora exista uma versão baseada no GCC ainda experimental[3].

### B.1.4 LM3S8962

O LM3S8962 é um micro-controlador mais caro (10 €) que o anterior, com a vantagem de ter todas as funcionalidades necessárias ao projecto, incluindo o registo temporal à chegada de pacotes de *Ethernet*. Aliás, este é um dos aspectos mais enfatizados na publicidade feita pelo seu fabricante e é efectiva, atendendo a que o tempo necessário para colocar em funcionamento esta metodologia diminui drasticamente comparativamente com a solução do ENC28J60, nomeadamente ao nível do *driver* da rede (comunicação entre o micro-controlador e o controlador de *Ethernet*). Outra grande vantagem relaciona-se com as ferramentas de programação que podem ser construídas a partir de *software open-source*, nomeadamente GNU GCC compilado para ARM Cortex-M3[4], Binutils 2.19.1[5] e newlib-1.17.0[6]. A Fig. B.10 (pág. 193) apresenta o diagrama esquemático da placa baseada neste micro-controlador.

[3] SDCC: http://sdcc.sourceforge.net
[4] Actualmente gcc-4.4.3, consultar http://gcc.gnu.org
[5] *Assembler* que gera o código baixo nível, consultar http://sourceware.org/binutils
[6] Versão compatível com a *libc* para sistemas embebidos, consultar http://sourceware.org/newlib

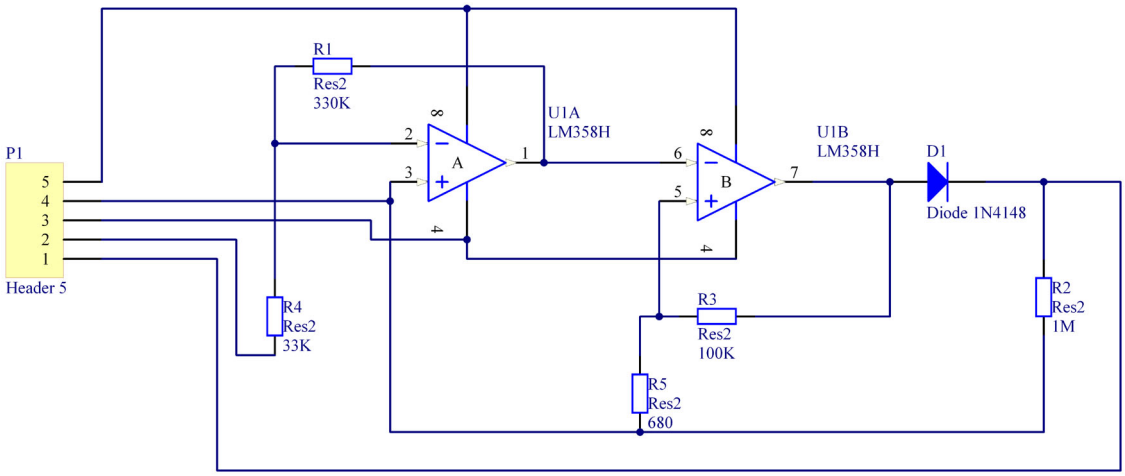



Figura B.3: Esquema para interligação com anemómetro analógico de copos.

### B.1.5 Interligação entre o sistema de aquisição de dados e o anemómetro analógico de copos

Existem pelo menos dois tipos de anemómetros de copos, um com saída digital e outro com saída analógica. Os anemómetros de saída digital funcionam, internamente, com um interruptor que, a cada volta, fornece um impulso. Os interruptores são, normalmente, implementados com relés *reed* (um íman e um sensor) ou com fotodíodos.

Relativamente aos anemómetros analógicos, que correspondem ao modelo utilizado neste exemplo, normalmente incorporam uma bobina, onde passam os ímanes em movimento, sendo equivalentes a um gerador AC.

Para efectuar a medição do vento existe a necessidade de converter este sinal em rotações por unidade de tempo e efectuar a respectiva calibração em túnel de vento.

Das duas possibilidades existentes para efectuar esta medida tem-se:

1) Converter a sinusóide para um sinal DC para obter uma tensão proporcional à média do módulo da onda sinusoidal;

2) Contar os picos positivos ou negativos.

Optou-se pela segunda hipótese, que se encontra ilustrada na Fig. 5.16 (pág. 85) através de uma onda quadrada (Canal 2, 5v/DIV), tendo sido utilizado o circuito da Fig. B.3 para converter o sinal do anemómetro para 5v TTL. No anemómetro utilizado, a tabela do fabricante para calibração em túnel de vento aponta para uma relação de:

$$Y(km/h) = 0.091046 * X(rpm) \tag{B.1}$$

O *software* desenvolvido conta o número de voltas do anemómetro para um $\Delta t = 60s$ levando em consideração que são dados dois pulsos por volta.

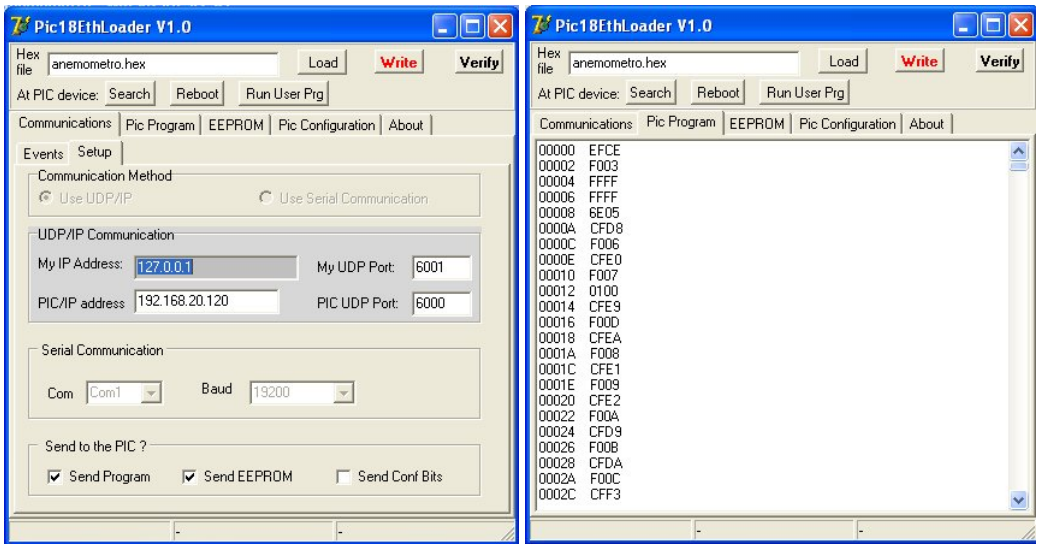



Figura B.4: Programa em *Windows* para "carregar" o *firmware*.

### B.1.6 *Bootloader* para as *gateway Ethernet-CAN*

Numa implementação real pode haver necessidade de efectuar alterações ao nível do *firmware*, sendo bastante incómoda a deslocação para realizar esta tarefa. Assim, foi desenvolvido um *bootloader* que permite alterar o programa (apenas foi efectuado para a PIC18F2685). Para o efeito foram desenvolvidos três programas, dois em *Windows* (são recompiláveis com *Linux* para colocar no servidor *Linux SMIT Server*) e um para a PIC. Quanto aos programas do *Windows*, um simula a PIC com o *bootloader* e o outro lê e envia o programa à PIC. Com o simulador tornou-se mais fácil corrigir os erros encontrados ao longo do desenvolvimento de melhoria do *firmware*.

A necessidade deste desenvolvimento deve-se ao facto de permitir "depositar" no servidor *Linux* SMIT os novos programas para cada placa; o servidor envia-os aos respectivos destinatários e coloca os programas em funcionamento. Desta forma é possível, remotamente, efectuar a manutenção do sistema, desde a recolha de dados até à actualização do *software* em qualquer elemento constituinte do sistema.

O programa que é executado no *Windows* é apresentado na Fig. B.4, tendo sido implementadas as seguintes funcionalidades:

- Ler o ficheiro em formato *Intel* (HEX - só foi suportado o essencial);
- Permitir configurar micro-controladores (PICs) em termos de memória e tamanho de bloco para escrita do programa e zona de configuração;
- Conseguir identificar qual a PIC que se está a programar;
- Reiniciar o micro-controlador remotamente;
- Colocar em execução o programa que foi carregado;
- Escrever na memória da PIC o novo programa e verificá-lo.

A Programação é feita através do protocolo IP/UDP, estando representado à esquerda na Fig.

B.4, onde se indica o IP da placa onde se pretende "carregar" o novo programa. À direita encontra-se o resultado da leitura do ficheiro "anemómetro.hex" e a zona da memória onde este deve ser carregado. Assim, existem dois programas essenciais ao processo de *bootloader*: 1º - programa do *Windows* que foi apresentado atrás (lê o novo ficheiro e encarrega-se de o transmitir); 2º - programa especial que está no micro-controlador (recebe a comunicação e grava o novo programa para este entrar em funcionamento). As mensagens trocadas entre os dois equipamentos obedecem a cinco tipos diferentes:

- **PRG_PIC** - pacote para programar ***size*** *bytes*;
- **GET_PRG_PIC** - leitura de uma zona de memória;
- **GET_PIC_ID** - perguntar qual o ID do micro-controlador;
- **RBT_PIC** - reiniciar o micro-controlador;
- **RUN_USER_PRG** - colocar em execução o programa previamente "carregado".

De início é necessário gravar o programa do *bootloder* que se posiciona no final da memória para deixar livre o espaço inicial para a aplicação. O algoritmo B.1 corresponde à estratégia de implementação do *bootloader*.

**Algoritmo B.1** *Algoritmo do bootloader da Gateway*

***Repete***
*Espera comunicações*
***Se*** *recebeu acção a executar* ***então***
***Caso*** *a acção a fazer* ***seja****:*
***Programar****: escrever na memória o bloco*
***Ler****: ler o bloco de memória*
***Reiniciar****: efectuar reset ao microcontrolador*
***Executar prg****: saltar para o programa do utilizador*
***Fim caso***
***Fim se***
***Se*** *durante x segundos não receber comunicações* ***então***
***Se*** *já há um programa carregado* ***então***
*saltar para o programa do utilizador*
***Fim se***
***Fim se***
***Até sempre***

### B.1.7 Listagem Parcial do protocolo SMIT-*Gateway*

A listagem B.1 apresenta parte do protocolo entre a *Gateway Ethernet* e o servidor SMIT, tal como já documentado no fluxograma da Fig. 5.10 (pág. 78).

Listagem B.1: Listagem Parcial da *gateway* LM3S8962

```
#include "smit_defs.h"
#include "inc/hw_ints.h"
#include "inc/hw_memmap.h"
```

```
//   Includes de CAN
#include "inc/hw_sysctl.h"
//
#include "inc/hw_nvic.h"
#include "inc/hw_types.h"
//   Includes de CAN
//
#include "driverlib/can.h"
#include "driverlib/ethernet.h"
#include "driverlib/flash.h"
#include "driverlib/gpio.h"
#include "driverlib/interrupt.h"
#include "driverlib/sysctl.h"
#include "driverlib/systick.h"
#include "driverlib/timer.h"
#include "utils/lwiplib.h"
#include "utils/uartstdio.h"
#include "utils/ustdlib.h"
#include "drivers/rit128x96x4.h"
#include "random.h"
#include "can_net.h"
#include "lifo.h"
#include "protocol_smit.h"
#include "timings.h"

extern volatile tLIFO gv_lifo_can;
extern volatile unsigned long g_ulSystemTimeSeconds;
extern volatile unsigned long gv_serverip, gv_serverip2senddata;
extern unsigned char pucMACArray[];
extern tTIMINGS_DAQ ga_sample_timings;
//-----------------------------------------
// Funciona como o DHCP, vai buscar o IP de um
// servidor à REDE : IP do SMIT SERVER...
// IP: 9945
//-----------------------------------------
int gv_ethernet_packets=0;

void recv_udp_packets(void *arg, struct udp_pcb *pcb, struct pbuf *p, struct ip_addr *addr, u16_t port)
{
    UDP_LINK *link;
    unsigned char * ptrdados;
    struct ip_addr myaddr;
    unsigned long *plong;
    unsigned int sizepacket;
    unsigned long long ttr;
    int i, flag;

    link=(UDP_LINK *)arg; // Recebe o link de onde está  a transmitir o pacote...
     if (p != NULL) {
        //udp_sendto(pcb, p, addr, port);
        if (p->len > 5) {
            gv_ethernet_packets++;
            ptrdados=(unsigned char*)p->payload;
            // recebimento de Perguntas directas, sem pedirmos confirmação
            flag=1;
            switch (ptrdados[0]) {
                case 253: // Forward para as PICS... Ainda para fazer e pensar bem como...
                          // implementar pedidos do SMIT directos às placas de CAN,... Fica
                          // para uma próxima oportunidade... não é relevante de momento.
                        break;
                case 254: // Envia a Versão do Software
                        memcpy ((unsigned char *)(link->ptr2send->payload)+1, VERSAO_SOFTWARE, sizeof(VERSAO_SOFTWARE));
                        link->ptr2send->tot_len = link->ptr2send->len = sizeof(VERSAO_SOFTWARE)+1 + link->size_header;
                        ((unsigned char*)link->ptr2send->payload)[0]=254;
                        udp_send(link->link, link->ptr2send);
                        link->ptr2send->tot_len = link->save2pckt_tot_len;
                        break;
                case 255: // receber um comando directo
                        myaddr = *addr; // Se quiser responder a quem nos contactou
                        udp_sendto(pcb, p, &myaddr, port); // RESPONDE COM ECHO PARA A REDE SABER QUE EXISTIMOS...
                        break;
                default: flag=0;
            }
            // RESPOSTA A PEDIDOS JA FEITOS PELO NOSSO PROGRAMA-----------------
            if (flag==0)
            if (ptrdados[0]==link->id_pacote_enviado) {
                link->id_ack_pacote_enviado = link->id_pacote_enviado;
```

```
                switch (ptrdados[0]) {
                case 0: // Receber o IP do Servidor
                        gv_serverip = ptrdados[4];
                        for (i=3;i>0;i--) {
                            gv_serverip <<=8;
                            gv_serverip+=ptrdados[i];
                        }
                        gv_serverip2senddata= ptrdados[8];
                        for (i=3;i>0;i--) {
                            gv_serverip2senddata <<=8;
                            gv_serverip2senddata+=ptrdados[i+4];
                        }
                        plong = (unsigned long*)(&ptrdados[9]);// Actualizamos o relógio caso o PTPD ainda não
                        if (plong[0]>g_ulSystemTimeSeconds) { // o tenha feito ... na SysTICK
                                //    sys_prot_t sProt;
                                //sProt = sys_arch_protect();
                                g_ulSystemTimeSeconds = plong[0];
                                // nanoseconds =0
                                //sys_arch_unprotect(sProt);
                        }
                        break;
                case 250: // Receber as tabelas de aquisição...+ Parametros do CAN...
                        if (p->len < 6+24) break;
                        sizepacket = 6;
                        ga_sample_timings.can_msgid=ptrdados[1];
                        ga_sample_timings.can_tck.uSyncPropPhase1Seg=ptrdados[2];
                        ga_sample_timings.can_tck.uPhase2Seg=ptrdados[3];
                        ga_sample_timings.can_tck.uSJW=ptrdados[4];
                        ga_sample_timings.can_tck.uQuantumPrescaler=ptrdados[5];
                        i=0;
                        ptrdados+=sizepacket;
                        // aceita apenas N_ELEMENTOS_TIMING temporizações (timings.h neste caso 20)
                        while ((sizepacket+24 <= p->len) && (i<=N_ELEMENTOS_TIMING)) {
                                ga_sample_timings.timings[i].linha=ptrdados[0];

                                ga_sample_timings.timings[i].epoch_sec = *((unsigned long*)(ptrdados+1));
                                ga_sample_timings.timings[i].epoch_nanosec = *((unsigned long*)(ptrdados+5));

                                ga_sample_timings.timings[i].epoch_sec_stop = *((unsigned long*)(ptrdados+9));
                                // qual o nº da mensagem de CAN a enviar para o Bus
                                ga_sample_timings.timings[i].can_msgid=
                                        ptrdados[13] | (ptrdados[14]<<8) | (ptrdados[15]<<16);
                                //
                                ga_sample_timings.timings[i].incr_seconds = *((unsigned long*)(ptrdados+16));
                                ga_sample_timings.timings[i].incr_nanoseconds = *((unsigned long*)(ptrdados+20));
                                // Calcula o Incremento em Segundos, quanto é ...derivado dos nanoseconds

                                ttr=(unsigned long long)ga_sample_timings.timings[i].incr_seconds;
                                ttr*=1000000000;
                                ttr+=(unsigned long long)ga_sample_timings.timings[i].incr_nanoseconds;
                                ga_sample_timings.timings[i].incr_tns=ttr;

                                ga_sample_timings.timings[i].esta_em_realtime=0;

                                ga_sample_timings.tempo_ordenado[i]=i;
                                ptrdados+=24;
                                sizepacket += 24;
                                i++;
                        }
                        ga_sample_timings.linhas = i;
                        ga_sample_timings.indice_time = 0;
                        // guarda a referência base do tempo ...
                        ga_sample_timings.current_time[0] = g_ulSystemTimeSeconds;
                        break;
                }
            }
        }
        pbuf_free(p);
    }
}
//-------------------------------------------------------------------
unsigned char ASK_SIP[]="ASERVERIP=..#";
//-------------------------------------------------------------------
unsigned long get_ip_server(UDP_LINK *link) {
    struct ip_addr ip_addr;
    int i;
    // Maquina de estados...    para trabalhar com a internet, protocolo UDP.
```

```
    switch (link->state) {
    case 0: // Cria o Socket UDP
            link->link = udp_new();
            if (link->link==NULL) { link->state = -1; break; }
            link->id_ack_pacote_enviado=-1;  // indica que ainda nenhum ACK foi feito
            if (udp_bind(link->link, IP_ADDR_ANY, 9945)!=ERR_OK) {link->state = -3; break; }
            udp_connect(link->link, IP_ADDR_BROADCAST, 9945);
            ASK_SIP[0]=0; // Pedir o IP do Servidor
            for (i=0;i<6;i++) ASK_SIP[i+1]=pucMACArray[i]; // Envia o MACADDRESS
            link->ptr2send = pbuf_alloc(PBUF_TRANSPORT, 1400, PBUF_RAM);
            if (link->ptr2send==NULL) { link->state = -2; break; }
            link->size_header=link->ptr2send->tot_len-1400;
            link->save2pckt_tot_len = link->ptr2send->tot_len;
            udp_recv(link->link,  recv_udp_packets, (void*)link);
            link->state=10;
            break;
    case 10: // ENVIA O PACOTE
            memcpy (link->ptr2send->payload, ASK_SIP, sizeof(ASK_SIP));
            link->ptr2send->tot_len=link->ptr2send->len=sizeof(ASK_SIP)+link->size_header;
            udp_send(link->link,link->ptr2send);
            link->id_pacote_enviado=0;  // PACOTE TIPO ZERO: PEDIDO IP SERVER ...
            link->ptr2send->tot_len = link->save2pckt_tot_len;
            link->time2timeout = g_ulSystemTimeSeconds+2;
            link->state=20;
            break;
    case 20: // REPETE ATE CONSEGUIR DE 2 EM 2 SEGUNDOS
            if (link->id_pacote_enviado==link->id_ack_pacote_enviado) {
                    //if (udp_bind(link->link, gv_serverip, 9945)!=ERR_OK) {link->state = -4; break; }
                    ip_addr.addr = gv_serverip;
                    udp_connect(link->link, &ip_addr, 9945);              // struct ip_addr *ipaddr
                    ASK_SIP[0]=250;  // diz que é para fazer o Pedido da Tabela de Acquisição e o pedido dos
                                     // Parâmetros do CAN-UPS
                    link->state=30;
                    break;
            }
            if (g_ulSystemTimeSeconds>link->time2timeout) link->state=10;
            break;
    case 30: // Vamos pedir para Carregar a Lista de Tabelas de Acquisição ...
            memcpy (link->ptr2send->payload, ASK_SIP, sizeof(ASK_SIP));
            link->ptr2send->tot_len=link->ptr2send->len=sizeof(ASK_SIP)+link->size_header;
            udp_send(link->link,link->ptr2send);
            link->id_pacote_enviado=250;  // PACOTE TIPO 250: Tab. aquisição + Param CAN
            link->ptr2send->tot_len = link->save2pckt_tot_len;
            link->time2timeout = g_ulSystemTimeSeconds+2;
            link->state=40;
    case 40: //-----------------------------------
            if (g_ulSystemTimeSeconds>link->time2timeout) link->state=30;
            if (link->id_pacote_enviado==link->id_ack_pacote_enviado)
            if (ga_sample_timings.linhas>0) { // Ok recebi a Tabela
                ip_addr.addr = gv_serverip2senddata;                  // liga para onde devemos mandar os dados...
                udp_connect(link->link, &ip_addr, 9945);              // struct ip_addr *ipaddr
                link->state=50;
                link->id_pacote_enviado=1;
                break;
            }
            break;
    case 50:
            return 20;
            break;
    case 60:
            link->tam_pack_a_enviar = size_lifo((tLIFO *)&gv_lifo_can);
            if (link->tam_pack_a_enviar > 1024) link->tam_pack_a_enviar=1024;
            read_lifo((tLIFO *)&gv_lifo_can, ((unsigned char*)link->ptr2send->payload)+1, link->tam_pack_a_enviar);
            link->state=61;
            break;
    case 61:
            ((unsigned char*)link->ptr2send->payload)[0]=link->id_pacote_enviado;
            link->ptr2send->tot_len=link->ptr2send->len=link->tam_pack_a_enviar+link->size_header+1;
            udp_send(link->link,link->ptr2send);
            link->ptr2send->tot_len = link->save2pckt_tot_len;
            link->time2timeout = g_ulSystemTimeSeconds+2;
            link->state=70;
            break;
    case 70:
            if (g_ulSystemTimeSeconds>link->time2timeout) link->state=61;
            if (link->id_pacote_enviado==link->id_ack_pacote_enviado) {
                link->state=50;
```

```
235                 link->id_pacote_enviado+=1;
                    if (link->id_pacote_enviado>40) link->id_pacote_enviado=1;
                    break;
                }
            }
240         return 0;
}
//---------------------------------------------------------------------------
```

## B.2 Conclusões

Este anexo apresentou muito sumariamente alguns aspectos práticos sobre a implementação dos três sensores utilizados para validar o sistema, sendo que o anemómetro digital foi configurado para enviar pela porta série os valores medidos de 1s em 1s, bastando apenas ao módulo E/S *CAN* ler e enviar pelo *CAN* quando solicitado. Quanto ao acelerómetro, apenas é necessário converter para digital a aceleração nos três eixos e efectuar o procedimento anterior.

Todos os esquemas foram implementados e testados, excepto a placa LM3S8962 por não existir equipamento no ISEC para a sua correcta soldadura. Neste caso utilizou-se um *Kit* de iniciação do fabricante.

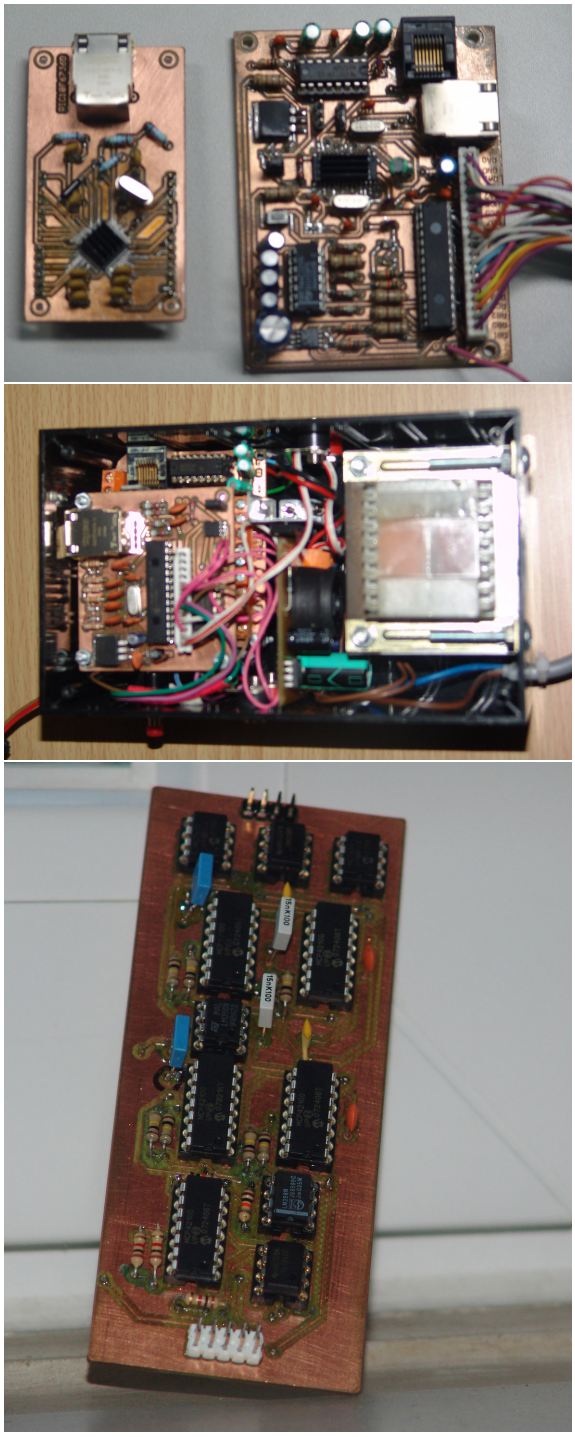


Figura B.5: Imagem de alguns dos protótipos de teste.

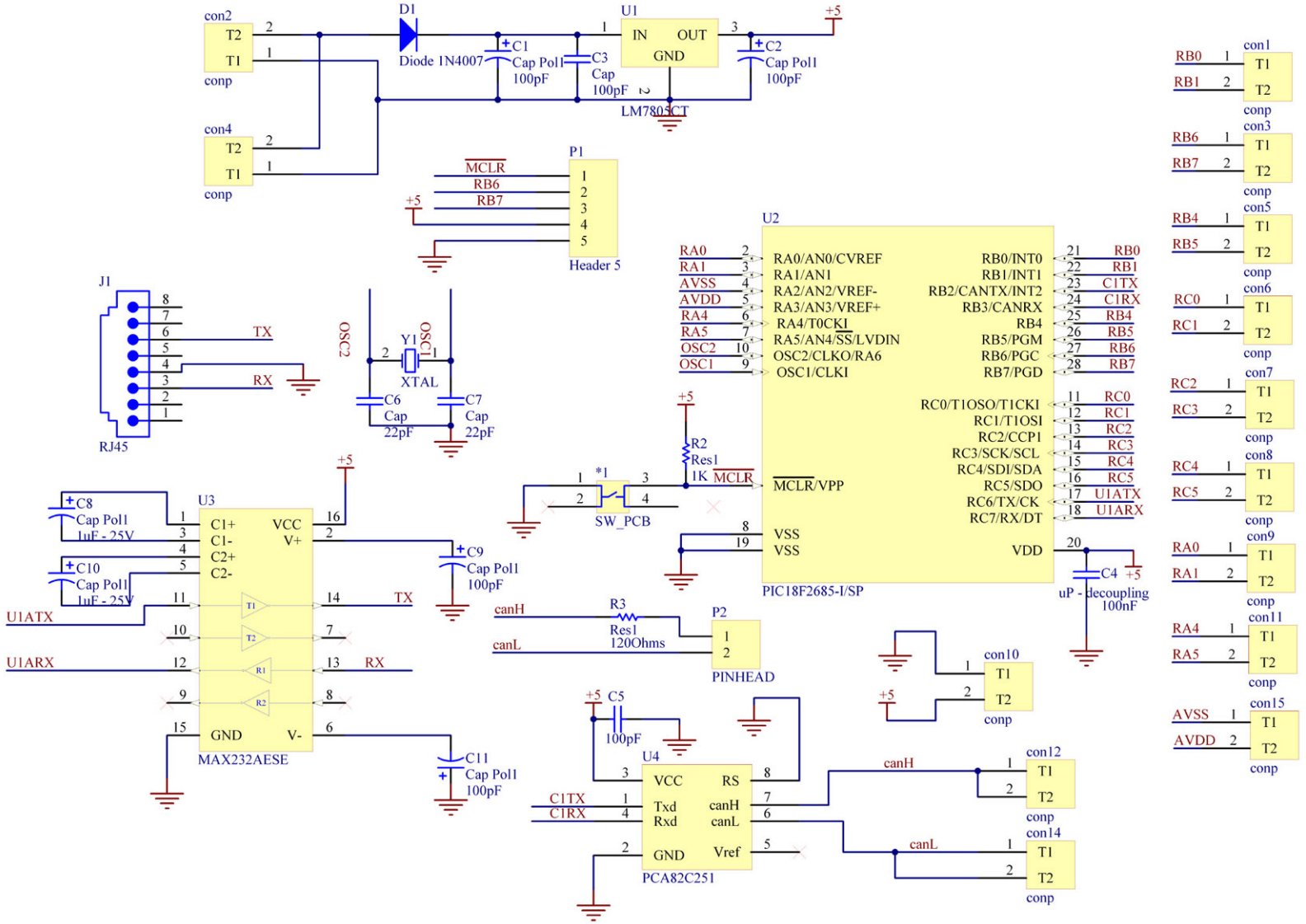



Figura B.6: Esquema da placa PIC18F2685.

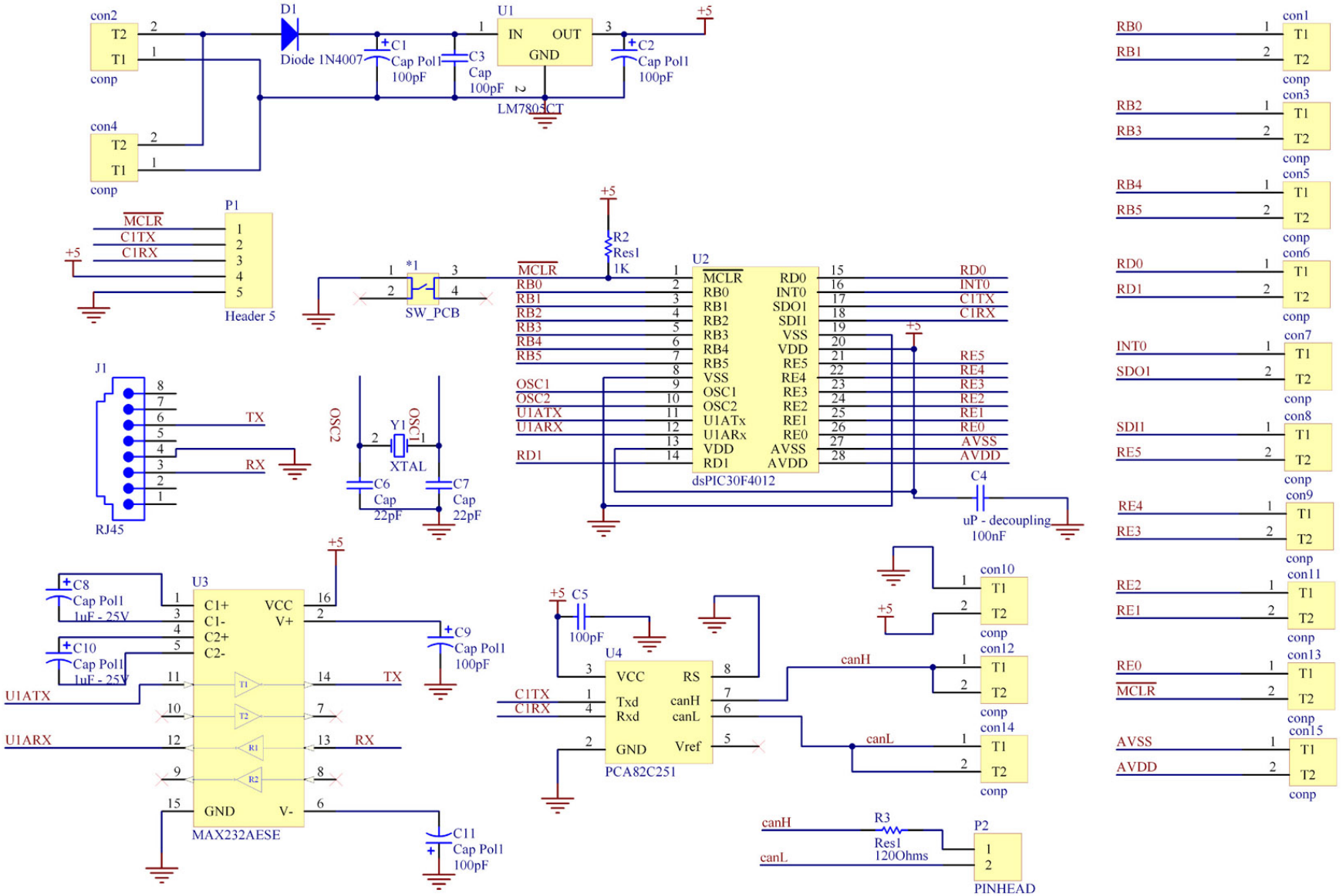



Figura B.7: Esquema da placa dsPIC30F4012.

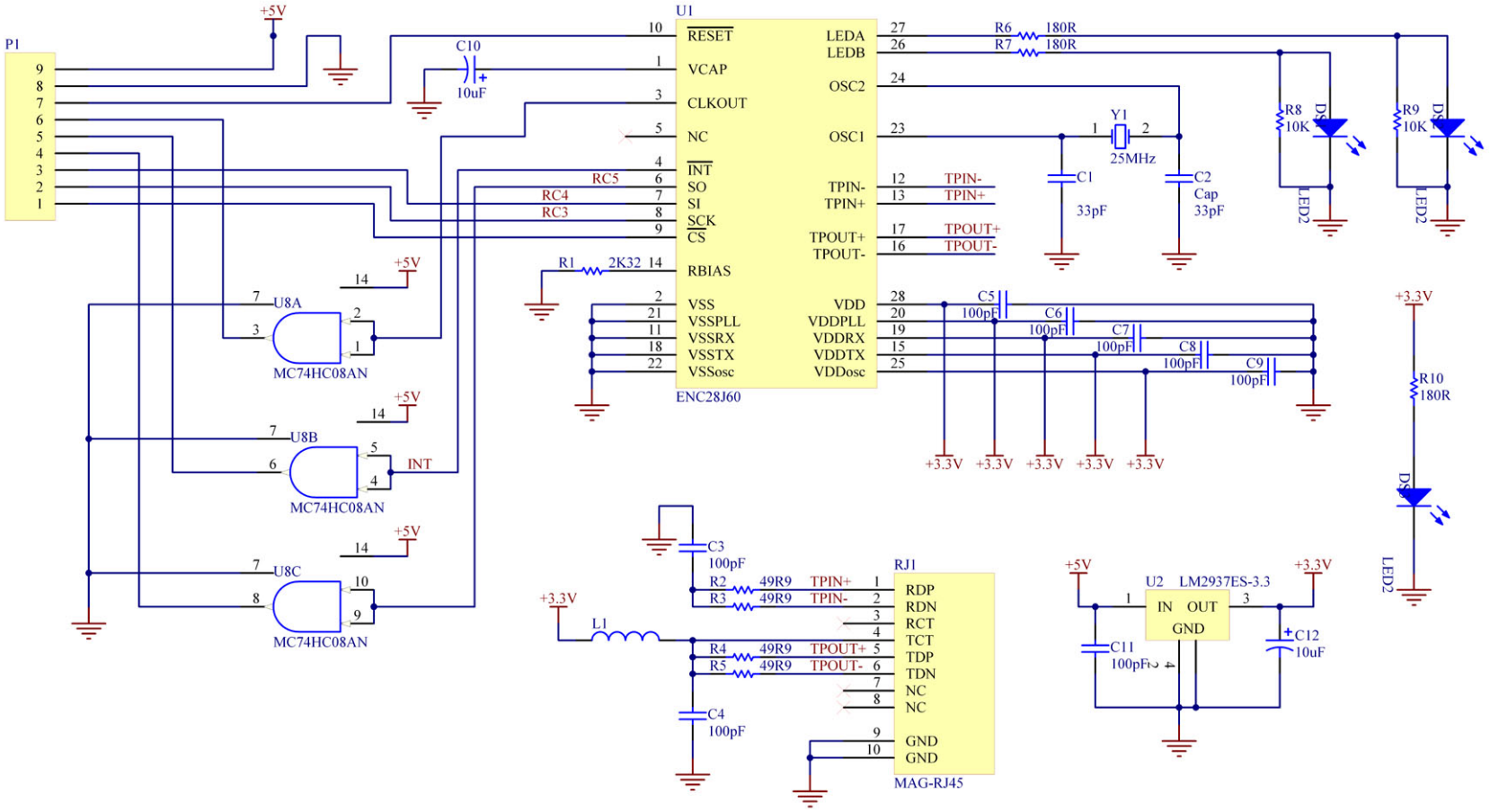



Figura B.8: Esquema da placa ligação *Ethernet* com ENC28J60.

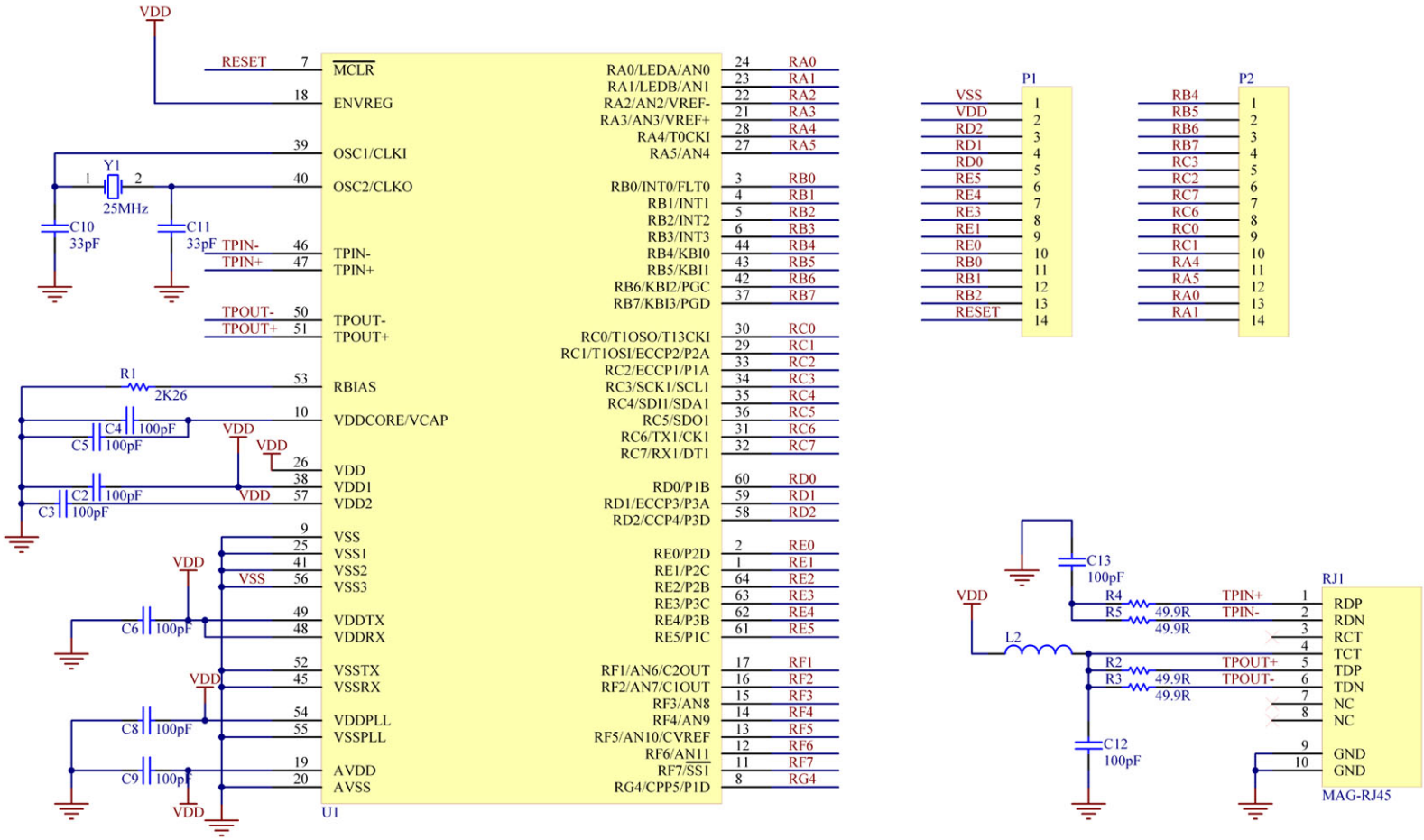



Figura B.9: Esquema da placa ligação *Ethernet* com PIC18F67J60.

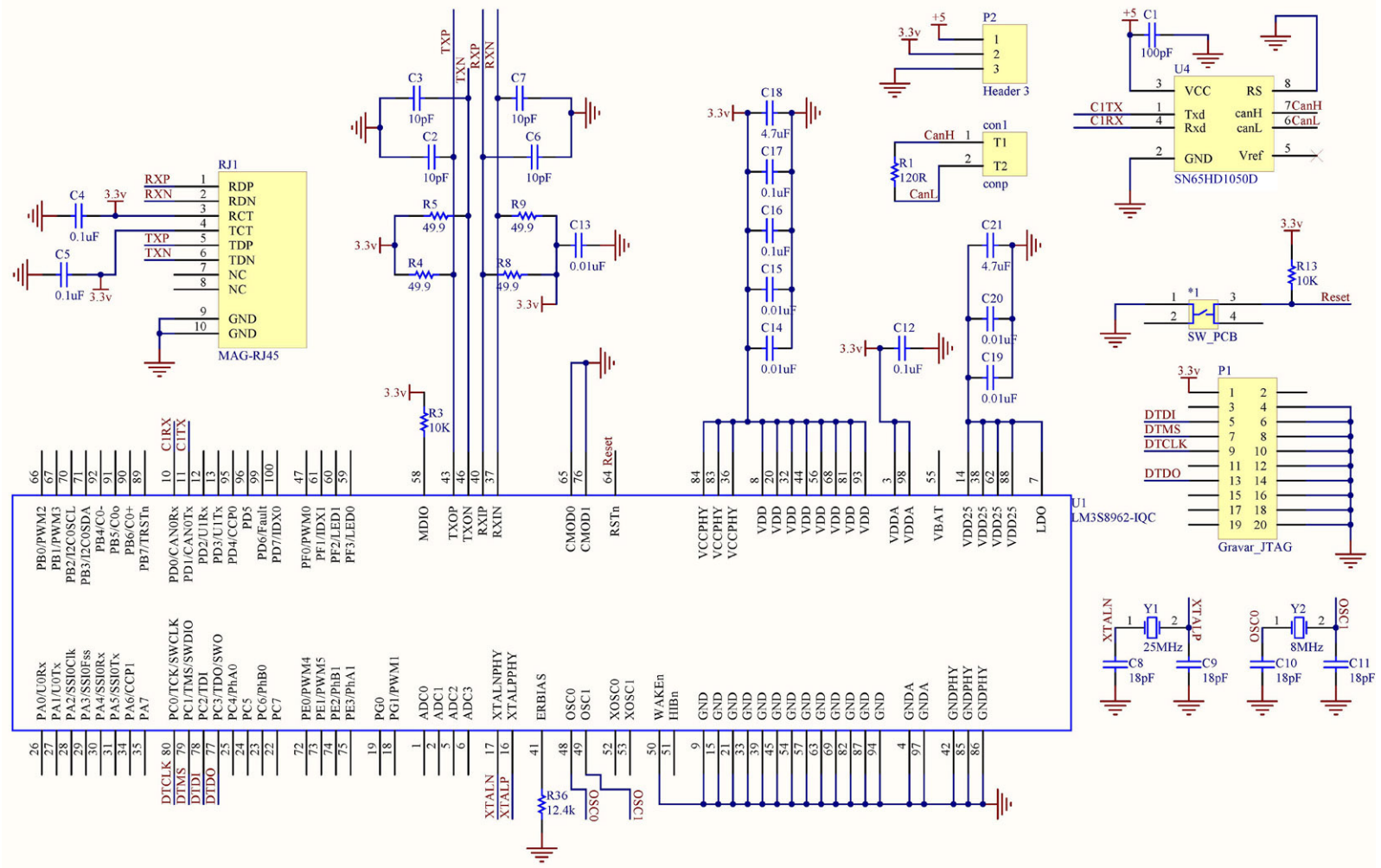



Figura B.10: Esquema da placa ligação *Ethernet* com Arm-Cortex LM3S8962.

# Anexo C

# Predição através de séries temporais

Para o estudo da problemática da manutenção de aero-geradores é importante salientar a importância das séries temporais na predição do "estado de saúde" dos equipamentos. Assim, o estudo e utilização das séries temporais enquadra-se num plano mais global, que é o de monitorizar as tendências de determinadas variáveis relevantes para a tomada de decisão na determinação atempada das intervenções de manutenção condicionada, antes das avarias se manifestarem em definitivo. Este anexo pretende completar a informação apresentada no capítulo 6. Se as previsões forem fiáveis, os gestores da manutenção poderão planear as suas intervenções de forma mais segura, atempada e eficiente.

As metodologias desenvolvidas ao longo das secções seguintes devem permitir a inclusão de um algoritmo de previsão de séries temporais no *software* de gestão de manutenção SMIT. É também importante a obtenção de algoritmos mais simples, que permitam antever a tendência de determinados parâmetros, logo ao nível da aquisição de dados, de forma a utilizar todo o poder de cálculo instalado. Obviamente, os resultados obtidos neste nível mais básico tendencialmente são menos precisos; no entanto, a ideia subjacente é o cálculo poder funcionar como um alerta para o sistema central (SMIT) e, posteriormente, efectuar um processamento mais detalhado através de um método mais elaborado, ou seja, o "equipamento monitor" prevê, mas apenas envia ao servidor do SMIT a amostra sem processamento, dando unicamente um alerta ao descortinar uma possível situação anómala.

Assim, na secção C.1, serão apresentados os métodos tradicionais, bem como algoritmos associados/adaptados para utilização no *software*/*hardware* desenvolvido no âmbito desta dissertação.

Na secção C.1.4 são detalhados os pormenores de cada um dos métodos com os respectivos formalismos.

## C.1 Estudo de métodos para previsão com séries temporais

### C.1.1 Séries temporais

A análise de séries temporais é uma das áreas que pretende compreender a evolução de dados obtidos ao longo do tempo, nomeadamente, características tais como auto correlação, sazonalida-

de, etc. Se a variável temporal da série for contínua, esta é contínua, se a variável for discreta, então a série temporal é discreta, sendo muitas vezes, na prática, uma variável aleatória da série temporal contínua. Salvo indicações em contrário, nesta dissertação serão apenas consideradas séries temporais discretas.

**Definição C.1** ***Série Temporal****: é uma sequência ordenada de valores de uma variável, igualmente espaçados ao longo do tempo.*

A análise de séries temporais tem por objectivo a compreensão do processo que dá origem aos dados da série, tendo como finalidade, nas principais aplicações, efectuar a previsão de tendências, monitorização, *feedback* ou controlo *feedforward*.

A investigação em séries temporais é transversal a várias áreas do conhecimento, desde a economia, vendas, *marketing*, mercado de acções, controlo de processos, censos, projecções em eleições, etc.

Na literatura encontram-se inúmeros métodos para análise de séries temporais, onde cada uma das técnicas é adequada ou está sintonizada para o cálculo de determinadas características da série. Exemplos disso são as técnicas de análise baseadas em modelos, onde se salienta [107, 108, 109, 110]:

- *Box-Jenkis ARIMA Models*;
- *Box-Jenkis Multivariate Models*;
- *Holt-Winters Exponential Smoothing (single, double, triple)*;
- SVR;
- *Neuronal Networks* [111, 112, 113];
- Etc.

As séries temporais são caracterizadas em cinco aspectos distintos: tendência; sazonalidade; uma variância mutável ao longo do tempo; não linearidade e pontos de influência divergentes.

Uma das principais aplicações nesta área é a da previsão sobre a evolução futura da sequência de dados. Na presente dissertação, este é o objectivo central da sua aplicação.

### C.1.2 Estado da Arte

O Livro publicado por *Box* e *Jenkins*[1] intitulado "*Time Series Analysis: Forecasting and Control*"[108], constituiu um primeiro passo para agregar os métodos publicados nesta área, à data da sua publicação, nomeadamente através da explicação e apresentação do modelo ARIMA, que considera que qualquer série temporal é um processo estocástico estacionário criado pela passagem

[1] Primeira edição em 1970

sucessiva de um processo de ruído branco por um modelo de médias móveis e auto regressivo. Avanços mais recentes introduziram variantes como é o caso do modelo GARCH, ARCH, ARIMA-GARCH, NARX, entre outros.

Os modelos ARCH, GARCH, pretendem modelizar momentos de ordem mais elevada, nomeadamente a variância, permitindo ainda a sua variação ao longo do tempo [107].

A técnica de *Holt* e *Winters* enquadra-se nas técnicas de alisamento exponencial bastante conhecida e utilizada em muitos trabalhos envolvendo séries temporais, cujo aparecimento data sensivelmente da mesma altura dos métodos de *Box* e *Jenkins* [108].

Inicialmente a estimativa dos valores futuros em séries temporais era feita pelo método ingénuo, cuja estimativa futura era dada pelo valor mais recente disponível.

Seguiu-se o método das médias móveis onde a estimativa era calculada pela média das últimas $n$ amostras. Ainda neste método, alguns autores usavam processos com médias móveis duplas, triplas, etc, sendo uma média móvel dupla dada pela média das últimas $n$ médias simples.

O método do alisamento exponencial apareceu então nesta altura sendo muito apreciado pela forma simples e eficiente em termos computacionais relativamente à forma como a próxima estimativa é calculada. Este método apresenta diferentes formulações e várias evoluções conforme se consideram séries com tendências ou com uma ou várias sazonalidades (diária, semanal, mensal, anual, etc.). Para o modelo do alisamento exponencial com sazonalidade, existem duas modelizações: o modelo sazonal aditivo e o modelo sazonal multiplicativo. As variantes mais recentes deste método incorporam modificações para modelos com tendência amortecida e modelos com erros estruturados [109]. No alisamento exponencial, de acordo com o método de *Brown*, os modelos podem ser lineares, quadráticos ou constantes. O valor da constante de amortecimento $\alpha$ deve ser elevado se a série variar muito, ou baixo quando o comprimento temporal da série é elevado, dando-se assim um peso maior ao passado da série. O método de *Holt*, na sequência do estudo de *Brown* [108] considera a existência de duas equações, a primeira idêntica ao método de *Brown* e a segunda que permite a adaptabilidade de $\alpha$ em função de um segundo parâmetro $\beta$. Sempre que a série apresentar sazonalidades, *Winters* reformulou o método do alisamento exponencial de *Holt* para "acomodar" a possibilidade de uma previsão correcta em séries com repetição ao longo do tempo, quer seja sazonalidade aditiva ou multiplicativa. Por último, o método estendido de *Holt-Winters* permite analisar séries com duas sazonalidades diferentes.

Entretanto, com as evoluções na área da inteligência artificial vários autores utilizaram redes neuronais para a predição em séries temporais. Existem vários modelos de redes, destacando-se as redes *feedforward* e as redes recorrentes, que não têm realimentação das camadas posteriores para as anteriores; no segundo caso, essa realimentação é utilizada. As redes neuronais mais indicadas para a análise das séries temporais são as que introduzem atraso temporal nas camadas de entrada ou efectuam realimentação entre camadas [111], nomeadamente, as redes TDNN, TLFN, TDFFN, ou redes RBF, que possuem a característica indicada anteriormente. As redes neuronais também foram usadas em conjunto com os métodos tradicionais, dando origem a métodos NARMA, NARIMA e NARX, onde se associam os métodos ARMA, ARIMA e AR com redes *feedforward*.

Ainda dentro desta prática, existem variantes da utilização de redes neuronais, introduzindo técnicas alternativas, das quais se destaca a utilização de filtros de *Kalman* para estimação dos pesos das ligações entre neurónios, ou da utilização de séries de *Fourier* ou *Wavelets* [112]. Um dos argumentos fortes para a utilização de redes neuronais é o facto de poderem ser utilizadas em séries não estacionárias e fortemente não lineares [113]. Ainda na área de inteligência artificial é hoje comummente aceite a utilização de algoritmos genéticos para a criação de algoritmos de regressão simbólica que consistem na utilização de determinadas operações base equivalentes aos conceitos da evolução genética conhecidos na natureza, tais como a combinação genética, a sobrevivência dos mais aptos, as mutações, etc. O conceito de algoritmos genéticos é inspirado na teoria da evolução das espécies de *Darwin*. Na realidade, trata-se de um algoritmo de optimização estocástico, que utiliza técnicas de "probabilidades" para ultrapassar o problema dos mínimos locais associados a outros métodos de pesquisa baseados em funções determinísticas. Segundo a opinião de alguns autores [130], a convergência para uma solução não é assegurada e, normalmente, existindo outra técnica conhecida para a resolução do problema com sucesso, aconselha-se a não utilização de algoritmos genéticos. Uma das particularidades determinante para o sucesso dos algoritmos deste género é a codificação da população e dos operadores [115, 116]. Muitos autores utilizam a representação em árvore binária para resolução do problema de regressão simbólica [117, 118].

Uma das técnicas de classificação mais recente denomina-se de SVM, um competidor actual das redes neuronais nos problemas de classificação; SVR no caso da regressão. O SVM utiliza um mapeamento de $\Re^n$ para $\Re^n$ através de várias funções, uma das quais é a função RBF. Na prática, tenta-se utilizar o mapeamento para linearizar o problema, encontrar uma solução nesse espaço, através de técnicas lineares e de seguida transpor a solução para o espaço de origem. Actualmente, esta técnica tem ganho muitos adeptos comparativamente às redes neuronais, sendo uma das técnicas mais utilizadas [119]. A motivação da escolha reside no facto do algoritmo de SVM ser mais fácil de treinar quando comparado com as redes neuronais, mantendo ainda um bom desempenho relativo. Na análise de séries temporais por algoritmos SVR, alguns autores afirmam ter aplicado com sucesso este método à análise de séries não estacionárias [114].

No âmbito dos métodos de análise no domínio da frequência, destaca-se a tradicional análise em série de *Fourier* (apresentada por *Joseph Fourier* em 1807). A série de *Fourier* permite efectuar a decomposição de um sinal em função de várias frequências, mas pressupõe o conhecimento de todo o sinal. Em problemas práticos, nomeadamente na análise de séries temporais, a série de *Fourier* necessita de ser aplicada através de janelas como é o caso das implementações práticas da FFT ou DFT. O tamanho da janela é preponderante, tendo influência no resultado. Um dos problemas da série de *Fourier* é exactamente a necessidade da estacionariedade do sinal observado, dado que em problemas reais o sinal não é conhecido na totalidade. Assim, por exemplo, para as funções:

$$f_1(t) = cos(2.\Pi.f_1.t) + cos(2.\Pi.f_2.t) \; e \; f_2(t) = \begin{cases} cos(2.\Pi.f_1.t), \; t \leq 0 \\ cos(2.\Pi.f_2.t), \; t > 0 \end{cases} \tag{C.1}$$

após o cálculo da transformada de *Fourier*, ambos os sinais apresentam semelhanças espectrais

evidentes, o que não permite uma correcta análise no domínio da frequência. Para responder a este facto foi desenvolvida a análise *wavelet* em 1930 por *Haar* na sua tese de doutoramento. Só na década de 90 o uso das *wavelets* se generalizou, nas áreas da codificação de vídeo, compressão, representação e filtragem de sinais. Entretanto, vários estudos conduziram a algumas variantes, nomeadamente os métodos *wavelets-packets* e *multi-wavelets*. A grande vantagem da transformada de *wavelet* é a possibilidade de análise no espaço tempo-frequência, permitindo analisar as diversas frequências em intervalos de tempo específicos, típico nas séries temporais. Na análise de séries temporais, a transformada de *wavelet* tem sido utilizada em conjunto com redes neuronais *feedforward* para predição temporal, sendo utilizada em análise de séries económicas [112].

### C.1.3 Séries temporais - métodos mais relevantes

A previsão a partir de séries temporais obriga à introdução de termos matemáticos [131]. Assim, se a previsão dos valores futuros da série puderem ser obtidos através de funções matemáticas, a série denomina-se determinística. Se a previsão dos valores futuros puder ser feita apenas por modelos estatísticos, através da função densidade de probabilidade, a série é denominada de estocástica. Estas séries são uma amostra de um processo estocástico que, na prática, representa um fenómeno estatístico com lei probabilística ao longo do tempo. A série temporal estocástica é uma de várias amostras possíveis desse mesmo processo.

**Definição C.2** *Um processo estocástico temporal* $Y = \{y(t,\theta)\,;\, t \in T,\ \theta \in \Omega\}$*, é um conjunto de variáveis aleatórias definidas no espaço probabilístico* $\Omega$ *e indexado no tempo* $T$*. Para uma concretização de* $\theta$ *e discretizando t obtêm-se assim uma série temporal estocástica* $y[k,\theta_{constante}]$ *que, resumidamente, se representa por* $y[k]$*, ou* $y_k$*. A variável k é assim uma realização discreta da variável temporal, amostrada em determinados instantes. A variável* $\theta$ *representa acontecimentos probabilísticos.*

Um dos aspectos importantes na análise das séries temporais refere-se às propriedades do processo estocástico que lhe deu origem. Dado que este tem dependência do tempo, convém efectuar algumas definições relativamente ao seu comportamento temporal.

**Definição C.3** *Um processo estocástico temporal é estacionário se for invariante no tempo, não sendo afectado por um deslocamento no tempo,* $y(t,\theta) = y(t+\Delta t,\theta),\ \forall\, \Delta t \in T$*, i.e., a distribuição de probabilidade conjunta é a mesma independentemente do tempo.*

**Definição C.4** *A média, variância, auto-covariância e auto-correlação de uma série temporal são dadas, respectivamente, por:*

$$\begin{gathered} Média = \mu(t) = E\{y(t,\theta)\} \\ Variância = \sigma(t) = Var\{y(t,\theta)\} = E\left\{y^2(t,\theta)\right\} \\ Autovariância = \gamma(t_1,t_2) = E\left\{(y(t_1,\theta) - \mu(t_1)).(y(t_2,\theta) - \mu(t_2))\right\} \end{gathered}$$

*onde a variância é um caso particular da autovariância, considerando* $t_1 = t_2 = t$

**Definição C.5** *Um processo estocástico temporal é estacionário de segunda ordem, se* $E\{y[k]\}$, $E\{y[k_1].y[k_2]\} = 0$ *e* $E\{y^2[k]\}$ *não dependem de k (k indexado a t-tempo).*

Das definições anteriores, constata-se que num processo estocástico temporal estacionário a média é constante dado que não depende do tempo. A visualização gráfica pode ser efectuada através de um plano bidimensional onde uma das coordenadas é o tempo e a outra coordenada a variável aleatória.

**Definição C.6** *Um processo estocástico* $wn[k]$ *é considerado ruído branco, se for estacionário e a esperança* $E\{wn[k]\} = const$, $E\{wn[k_1].wn[k_2]\} = 0$ *e* $E\{wn^2[k]\} = \sigma^2$.

A definição C.6 é importante para construir os próximos modelos que serão descritos na secção seguinte.

#### C.1.3.1 Modelo ARMA

Os modelos auto-regressivos de médias móveis [107, 110], são a junção dos modelos AR (auto regressivos) com os modelos MA (médias móveis), que constituem uma das metodologias de análise de séries temporais existentes na literatura da especialidade. Esta metodologia deu origem a variações como é o caso dos métodos ARIMA, ARMAX, ARX e outros, descritos no texto abaixo. Nos métodos ARMA, os dados experimentais são usados para determinar se verificam o modelo e, caso se confirmem as condições do modelo, então o método pode ser utilizado. Caso contrário, evidentemente, o modelo é incorrecto para os dados concretos.

Para uma apresentação mais compacta, considere-se o operador deslocamento temporal **L** como:

$$Ly[k] = y[k-1]$$

com as propriedades $L^{-n}y[k] = y[k+n]$, $L^n y[k] = y[k-n]$ e $L^0 y[k] = y[k]$. A aplicação do operador L em polinómios é descrita de tal forma que:

$$a_p(L) = a_0 + a_1.L + \cdots + a_p.L^p$$

e os respectivos polinómios aplicados a séries temporais de acordo com a expressão:

$$a_p(L).y[k] = a_0.y[k] + a_1.y[k-1] + \cdots + a_p.y[k-p]$$

onde, na expressão, **p** denomina a ordem do polinómio.

**Definição C.7** *Uma série temporal* $y[k]$ *segue um modelo ARMA(p,q) de ordem* ***p***, ***q*** *se verifica a equação:*

$$a_p(L).y[k] = b_q(L).wn[k] \tag{C.2}$$

*onde* $wn[k]$ *representa um processo com ruído branco de média nula e variância constante, sendo as variáveis* $a_1, \cdots, a_p$, $b_0, \cdots, b_q$ *parâmetros do modelo.*

Na definição C.7 verifica-se que os parâmetros $a_i$ correspondem aos pesos de um filtro recursivo linear e os parâmetros $b_i$ aos parâmetros de um filtro não recursivo. Neste modelo, para $p = 0$, obtém-se um modelo $MA(q)$ e, para $q = 0$, obtém-se um modelo $AR(p)$. Os polinómios $a_p(L)$ e $b_q(L)$ podem ser estudados de forma semelhante aos polinómios normais, com o grau de abstracção relativamente ao operador $L$. Conhecendo-se então as raízes das equações $a_p(L) = 0$ e $b_q(L) = 0$, é possível definir critérios de estabilidade do modelo, nomeadamente através da localização das raízes relativamente a um círculo unitário [108].

### C.1.3.2 Modelos ARMAX

O modelo ARMAX é uma das várias generalizações do modelo ARMA [107]. Este método modeliza a série $y[k]$ em dependência de uma terceira série (uma variável exógena). Este facto evidencia uma dependência indexante de $y[k]$. Exemplos claros e inequívocos encontram-se na área financeira, onde o estudo das séries associadas às cotações das acções, em que a dependência de cada cotação, muitas vezes, está indexada a outras séries externas, tais como, por exemplo, o anúncio de resultados operacionais das empresas, a medida de confiança dos empresários, consumidores e investidores, as previsões de crescimento, etc. Este modelo pretende assim conciliar a dependência externa de uma variável e, em muitos casos, pode ser generalizado para mais do que uma dependência. A sua representação matemática pode ser descrita por:

**Definição C.8** *Uma série temporal $y[k]$ segue um modelo $ARMAX(p,q,j)$ de ordem **p**, **q**, **j** se e só se verificar a equação:*

$$a_p(L).y[k] = b_q(L).wn[k] + c_j(L).x[k] \tag{C.3}$$

*onde $wn[k]$ representa um processo com ruído branco de média nula e variância constante, $x[k]$ é a série da qual $y[k]$ depende e as variáveis $a_1, \cdots, a_p$, $b_0, \cdots, b_q$, $c_0, \cdots, c_j$ são parâmetros do modelo.*

O aspecto complexo neste modelo é a determinação dos valores de **p**, **q** e **j** para um dado conjunto de dados. Evidentemente, o modelo pode ser expansível a mais do que uma dependência de $y[k]$ a séries externas.

### C.1.3.3 Modelo ARIMA

Os modelos ARIMA, são semelhantes aos modelos ARMA [110]. Uma série temporal satisfaz as condições de um modelo ARIMA, se a sua derivada de ordem **n** satisfaz um modelo ARMA. Na prática, uma série temporal segue um modelo ARIMA(p,n,q) se a sua derivada de ordem **n** segue um modelo ARMA(p,q). Este modelo é utilizado quando a série temporal original é não estacionária. Prova-se que para uma série não estacionária existe um valor **n** para o qual a derivada de ordem **n** da série original dá origem a uma série estacionária ou pseudo-estacionária. Assim, tem-se para um modelo ARIMA(p,n,q):

$$a_p(L).\nabla^n y[k] = b_q(L).wn[k] \tag{C.4}$$

onde $\nabla^n y[k]$ denomina a derivada de ordem **n** da série temporal. Curiosamente, fazendo **n=0** está-se na situação de um modelo ARMA. Por consequência, o modelo ARIMA é mais generalista. Para efectuar o cálculo da derivada, será utilizada a aproximação obtida através da série de *Taylor* em redor do ponto $t_1$, cuja expressão é:

$$\begin{gathered} f(t) = f(t_1) + f'(t_1).(t-t_1) + \frac{f''(t_1)}{2!}.(t-t_1)^2 + \cdots + \frac{f^{(n)}(t_1)}{n!}.(t-t_1)^n = \\ = \sum_{n=0}^{+\infty} \frac{f^{(n)}(t_1)}{n!}.(t-t_1)^n \end{gathered} \tag{C.5}$$

A Eq. C.5 pode ser aplicada a séries discretas, tendo em conta a sua validade para pontos próximos do ponto central $t_1$. Considerando $t = t_0$, assim:

$$\begin{gathered} f'(t_1) = \frac{f(t_0) - f(t_1)}{t_0 - t_1} - \frac{f''(t_1)}{2!}.(t_0 - t_1) - \cdots - \frac{f^{(n)}(t_1)}{n!}.(t_0 - t_1)^{n-1} = \\ = \frac{f(t_0)-f(t_1)}{t_0-t_1} - \sum_{n=2}^{+\infty} \frac{f^{(n)}(t_1)}{n!}.(t_0 - t_1)^{n-1},\ para\ t_1 > t_0 \end{gathered}$$

Supondo $t_1 - t_0 = \Delta t > 0$ temos:

$$f'(t_1) = \frac{f(t_1)-f(t_0)}{\Delta t} + \sum_{n=2}^{+\infty} \frac{f^{(n)}(t_1)}{n!}.(\Delta t)^{n-1} \approx \frac{f(t_1)-f(t_0)}{\Delta t}$$

Em consequência, podemos aproximar a primeira derivada pela diferença de 1ª ordem por:

$$\Delta f(t_1) = \frac{f(t_1) - f(t_1 - \Delta t)}{\Delta t} \tag{C.6}$$

a segunda derivada será então aproximada pela diferença de segunda ordem:

$$\Delta^2 f(t_1) = \frac{\Delta f(t_1) - \Delta f(t_1 - \Delta t)}{\Delta t} = \frac{f(t_1) - 2.f(t_1 - \Delta t) + f(t_1 - 2.\Delta t)}{\Delta t^2} \tag{C.7}$$

a terceira derivada aproximada pela diferença de terceira ordem:

$$\begin{gathered} \Delta^3 f(t_1) = \frac{\Delta^2 f(t_1) - \Delta^2 f(t_1 - \Delta t)}{\Delta t} = \\ = \frac{f(t_1) - 3.f(t_1 - \Delta t) + 3.f(t_1 - 2.\Delta t) - f(t_1 - 3.\Delta t)}{\Delta t^3} \end{gathered} \tag{C.8}$$

a quarta derivada aproximada pela diferença de quarta ordem:

$$\begin{gathered} \Delta^4 f(t_1) = \frac{\Delta^3 f(t_1) - \Delta^3 f(t_1 - \Delta t)}{\Delta t} = \\ = \frac{f(t_1) - 4.f(t_1 - \Delta t) + 6.f(t_1 - 2.\Delta t) - 4.f(t_1 - 3.\Delta t) + f(t_1 - 4.\Delta t)}{\Delta t^4} \end{gathered} \tag{C.9}$$

e, genericamente,

$$\begin{gathered} \Delta^n f(t_1) = \frac{\Delta^{n-1} f(t_1) - \Delta^{n-1} f(t_1 - \Delta t)}{\Delta t} = \\ = \sum_{i=0}^{n} (-1)^{n-i} . \frac{n!}{i!(n-i)!} \frac{f(t_1 - i\Delta t)}{\Delta t^n} = \sum_{i=0}^{n} (-1)^{n-i} . C_i^n . \frac{f(t_1 - i\Delta t)}{\Delta t^n} \end{gathered} \tag{C.10}$$

onde $C_i^n = \binom{n}{i} = \frac{P_i^n}{i!} = \frac{n!}{i!(n-i)!}$ são as combinações de **n**, **i** a **i** elementos e $P_i^n$ as permutações de **n**, **i** a **i**. Aplicando este resultado a uma série temporal y[k], supondo $\Delta t = 1$ como a unidade de tempo (segundos ou minutos, horas, dias, semanas, meses, anos), tem-se para a formulação genérica:

$$\nabla^n y[k] \approx \Delta^n y[k] = \Delta^{n-1} y[k] - \Delta^{n-1} y[k-1] = \sum_{i=0}^{n} (-1)^{n-i} . C_i^n . y[k-i] \tag{C.11}$$

de notar que as combinações na Eq. C.11, à medida que **n** cresce, originam termos de valor elevado, dando origem a erros de aproximação devido à representação em vírgula flutuante nos sistemas computacionais e, ainda, à circunstância de que, para calcular a diferença de ordem **n** serem necessários apenas valores do passado, o que é importantíssimo para o cálculo *on-line* da derivada.

Evidentemente, podem-se conjugar os modelos ARIMA e ARMAX ou outras combinações, em função do número de dependências externas.

#### C.1.3.4 Modelos SARIMA

O modelo SARIMA [108] é uma generalização do modelo ARIMA para lidar com sazonalidade. Admitindo que uma série y[k] é sazonal com período **sz**, i.e, y[k] depende de alguma forma de y[k-sz], para a correcta formulação matemática será fundamental a introdução de um novo operador. A necessidade de reformulação do modelo ARIMA para séries sazonais advém do facto (não esquecendo que este modelo é aplicado a séries não estacionárias) de que a 1ª derivada não é suficiente para que $\nabla y[k]$ seja estacionária, apesar de, note-se, $\nabla^{sz} y[k]$ ser, necessariamente, estacionária, no que se refere à sazonalidade. No entanto, a diferença de ordem **sz** depende das combinações de **sz**, tal como descrito pela Eq. C.11; sabendo-se o período, a sazonalidade seria removida ao efectuar a diferença y[k]-y[k-sz]. Assim, é relevante introduzir o operador derivada sazonal como:

$$\nabla_{sz} y[k] \approx \Delta_{sz} y[k] = (1 - L^{sz}) y[k] = y[k] - y[k-sz]$$

permitindo remover a componente de sazonalidade apenas com uma diferença, em oposição a derivar **sz** vezes a série y[k].

Analogamente, é possível definir a derivada sazonal de ordem N cuja designação é dada por:

$$\nabla_{sz}^N y[k] \approx \Delta_{sz}^N y[k]$$

Combinando os dois tipos de diferenciação, obtém-se o operador:

$$\nabla^n \nabla_{sz}^N y[k] \approx \Delta^n \Delta_{sz}^N y[k]$$

Neste momento, é possível definir matematicamente o modelo SARIMA, através de:

**Definição C.9** *Uma série temporal y[k] segue um modelo SARIMA*$(p,n,q) \times (P,N,Q)_{sz}$ *se verifica a equação*

$$a_p(L) . c_P(L^{sz}) . \nabla^n \nabla_{sz}^N y[k] = b_q(L) . d_Q(L^{sz}) . wn[k] \tag{C.12}$$

*onde $wn[k]$ representa um processo com ruído branco de média nula e variância constante e as variáveis $a_1,\cdots,a_p$, $b_0,\cdots,b_q$, $c_0,\cdots,c_P$, $d_0,\cdots,d_Q$ são parâmetros do modelo.*

Os polinómios são definidos por:

$$\begin{aligned} a_p(L) &= a_0 + a_1.L + \cdots + a_p.L^p \\ b_q(L) &= b_0 + b_1.L + \cdots + b_q.L^q \\ c_P(L^{sz}) &= c_0 + c_1.L^{sz} + \cdots + c_P.L^{P.sz} \\ d_Q(L^{sz}) &= d_0 + d_1.L^{sz} + \cdots + d_Q.L^{Q.sz} \end{aligned}$$

Apesar deste modelo aparentar alguma complexidade, o que se verifica, na prática, segundo exemplos na literatura da especialidade, é o facto da ordem das derivadas não ser muito elevada e, portanto, os valores de **n** e **N**, normalmente não serem superiores a 1.

Ainda assim, deduzindo a fórmula para o operador $\nabla_{sz}y[k] \approx \Delta_{sz}y[k]$, obtém-se uma equação semelhante à Eq. C.11, pois facilmente se conclui:

$$\nabla^N_{sz}y[k] \approx \Delta^N_{sz}y[k] = \Delta^{N-1}_{sz}y[k] - \Delta^{N-1}_{sz}y[k-sz] = \sum_{j=0}^{N}(-1)^{N-j}.C^N_j.y[k-j.sz] \tag{C.13}$$

aplicando a Eq. C.11, obtém-se:

$$\nabla^n\nabla^N_{sz}y[k] \approx \Delta^n\Delta^N_{sz}y[k] = \textstyle\sum_{i=0}^{n}(-1)^{n-i}.C^n_i.\Delta^N_{sz}y[k-i]$$

e conjugando as duas equações anteriores, obtêm-se o resultado pretendido:

$$\nabla^n\nabla^N_{sz}y[k] \approx \Delta^n\Delta^N_{sz}y[k] = \sum_{i=0}^{n}(-1)^{n-i}.C^n_i.\sum_{j=0}^{N}(-1)^{N-j}.C^N_j.y[k-i-j.sz] \tag{C.14}$$

#### C.1.3.5 Modelos ARCH e GARCH

Os modelos ARCH e GARCH [107] assumem o facto do momento de segunda ordem - a variância - não ser constante ao longo do tempo. Os outros métodos apresentados consideravam, pelo contrário, uma variância constante. A variância incondicional (de toda a série) pode ser constante, mas a variância condicional (de parte da série, tirada ao longo do tempo) pode-se alterar e, normalmente, varia ao longo do tempo. A variância condicional, na análise de séries económicas, é denominada de volatilidade.

#### C.1.3.6 Modelo de previsão baseado na série de *Taylor*

Na secção C.1.3.3 foi apresentada a série de *Taylor* (Eq. C.5) e, através de aproximações, descreveu-se a derivada de ordem *n* de uma série em função da diferença de ordem **n** dessa mesma série (Eq. C.11), não esquecendo tratar-se de uma aproximação. Analisando a própria série de *Taylor*, constata-se que encerra em si uma forma de previsão e, generalizando, para uma versão

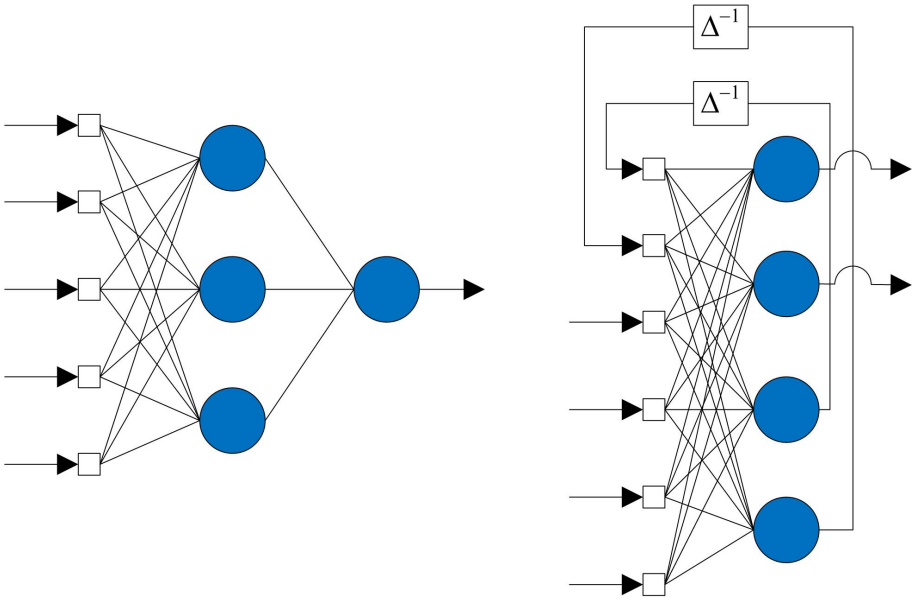



Figura C.1: À esquerda uma rede *feedforward*. À direita uma rede recorrente com realimentação de estado.

discreta poderia ser re-escrita como:

$$\hat{y}[k+m] =$$

$$= y[k] + \Delta y[k].(m+1).\Delta t + \frac{\Delta^2 y[k]}{2!}\left((m+1).\Delta t\right)^2 + \cdots + \frac{\Delta^n y[k]}{n!}\left((m+1).\Delta t\right)^n = \tag{C.15}$$

$$= \sum_{n=0}^{+\infty} \frac{\Delta^n y[k]}{n!}\left((m+1).\Delta t\right)^n$$

Evidentemente que a aproximação só é válida pelas próprias condições da série de *Taylor* para $m$ e $\Delta t$ pequenos. De notar ainda uma alteração na notação, dado $\hat{y}[k+m]$ significar a estimativa para o valor de $y[k+m+1]$, razão necessária para, no somatório, aparecer $m+1$ em detrimento de $m$. Este facto foi previamente explicado no capítulo 6, secção 6.2.3.1 (pág. 98). Com base nesta fórmula aqui apresentada será desenvolvido um método para previsão de valores futuros descrito na secção C.1.4.

### C.1.3.7 Redes Neuronais na previsão de séries

As redes MLP [111, 112, 113] são formadas por várias camadas de neurónios, as quais têm a capacidade de lidarem com problemas não linearmente separáveis. Existem duas modalidades de redes neuronais, *feedforward* e recorrentes, relativamente à classificação de redes com e sem memória.

A Fig. C.1 apresenta duas topologias de redes neuronais MLP, onde o critério do processamento temporal é preponderante para o estudo realizado nesta dissertação. O processamento temporal é alcançado pela inclusão de memória na rede, obtida através da inserção de atrasos temporais.

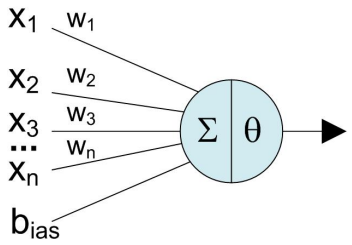



Figura C.2: Exemplo de um neurónio na camada $j$, elemento de processamento básico de uma rede neuronal.

Dado que as redes neuronais podem ter várias camadas, existem diferentes variantes em função do número de camadas e da forma como os atrasos temporais são inseridos, o local de proveniência e o local de destino. Exemplos de redes neuronais recorrentes são o caso da rede de *Elman*, *Jordan*, LRN, NARX, NARMA e TDNN. Nas MLP cada neurónio é representado por um somatório de um conjunto de entradas pesadas e de uma função de activação. A Fig. C.2 apresenta a unidade básica de processamento - o neurónio - onde a saída é dada pela seguinte equação:

$$\begin{cases} y = \theta(sx) \\ sx = \sum_{i=1}^{n} w_i.x_i + bias \end{cases}$$

A função $\theta(x)$ é denominada de função de activação e pretende simular as características não lineares dos neurónios. Exemplos de funções de activação: função linear; função degrau; função sigmóide; função ERF e função tangente hiperbólica. Estas funções são exemplificadas na Tab. C.1. Existem, no entanto, muitas mais funções de activação usadas na literatura da especialidade, onde se destacam as seguintes: seno; co-seno; arco de tangente; linear por partes; y=1/(1+|y|); e gaussiana. A mais utilizada é a função sigmóide.

Uma das técnicas apresentadas por alguns autores utiliza uma mistura de modelos ARMA com redes neuronais, dando origem a um método descrito como NARMA e, caso se utilizem as derivadas da série original, ao método NARIMA. A Fig. C.3 mostra graficamente o método NARMA.

### C.1.3.8 Algoritmos genéticos na previsão de séries

No algoritmo genético [117, 118, 119] gera-se uma população de indivíduos, sendo cada um uma solução possível para o problema; num segundo passo os indivíduos são avaliados individu-

| | | | | |
|---|---|---|---|---|
| $\theta(x) = a.x$ | $\theta(x) = \frac{1}{1+e^{-x}}$ | $\theta(x) = \frac{1-e^{-x}}{1+e^{-x}}$ | $\theta(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0 \end{cases}$ | $ERF(u) = \frac{2}{\sqrt{\Pi}} \int_0^u e^{-t^2} dt$ |

Tabela C.1: Funções de activação: função linear; função sigmóide; função tangente hiperbólica; função degrau e função ERF.

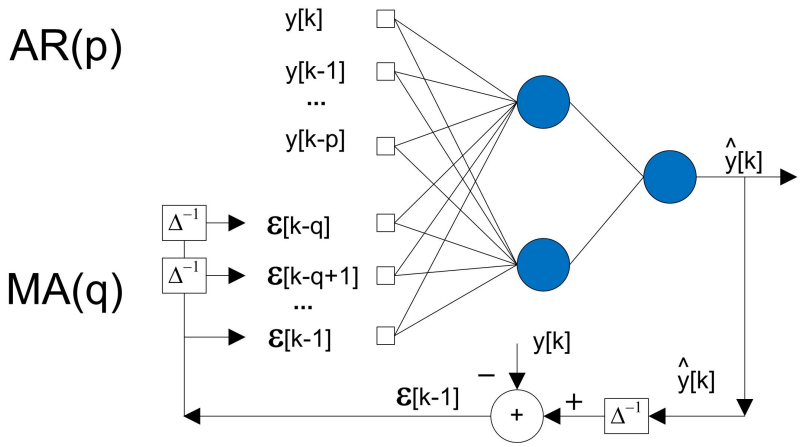



Figura C.3: Modelo neuronal de uma rede *feedforward* adaptada em conjunto com o método ARMA, dando origem ao método denominado de NARMA.

almente de forma a associar um grau de aptidão relativamente a serem uma boa solução. Após avaliados, os indivíduos mais aptos são seleccionados e cruzam com outros indivíduos, ou sofrem mutações. As propriedades de cada indivíduo são denominadas de cromossomas e, através de uma operação de reprodução (*crossover*), um ou mais indivíduos dão origem a novos indivíduos com algumas características de todos.

O processo da mutação consiste em seleccionar um indivíduo e, aleatoriamente, alterar uma das partes do cromossoma dando assim origem a um novo indivíduo. Normalmente, o processo de mutação deve ocorrer com uma taxa de probabilidade baixa, regularmente 1% para cada posição.

Os processos de mutação e reprodução dão origem a uma nova população (denominada de geração seguinte - filhos). O mesmo processo efectuado nas gerações seguintes dará origem às sucessivas gerações de forma cíclica até que a população venha a convergir para uma solução óptima ou se tenha atingido um número pré-determinado de gerações. Existe ainda um método para desempate, determinando quem são os indivíduos que se cruzam entre si, chamado de torneio aleatório, os quais são sorteados por método aleatório. Outros operadores genéticos existentes são: clonagem; elitismo; transposição; etc.

Na Programação Genética (PG), com o objectivo de resolver problemas de regressão simbólica, um indivíduo assume a forma de expressões matemáticas simbólicas das quais, como exemplo, se destacam dois casos: A Tab. C.2 mostra três codificações possíveis para cada expressão matemática simbólica. Em a) a equação; em b) versão usada pela linguagem de inteligência artificial - *LISP*; em c) uma representação em árvore muito utilizada na regressão simbólica e em d) uma outra representação obtida ao visitar a árvore de acordo com a numeração representada ao lado. Esta última exposição segue a sintaxe da linguagem *karva*.

A Fig. C.4 representa agora, na simbologia de árvore, um exemplo de *crossover* aplicado aos dois indivíduos apresentados na Tab. C.2.

As mutações podem ser efectuadas, por exemplo, através da troca de uma das operações matemáticas através da escolha aleatória da operação a substituir e da substituta.

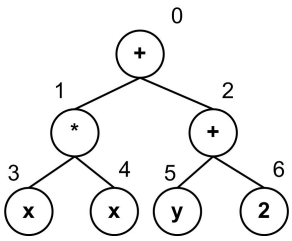
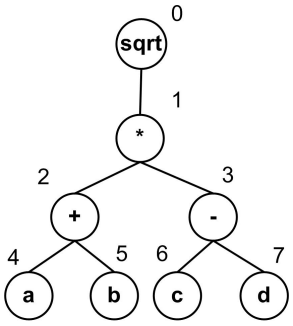
| | | |
|---|---|---|
| a) | $x^2+y+2$ | $\sqrt{(a+b)*(c-d)}$ |
| b) | ( + ( * x x ) ( + y 2 ) ) | (sqrt (* (+ a b) (- c d) ) ) |
| c) |  |  |
| d) | 0 1 2 3 4 5 6<br>+ * + x x y 2 | 0 1 2 3 4 5 6 7<br>sqrt * + - a b c d |

Tabela C.2: Exemplo de três codificações possíveis dos indivíduos num algoritmo de previsão de séries temporais através de algoritmos genéticos.

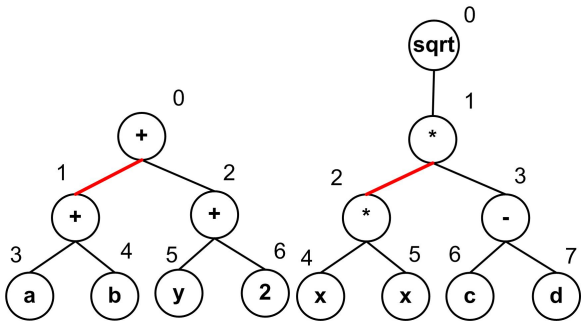



Figura C.4: Exemplificação do operador de *crossover* aplicado aos indivíduos apresentados na Tab. C.2, cuja resultados foram dois novos indivíduos.

Normalmente, nos algoritmos de programação genética simbólica, indica-se o conjunto de funções e o número de parâmetros que aceitam. Como exemplo, para o caso apresentado, ter-se-ia:

$$\{(+,2)(-,2)(*,2)(/,2)(sqrt,1)\}$$

Estes algoritmos são, normalmente, lentos, devendo ser utilizados em último recurso, sempre mais numa perspectiva de retaguarda.

### C.1.4 Principais Métodos de Análise Estudados

Nesta secção apresentam-se os principais métodos para análise de séries temporais com ênfase para os modelos MA, AR, ARMA, ARIMA e alisamento exponencial.

Intuitivamente, o método mais adequado a utilizar depende das características dos dados. Por exemplo, está demonstrado que o melhor estimador para um conjunto de dados aleatórios é a média (minimiza a média do quadrado dos erros - **MSE**). No entanto, este estimador já não é adequado quando a série apresenta tendências.

De seguida, apresentam-se os métodos estudados, implementados e a teoria matemática de suporte.

#### C.1.4.1 Estimador de Médias Móveis

O estimador de médias móveis (MA-*Moving Average*) [108] utiliza os últimos $N$ valores da série para calcular uma média, isto é, o estimador. Matematicamente tem-se:

$$\hat{y}[k] = \frac{1}{N}\sum_{i=0}^{N-1} y[k-i] \tag{C.16}$$

Facilmente se demonstra que:

$$\hat{y}[k+1] = \frac{1}{N}\left(y[k+1] - y[k-N+1]\right) + \hat{y}[k],\ N \geq 2 \tag{C.17}$$

o que permite uma implementação em computador com o mesmo número de operações, independente do valor de $N$.

Quando os valores da série são obtidos a intervalos irregulares, o estimador de médias móveis, deverá ser modificado levando em consideração os espaçamentos entre as amostras:

$$\hat{y}[k] = \frac{\sum_{i=0}^{N-1} y[k-i].\Delta t[k-i]}{\sum_{i=0}^{N-1} \Delta t[k-i]} \tag{C.18}$$

A implementação em computador é independente do valor de N pois, após pequenas modificações, obtém-se:

$$\begin{cases} sum_y[k] = \sum_{i=0}^{N-1} y[k-i].\Delta t[k-i] \\ \\ sum_\Delta t[k] = \sum_{i=0}^{N-1} \Delta t[k-i] \\ \\ \hat{y}[k+1] = \frac{y[k+1]+sum_y[k]-y[k-N+1]}{\Delta t[k+1]+sum_\Delta t[k]+\Delta t[k+N-1]} \end{cases} \tag{C.19}$$

Tal como foi descrito na secção anterior, o estimador de médias móveis deve ser utilizado em séries aleatórias e funciona razoavelmente quando a série evidencia crescimento/decrescimento.

**Algoritmo C.1** *Médias Móveis - MAS (sigla usada para referenciar os resultados)*

***N***=$n^o$ *de amostras a utilizar* ($N \geq 2$)
*Iniciar vector[i]=0, i=1,...,N*
*Iniciar soma=0*
***Para*** *casa amostra*
    *adquirir* $y[k]$
    $soma = soma + y[k]$ − *amostra mais antiga no vector*
    *guarda no vector[i] a amostra*
    $\hat{y}[k] =$ *média da soma*
    $\hat{y}[k+m] = \hat{y}[k] + m.(y[k] - \hat{y}[k-1])$ $\Leftrightarrow$ *Estimativa de valores futuros*
***Fim para***

O algoritmo das médias móveis pode ainda ser modificado de forma a pesar cada uma das amostras de acordo com a indicação de um vector de pesos, concretamente:

$$\hat{y}[k] = \frac{\sum_{i=0}^{N-1} y[k-i].w[i]}{\sum_{i=0}^{N-1} w[i]} \; e, \; normalmente, \; \sum_{i=0}^{N-1} w[i] = 1 \tag{C.20}$$

A Eq. C.20 é bastante semelhante à Eq. C.18, excepto no facto do somatório dos pesos não ser unitário. Para uma série aperiódica, incluí-se na equação a fórmula das diferenças temporais, tal como referido anteriormente, substituindo $w[i]$ por $w[i].\Delta t[k-i]$ na Eq. C.20.

### C.1.4.2 Estimador de Alisamento Exponencial

O alisamento exponencial (*Exponential Smoothing*) [108] é um estimador facilmente utilizável em algoritmos numéricos pois envolve poucas operações matemáticas e aparece sob a forma recursiva:

$$\begin{aligned} \hat{y}[k] &= \alpha.y[k] + (1-\alpha).\hat{y}[k-1] \quad [2] \\ \Leftrightarrow \hat{y}[k] &= \hat{y}[k-1] + \alpha.(y[k] - \hat{y}[k-1]) \end{aligned} \tag{C.21}$$

onde $\hat{y}[1] = y[1]$, sendo $0 < \alpha \leq 1$ um parâmetro a indicar. Se $\alpha$ for próximo de 1 o estimador seguirá mais rapidamente as novas amostras, se for próximo de zero, as estimativas mais antigas

[2] Considerou-se aqui uma das hipóteses do alisamento exponencial, dado que, normalmente, a Eq. C.21 também pode ser escrita como $\hat{y}[k+1] = \alpha.y[k] + (1-\alpha).\hat{y}[k]$, residindo a diferença apenas nos índices de $\hat{y}[k]$. Como critério

terão um peso superior na estimativa actual. A escolha de $\alpha$ é um parâmetro fornecido ou, em alternativa, deve ser estimado em função de um dos critérios indicados no capítulo 6, secção 6.2.3.1 (pág. 98 ). Uma conclusão evidente, $\hat{y}[k-1]$ é a estimativa para $y[k]$ e $\hat{y}[k]$ para $y[k+1]$ e, aparentemente, se o erro da estimativa for quase nulo numa determinada iteração, temos $\hat{y}[k] = \hat{y}[k-1]$, i.e., a próxima estimativa é igual à estimativa anterior. Por absurdo, o estimador, sistematicamente, não pode prever correctamente o próximo valor pois se assim acontecer o erro será nulo e, no limite, se numa iteração o erro for nulo, na seguinte o erro aumentará consideravelmente (no caso de séries não constantes). O nome de alisamento exponencial advém da particularidade de, ao expandir a Eq. C.21, obter-se uma dependência exponencial ao longo das amostras:

$$\begin{aligned}
\hat{y}[2] &= \alpha.y[2] + (1-\alpha).y[1] \\
\hat{y}[3] &= \alpha.y[3] + \alpha.(1-\alpha).y[2] + (1-\alpha)^2.y[1] \\
\hat{y}[4] &= \alpha.y[4] + \alpha.(1-\alpha).y[3] + \alpha.(1-\alpha)^2.y[2] + (1-\alpha)^3.y[1] \\
&\cdots \\
\hat{y}[k] &= (1-\alpha)^{k-1}.y[1] + \alpha.\textstyle\sum_{n=2}^{k}(1-\alpha)^{k-n}.y[n]
\end{aligned} \tag{C.22}$$

A Eq. C.22 mostra a propriedade de alisamento exponencial, sendo que todos os pesos somados à medida que $k \to +\infty$ é igual a 1. O valor de $\alpha$ na implementação elaborada neste trabalho deve ser escolhido de forma a minimizar o MSE, cujo valor pode ser calculado através de métodos iterativos ou de optimização não linear, sendo um dos métodos apontados em várias referências consultadas, o método de *Levenberg-Marquardt*. Na realidade, pretende-se obter o valor de $\alpha$ que minimiza o MSE, ou seja:

$$\begin{aligned}
min\{\bar{F}(\alpha,k)\} &= min\left\{\tfrac{1}{k}\textstyle\sum_{i=1}^{k}\left(y[i+1]-\hat{y}[i]\right)^2\right\} = \\
&= min\left\{\tfrac{1}{k}\textstyle\sum_{i=1}^{k}\left(y[i+1]-\alpha.\sum_{n=2}^{i}(i-\alpha)^{i-n}.y[n]-(1-\alpha)^{i-1}.y[1]\right)^2\right\}, \quad [3]
\end{aligned}$$

que, claramente, é um problema de optimização não linear.

O valor de $\alpha$, calculado de acordo com o método de *Levenberg-Marquardt* [132, 133], deverá, em princípio, ser utilizado para recalcular todas as amostras da série, tal como a fórmula indica. Numa situação real tal procedimento não é praticável. Uma opção passa pelo recalculo para o intervalo das últimas **n** amostras em detrimento do cálculo desde $k = 1$ até ao instante actual.

---

adoptado neste texto, tal como já referido na secção 6.2.3.1 (pág. 98), o índice k em $\hat{y}[k]$ representa o instante de cálculo e não o instante temporal que esta estimativa pretende aproximar. Tanto uma como outra são, no entanto, notações normais em alguns autores.

[3] Comando maple: solve(diff(1/i*sum(y[n+1]-x*sum((1-x)^ (n-j)*y[j],j=2..n)+(1-x)^ (n-1)*y[1],n=2..k),x),x), cuja solução é demasiado complexa não resolúvel pelo programa.

**Algoritmo C.2** *Alisamento exponencial simples, preferencialmente executar em servidores, ESAA (sigla usada para referenciar os resultados)*

***N***= *n º de amostras a utilizar para calcular o erro quadrático (N>0)*
$k \leftarrow 1$
$\alpha \leftarrow 0.5$
*adquirir* $y[1]$
$\hat{y}[k] \leftarrow y[1]$
***Para*** *cada amostra*
    *adquirir* $y[k]$
    $\hat{y}[k] = \alpha.y[k] + (1-\alpha).\hat{y}[k-1]$
    *Calcular* $\alpha$ *de acordo com o método de Levenberg-Marquardt*
    *Restringe* $\alpha$ *de modo a* $0 < \alpha \leq 1$
    ***Para*** *os últimos N valores*
        *re-calcular* $\hat{y}[k]$
    ***Fim*** *Para*
    $\hat{y}[k+m] = \hat{y}[k] + m.\alpha.(y[k] - \hat{y}[k-1]) \Leftrightarrow$ *Estimativa de valores futuros*
***Fim para***

Introduzindo uma aproximação de primeira ordem para encontrar uma primeira estimativa de $\alpha$, alterando a Eq. C.21 para $\hat{y}[k] = \alpha.y[k] + (1-\alpha).y[k-1]$, e/ou, em alternativa, considerando que $\hat{y}[i-1]$ é constante, tem-se:

$$min\{\bar{F}(\alpha,k)\} = min\left\{\frac{1}{k}\sum_{i=1}^{k}(y[i+1] - \hat{y}[i])^2\right\}$$

ao expandir $\hat{y}[i] = \alpha.y[i] + (1-\alpha).\hat{y}[i-1]$ há a necessidade de alterar o índice inferior do somatório para dois, dado considerar-se que a série se inicia no índice um, isto é,

$$min\left\{\frac{1}{k}\sum_{i=2}^{k}(y[i+1] - \alpha.y[i] - (1-\alpha).\hat{y}[i-1])^2\right\}$$

que corresponde ao problema de determinar $\alpha$ onde a sua derivada é nula, o que não oferece problemas devido a $F$ ser semi-definida positiva.

$$\frac{dF(\alpha)}{d\alpha} = 0 \Leftrightarrow 0 = -\frac{1}{k}.2.\sum_{i=2}^{k}(y[i+1] - \alpha.(y[i] - \hat{y}[i-1]) - \hat{y}[i-1]).(y[i] - \hat{y}[i-1]) \Leftrightarrow$$

$$\alpha = \frac{\sum_{i=n}^{k}(y[i+1] - \hat{y}[i-1]).(y[i] - \hat{y}[i-1])}{\sum_{i=n}^{k}(y[i] - \hat{y}[i-1])^2},\ n \geq 2 \qquad \text{(C.23)}$$

A Eq. C.23[4] deve verificar a restrição de $0 < \alpha \leq 1$[5].

Este valor e o valor obtido, substituindo $\hat{y}[i-1] = y[i-1]$ na Eq. C.23, serão uma primeira estimativa, podendo, inclusive, ser utilizada a média dos dois valores, cujo valor poderia ser refinado com um método de pesquisa não linear.

Pode-se ainda aproximar mais a estimativa, considerando uma aproximação de segundo grau, nomeadamente:

[4]Comando maple: simplify(solve(diff(1/k*sum((y[n+1]-x*y[n]-(1-x)*y[n-1])^ 2,n=2..k),x),x),x)
[5]Nota: esta equação é uma estimativa de $\alpha$, a qual foi obtida após a introdução clara de um erro inicial

$$min\left\{\frac{1}{k}\sum_{i=2}^{k}(y[i+1]-\hat{y}[i])^2\right\} =$$
$$min\left\{\frac{1}{k}\sum_{i=2}^{k}(y[i+1]-\alpha.y[i]-(1-\alpha).(\alpha.y[i-1]-(1-\alpha).\hat{y}[i-2]))^2\right\}$$

o que é equivalente a

$$\begin{cases} M[k] = y[k-1]+\hat{y}[k-2] \\ N[k] = y[k]+y[k-1]+2.\hat{y}[k-2] \\ min\left\{\frac{1}{k}\sum_{i=2}^{k}\left(y[i+1]-\alpha.N[i]+\alpha^2.M[i]-\hat{y}[i-2]\right)^2\right\} \end{cases} \tag{C.24}$$

cuja solução[6] é demasiado complexa para ser descrita neste texto. Se uma solução baseada no método do alisamento exponencial for implementada num sistema de *hardware* para detecção *on-line*, conforme é o objectivo da presente dissertação, a Eq. C.23 pode ser utilizada para "sintonizar" o valor de $\alpha$, dado que não será possível, após cada amostra, utilizando o método de optimização não linear de *Levenberg-Marquardt*, recalcular a sequência de $\hat{y}[k]$ desde o primeiro instante, tal como a Eq. C.22 sugere. Apresenta-se de seguida um algoritmo que permitirá, com este método, "sintonizar" o valor de $\alpha$ à medida que as amostras são adquiridas.

**Algoritmo C.3** *Alisamento exponencial simples, modificado para aplicação em micro-controladores, ESMSE (sigla utilizada nos resultados para referenciar este método)*

***N*** *= n º de amostras a utilizar para calcular o erro quadrático (0 infinito)*
*Soma[1]=Soma[2]=Soma[3]=Soma[4]=0*
*Inicializa vector Amostras=0, com 3 amostras*
***Se*** $N > 0$ *então*
    *Vectores rotativos =0, com L amostras, onde* $L = máx\{3, N\}$
***Fim Se***
$k \leftarrow 1$
*adquirir* $y[1]$
$\hat{y}[k] \leftarrow y[1]$
$\alpha \leftarrow 0.5$
***Para*** *cada amostra*
    *adquirir* $y[k]$ *e guardar no vector de amostras*
    $\hat{y}[k] = \alpha.y[k]+(1-\alpha).\hat{y}[k-1]$
    ***Se*** $N > 0$ *então*
        *Actualizar somas com a nova amostra esquecendo a N-ésima amostra*
    ***Se não***
        *Actualizar somas com a nova amostra*
    ***Fim Se***
    *Calcular* $\alpha$ *de acordo com a Eq. C.23 (média, usando* $\hat{y}[i-1]$ *e de seguida* $y[i-1]$*)*
    ***Restringe*** $\alpha$ ***de modo a*** $0 < \alpha \leq 1$
    $\hat{y}[k+m] = \hat{y}[k]+m.\alpha.(y[k]-\hat{y}[k-1]) \Leftrightarrow$ *Estimativa de valores futuros*
***Fim para***

[6] No maple usar o comando: solve(diff(1/k*sum((y[n+1]+xˆ 2*M[n]-x*N[n]+y[n-2] )ˆ 2,n=2..k),x),x)

O método do alisamento exponencial pode ainda ser modificado de acordo com um princípio semelhante ao indicado anteriormente, de modo a adaptar o valor de $\alpha$ ao longo do tempo. Este método é denominado de alisamento exponencial com taxa de resposta adaptativa, sendo agora apenas necessário indicar o valor inicial de $\alpha$, que será adaptado ao longo das sucessivas amostras, sendo, no entanto, o parâmetro $\beta$ fixo (0.2 é uma boa escolha):

$$\begin{cases} \hat{y}[k] = \alpha[k].y[k] + (1-\alpha[k]).\hat{y}[k-1] \\ e[k] = y[k] - \hat{y}[k-1] \\ E[k] = \beta.e[k] + (1-\beta).E[k-1] \\ M[k] = \beta.|e[k]| + (1-\beta).M[k-1] \\ \alpha[k+1] = \left|\frac{E[k]}{M[k]}\right| \end{cases}$$

**Algoritmo C.4** *Alisamento exponencial simples com taxa de resposta adaptativa para micro-controladores, ARRSES (sigla utilizada nos resultados para referenciar este método)*

$\alpha \leftarrow 0.5$
$\beta \leftarrow 0.5$
***Para*** *cada amostra*
    *adquirir* $y[k]$ *e guardar no vector de amostras*
    $\hat{y}[k] = \alpha.y[k] + (1-\alpha).\hat{y}[k-1]$
    $erro = y[k] - \hat{y}[k-1]$
    $E[k] = \beta.erro + (1-\beta).E[k-1]$
    $M[k] = \beta.|erro| + (1-\beta).M[k-1]$
    $\alpha = \left|\frac{E[k]}{M[k]}\right|$
    $\hat{y}[k+m] = \hat{y}[k] + m.erro \Leftrightarrow Estimativa\ de\ valores\ futuros$
***Fim para***

Quando os intervalos das amostras da série são aperiódicos (considerando $\Delta t[1] = \Delta t$), tem que se alterar a equação básica do estimador de alisamento exponencial e, tendo como base, sem perda de generalidade, a Fig. C.5, pode-se deduzir que:

$$\hat{z}[4] = \alpha.z[4] + \alpha.(1-\alpha).z[3] + \alpha.(1-\alpha)^2.z[2] + (1-\alpha)^3.z[1] \ (da\ Eq.\ C.22)$$
$$\hat{y}[2] = \alpha.y[2] + (1-\alpha).y[1] = \alpha.z[4] + (1-\alpha).z[1] \neq \hat{z}[4] \ (da\ Eq.\ C.22)$$
$$\Delta t[2] = t_2 - t_1 = T_4 - T_1 = 3.\Delta t$$

e concluí-se das equações indicadas acima que a previsão de $z[k]$ e $y[k]$ serão diferentes se se aplicar simplesmente a equação do alisamento exponencial a $y[k]$ sem levar em consideração o facto de $y[k]$ ser aperiódica. Facilmente se verifica que na previsão de $y[2]$, o termo dependente de $(1-\alpha)$ deveria estar elevado à potência de 3, exactamente $\frac{\Delta t[2]}{\Delta t} = \frac{t_2 - t_1}{\Delta t} = \frac{T_4 - T_1}{\Delta t} = 3$, ficando ainda a carecer do termo relativo às amostras em falta, $\alpha.(1-\alpha).z[3] + \alpha.(1-\alpha)^2.z[2]$, para as duas expressões serem idênticas. Tomando, por suposição, que o valor de $\hat{z}[4]$ é o mais correcto, $\hat{y}[2]$ necessita de contabilizar ainda alguns termos para estar o mais próximo possível de $\hat{z}[4]$. Uma hipótese é efectuar uma interpolação linear entre $y[1]$ e $y[2]$, recolhendo por aproximação valores

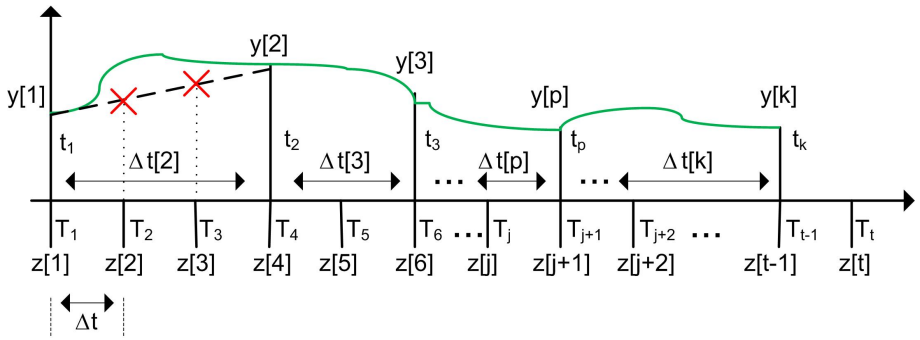



Figura C.5: Relação entre duas séries, uma aperiódica e outra periódica para dedução da fórmula do alisamento exponencial aperiódico.

admissíveis para $z[3]$ e $z[4]$ e contabilizando-os na estimativa de $\hat{y}[2]$. Outra hipótese é "esquecer" o número de amostras que se perderam e ponderar os valores em função dos respectivos intervalos temporais para tornar o processo de cálculo tão simples como o próprio processo de alisamento exponencial indica.

As duas hipóteses formuladas podem então ser descritas matematicamente pelas seguintes equações:

$$\begin{cases} \hat{y}[k] = \alpha.y[k] + A + (1-\alpha)\frac{\Delta t[k]}{\Delta t}.\hat{y}[k-1] \\ \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad , \; hipótese\ 1 \\ A = \sum_{i=1}^{int\left(\frac{\Delta t[k]}{\Delta t}\right)-1} \left(y[k] - \frac{y[k]-y[k-1]}{\Delta t[k]}.i.\Delta t\right).\alpha.(1-\alpha)^i \end{cases} \tag{C.25}$$

$$\hat{y}[k] = \alpha.y[k]\frac{\Delta t[k]}{\Delta t} + (1-\alpha)^{\frac{\Delta t[k]}{\Delta t}}.\hat{y}[k-1], \; hipótese\ 2$$

A equação associada à *hipótese 1* é de importância primordial, para o método de previsão baseado em redes neuronais, dado ser necessário *alimentar* a rede com amostras igualmente espaçadas, tal como será apresentado na secção C.1.4.5.

### C.1.4.3 Estimador de Alisamento Exponencial Duplo

O estimador de alisamento exponencial (*Double Exponential Smoothing*) [108] apresentado na secção anterior é também conhecido na literatura inglesa como *Single Exponencial Smoothing*, alisamento exponencial simples. Como características principais, não consegue estimar uma sequência de dados crescente/decrescente em termos de previsão de *m* valores no futuro.

Assim, para ultrapassar este problema, à equação básica do alisamento exponencial adiciona-se uma segunda equação, que leva em conta o erro entre estimativas consecutivas, que depois é re-alimentado na equação principal, isto é:

$$\begin{aligned} \hat{y}[k] &= \alpha.y[k] + (1-\alpha).(\hat{y}[k-1] + b[k-1]) \\ b[k] &= \gamma.(\hat{y}[k] - \hat{y}[k-1]) + (1-\gamma).b[k-1] \end{aligned} \tag{C.26}$$

sendo agora necessário indicar duas constantes $0 < \alpha, \gamma \leq 1$. A inicialização do estimador é importante, sendo, como no caso anterior, $\hat{y}[1] = y[1]$ e, para a nova sequência, $b[1] = 0$. A sigla HW será utilizada para referenciar o método descrito pelas Eq. C.26.

Reescrevendo as equações C.26 obtém-se:

$$\begin{aligned} \hat{y}[k] &= \hat{y}[k-1] + b[k-1] + \alpha.(y[k] - \hat{y}[k-1] - b[k-1]) \\ b[k] &= b[k-1] + \gamma.(\hat{y}[k] - \hat{y}[k-1] - b[k-1]) \end{aligned} \tag{C.27}$$

Analisando as equações obtidas facilmente se concluí uma melhoria significativa relativamente ao alisamento exponencial simples, nomeadamente a nível da estimativa do erro, dado não ser apenas função da diferença entre o valor medido e a sua previsão, o que permite melhorar significativamente a previsão do próximo valor.

**Algoritmo C.5** *Algoritmo do alisamento exponencial duplo, HW*

$\alpha \leftarrow 0.2$
$\gamma \leftarrow 0.2$
$b[1] \leftarrow 0$
$\hat{y}[1] \leftarrow y[1]$
***Para*** *cada amostra*
    *adquirir* $y[k]$ *e guardar no vector de amostras*
    $\hat{y}[k] = \alpha.y[k] + (1-\alpha).(\hat{y}[k-1] + b[k-1])$
    $b[k] = \gamma.(\hat{y}[k] - \hat{y}[k-1]) + (1-\gamma).b[k-1]$
    $\hat{y}[k+m] = \hat{y}[k] + m.b[k] \Leftrightarrow Estimativa\ de\ valores\ futuros$
***Fim para***

### C.1.4.4 Estimador de Alisamento Exponencial triplo

Os estimadores de alisamento exponencial apresentados até ao momento, nomeadamente, alisamento exponencial simples com taxa de resposta adaptativa e alisamento exponencial duplo, adaptam-se correctamente a uma sequência de dados aleatórios e com tendências de crescimento. Não funcionam, no entanto, correctamente com periodicidades, designadamente sazonalidades aditiva com período **sz**, assim, a Eq. C.28 apresenta as equações deste modelo [108], onde $\alpha, \beta, \gamma \in [0,1]$. As inicializações, mais uma vez, são importantíssimas e podem ser dadas por $b[1] = y[2] - y[1]$, ou, em alternativa, $b[1] = (y[sz] - y[1])/(sz-1)$, para $sz > 1$. Este algoritmo será designado por HWSas na secção dos resultados.

$$\begin{cases} \hat{y}[k] = \alpha.(y[k] + s[k-sz]) + (1-\alpha).(\hat{y}[k-1] + b[k-1]) \\ b[k] = \beta.(\hat{y}[k] - \hat{y}[k-1]) + (1-\beta).b[k-1] \\ s[k] = \gamma.(y[k] - \hat{y}[k]) + (1-\gamma).s[k-sz] \\ \hat{y}[k+m] = \hat{y}[k] + m.b[k] + s[k+m-sz] \end{cases} \tag{C.28}$$

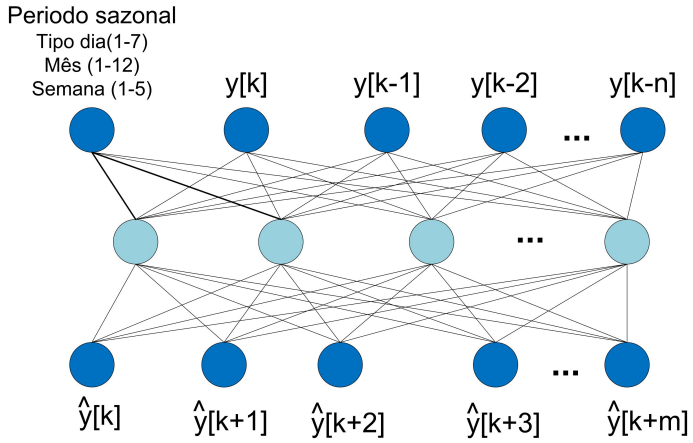



Figura C.6: Rede *feedforward* de 3 camadas usada para prever $m+1$ valores no futuro.

**Algoritmo C.6** *Algoritmo do alisamento exponencial tripo, HWSas*

$\alpha \leftarrow 0.5$
$\beta \leftarrow 0.2$
$\gamma \leftarrow 0.2$
$b[1] \leftarrow (y[sz]-y[1])/(sz-1)$
$\hat{y}[1] \leftarrow y[1]$
***Para*** *cada amostra*
    *adquirir* $y[k]$ *e guardar no vector de amostras*
    $\hat{y}[k] = \alpha.(y[k]+s[k-sz])+(1-\alpha).(\hat{y}[k-1]+b[k-1])$
    $b[k] = \beta.(\hat{y}[k]-\hat{y}[k-1])+(1-\beta).b[k-1]$
    $s[k] = \gamma.(y[k]-\hat{y}[k])+(1-\gamma).s[k-sz]$
    $\hat{y}[k+m] = \hat{y}[k]+m.b[k]+s[k+m-sz] \Leftrightarrow$ *Estimativa de valores futuros*
***Fim para***

### C.1.4.5 Estimação com redes neuronais *feedforward*

Utilizando o modelo da rede neuronal [111, 112, 113] *feedforward*, realimentaram-se os vários valores da série de acordo com uma janela pré-estabelecida de comprimento *n*, sendo a saída da rede constituída por *m* neurónios em correspondência com o número de passos no futuro a estimar.

A rede utilizada para previsão está indicada na Fig. C.6 e contempla um neurónio para indicação de uma métrica adequada à medida da sazonalidade que, evidentemente, terá de ser estimada por outro processo. No entanto, caso não seja possível antever essa medida, à rede indicada podem ser adicionados mais neurónios, cujos valores podem ser o tipo do dia da semana, o número do mês ou o número da semana dentro do mês ou dentro do ano, etc. Desta forma, através do algoritmo de aprendizagem *backpropagation*, a rede neuronal adaptará os pesos das várias *sinapses*. A função

de activação utilizada nas camadas de entrada e escondida é a função sigmoíde e, na camada da saída, função linear. Apesar desta topologia neuronal ser mais adequada para sinais estáticos, essa limitação é ultrapassada através da inclusão de atrasos temporais inseridos na camada de entrada, tal como se visualiza na Fig. C.6. Este método incorpora um parâmetro, que indica qual o comprimento da série a usar no treino (janela a considerar). Após cada amostra a rede será re-treinada com o mesmo número de amostras, deslizando uma amostra à frente. De seguida, serão previstos os valores da série **m** passos à frente. Este processo está exemplificado no algoritmo C.7. Um dos problemas deste método é a necessidade de muitas amostras iniciais para elaborar o treino da rede.

**Algoritmo C.7** *Previsão com redes neuronais feedforward (RNFF, sigla utilizada nos resultados para referenciar este método)*

***N**= Tamanho da janela para treinar a rede*
*n= $n^o$ de neurónios na camada de entrada*
*m=$n^o$ de neurónios da camada de saída*
*ns=0 // ($n^o$ de neurónios de sazonalidade)*
*nNCE = valor pretendido ($n^o$ de neurónios na camada escondida)*
***Se** $N < 5.(n+m+2)$ então*
    *Termina, método não pode ser aplicado*
***Fim Se***
*Calcula matriz de entrada da rede com N neurónios deslizados $(n+1)$ a $(n+1)$*
*Calcula matriz de saída da rede com N neurónios deslizados $(m+1)$ a $(m+1)$*
*Treina a rede neural*
***Para** cada amostra*
    *Adquire y[k] e guarda valor*
    *Calcula matriz de entrada da rede*
    *Calcula matriz de saída da rede*
    *Re-treina a rede esquecendo a amostra mais antiga*
    *Efectua previsão dos $(m+1)$ valores $\Leftrightarrow$ Estimativa de valores futuros*
***Fim para***

### C.1.4.6 Estimação com a série de *Taylor*

Um dos métodos que pode ser utilizado para estimar valores futuros depende da fórmula da série de *Taylor*, Eq. C.15, truncada à n-ésima derivada [134]. Para aproximar a derivada utiliza-se a diferença de ordem *n* de acordo com a Eq. C.11. Assume-se, neste método, $n = 2$ e $\Delta t = 1$.

$$\hat{y}[k+m] = y[k] + \Delta y[k].(m+1).\Delta t + \frac{\Delta^2 y[k]}{2!} + Erro_aproximacão \Leftrightarrow$$
$$\Leftrightarrow \hat{y}[k+m] = y[k] + \Delta y[k].(m+1) + \frac{\Delta^2 y[k]}{2!} + Erro_aproximacão$$

Efectuar a estimação um passo à frente, equivale a assumir $m = 0$ e, consequentemente, a equação anterior reduz-se assim a:

$$\hat{y}[k] = y[k] + \Delta y[k] + \frac{\Delta^2 y[k]}{2!} + Erro_aproximacão$$
$$\hat{y}[k] - y[k] = \Delta y[k] + \frac{\Delta^2 y[k]}{2!} + Erro_aproximacão$$
$$\hat{y}[k] - y[k] = y[k] - y[k-1] + \frac{\Delta^2 y[k]}{2!} + Erro_aproximacão$$

Para cada uma das diferenças são calculadas duas séries, uma consistindo na média dos últimos três elementos e, a segunda, a um alisamento exponencial com $\alpha = 0.2$. O erro entre a estimativa e o valor real é ainda considerado para a correcção da próxima estimativa mas, de acordo com a sua série de alisamento exponencial e da média dos últimos três erros. O algoritmo C.8 mostra em detalhe os vários passos.

**Algoritmo C.8** *Previsão com base parcial na série de Taylor (Taylor, sigla associada a este método)*

$\alpha \leftarrow 0.2$
$\Delta y[0] = \Delta^2 y[0] = erro_real[0] = 0$
$\Delta y[1] = \Delta^2 y[1] = y[1]$
$erro_real[1] \leftarrow y[1]$
$\sum \Delta = \Delta y[1]$
$\sum \Delta^2 = \Delta^2 y[1]$
$\alpha_\Delta[1] = \Delta y[1]$
$\alpha_\Delta^2[1] = \Delta^2 y[1]$
$\sum erro = y[1]$
$\alpha_erro[1] = y[1]$
$k \leftarrow 2$
$\hat{y}[1] = y[1]$
***Para*** *cada amostra*
  $\Delta y[k] = y[k] - y[k-1]$
  $\Delta^2 y[k] = \Delta y[k] - \Delta y[k-1]$
  $\sum \Delta = \sum \Delta + \Delta y[k] - \Delta y[k-2]$
  $\sum \Delta^2 = \sum \Delta^2 + \Delta^2 y[k] - \Delta^2 y[k-2]$
  $\alpha_\Delta[k] = \alpha.\Delta[k] + (1-\alpha).\alpha_\Delta[k]$
  $\alpha_\Delta^2[k] = \alpha.\Delta^2[k] + (1-\alpha).\alpha_\Delta^2[k]$
  $erro_real[k] = y[k] - \hat{y}[k-1]$
  $\sum erro = \sum erro + erro_real[k] - erro_real[k-2]$
  $\alpha_erro[k] = \alpha.erro_real[k] + (1-\alpha).\alpha_erro[k]$
  $\hat{y}[k] = \frac{\sum \Delta}{3}.a + \alpha_\Delta[k].b + \frac{(\sum \Delta^2)^2}{3*2!}.c + \frac{(\alpha_\Delta^2[k])^2}{2!}.d + \frac{\sum erro}{3}.e + \alpha_erro[k].f$
***Fim para***

## C.2 Cooperação e integração de vários métodos

A organização e interacção humana têm muitas vezes inspirado os investigadores na resolução de vários problemas que se lhes deparam [135]. Encontram-se inúmeros exemplos ao longo dos

tempos, designadamente, no caso da robótica humanóide onde as técnicas utilizadas foram inspiradas na forma de andar do homem, no caso dos algoritmos genéticos e na situação da indústria, onde o trabalho em equipa de diversos equipamentos industriais optimiza o desempenho global... etc. Na área de cooperação de equipas, a inteligência artificial, mais uma vez, apresenta algumas soluções interessantes, apesar de simples. A questão que se põe é a seguinte:

> "... haverá um método de previsão para séries temporais que resolva eficazmente a previsão **m** passos à frente para toda e qualquer série? Obviamente, será difícil, senão impossível, resolver eficazmente este problema, dado que "decifrar" o futuro é impossível..."

Admitindo o conhecimento do desempenho de várias técnicas de previsão de séries, todas aplicadas ao mesmo problema, pode-se combinar o resultado de cada uma num único resultado, em função do tempo e da variação do desempenho. Mais uma vez a própria humanidade mostra-nos de forma clara este aspecto, na nossa capacidade e desempenho no trabalho ao longo da vida. É um facto inequívoco, uma das leis da natureza.

A Programação Linear Genética (LPG) apresenta soluções para este problema [135].

O resultado final pode ser dado por uma combinação linear das previsões dos $h$ algoritmos, segundo uma fórmula ponderada, i.e.

$$\hat{y}_{equipa}[k] = \sum_{i=1}^{h} w_i.\hat{y}_{método_i}[k]$$

A metodologia para encontrar os pesos de ponderação pode seguir várias estratégias:

a) Média dos **h** algoritmos;

b) Pesagem através do erro, onde o valor de wi é função da métrica a minimizar (no caso deste texto, MSE), nomeadamente:

- $w_i = \frac{1}{e^{\beta.Métrica_erro(método_i)}}$ e de seguida os pesos são normalizados de modo à sua soma ser unitária através de $w_i = \| \frac{w_i}{\sum_{j=1}^{h} w_j} \|$, sendo $\beta$ um factor de escala positivo;

c) O melhor método é a solução dada pela equipa a cada instante;

d) Utilização de uma rede neuronal linear para determinar os pesos, treinada através dos $k$ passos, onde a cada instante, após conhecida a resolução do sistema linear, se volta a treinar a rede.

## C.3 Optimização na estimação de parâmetros das séries temporais

Tal como foi mostrado nas secções anteriores, existem vários métodos que necessitam da estimação de parâmetros, tais como, por exemplo, o do alisamento exponencial. Desta forma, nesta secção ir-se-ão descrever alguns dos métodos que podem ser utilizados para resolver o problema, dando ênfase ao facto de tornar esta estimação dinâmica, isto é, que o parâmetro estimado seja adaptado *on-line* de acordo com os dados.

### C.3.1 Método de optimização de *Levenberg-Marquardt*

O método de *Levenberg-Marquardt* [132, 133] é uma alternativa ao método de *Gauss-Newton* para determinar o mínimo de uma função $F(x)$ que é a soma dos quadrados de funções não lineares.

$$F(x) = \frac{1}{2}\sum_{i=1}^{m}[f_i(x)]^2 \tag{C.29}$$

Considerando $J_i(x)$ o jacobiano de $f_i(x)$, o método de *Levenberg-Marquardt* utiliza uma técnica de procura em direcção à solução $\vec{p}$ dada pela equação:

$$\left(J_i^T.J_i + \lambda_i.I\right).p_i = -J_i^T.f_i \tag{C.30}$$

onde os $\lambda_i \geq 0$ e I é a matriz identidade. O método tem uma propriedade interessante, dado que para um $\Delta$ relacionado com $\lambda_i$, sabendo que $\|\vec{p}\| \leq \Delta$ e, nestas condições, o vector $\vec{p}$ é a solução de optimização para o sub-problema de minimizar $\|J_i.\vec{p} + f_i\|^2/2$.

### C.3.2 Método dos mínimos quadrados

Genericamente, muitas das soluções podem utilizar o método dos mínimos quadrados[7] em $X$ e/ou $Y$.

Em geral, dado um sistema linear de equações:

$$A.\vec{x} = \vec{k} \tag{C.31}$$

onde $A$ é uma matriz de $m \times n$ elementos (m>n), $\vec{k}$ é um vector de $m$ elementos e $\vec{x}$ é um vector de $n$ incógnitas. A pseudo-inversa, ou solução dos mínimos quadrados, é dada por:

$$\vec{x} = \left(A^T.A\right)^{-1}.A^T.\vec{k} \tag{C.32}$$

## C.4 Conclusões

Este anexo teve por objectivo introduzir alguns dos conceitos estudados na análise de previsão em séries de dados, com ênfase nas séries temporais. Este estudo engloba conceitos científicos extremamente vastos e transversais a muitas áreas do saber, nomeadamente engenharia e finanças.

Na literatura existem inúmeros métodos, estritamente matemáticos, estatísticos, da área da optimização ou da área da inteligência artificial ou variantes com conceitos indicados nas várias opções apresentadas.

Este anexo apresenta alguns desses métodos e introduz duas contribuições, face ao existente na literatura da especialidade à data do seu desenvolvimento, bem como modificações noutros, cuja

[7] Do Inglês: *Regression Analysis*, normalmente *Rank Regression* significa *Least Square*

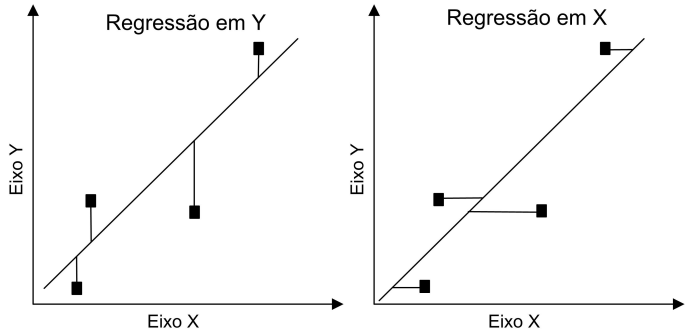



Figura C.7: Exemplificação gráfica do método dos mínimos quadrados, aplicado neste caso a uma recta. O método minimiza a distância em X, ou Y, conforme o caso.

aplicação é eficiente para a aplicação em sistemas embebidos[8], nomeadamente na secção C.1.4.2, Eq.C.23 e na secção C.1.4.6, algoritmo C.8. Esta área é fascinante, mas muito vasta e, por si só, daria um bom tema de doutoramento, designadamente, "desenvolvimento de técnicas de predição para sistemas embebidos".

Os resultados apresentados no capítulo 8 demonstram uma boa estabilidade, tendo, os algoritmos aqui modificados em algumas situações apresentado o melhor desempenho.

[8]Do Inglês, *Embedded systems*

# Anexo D

# Estudo de avarias em motores de indução

## D.1 Introdução

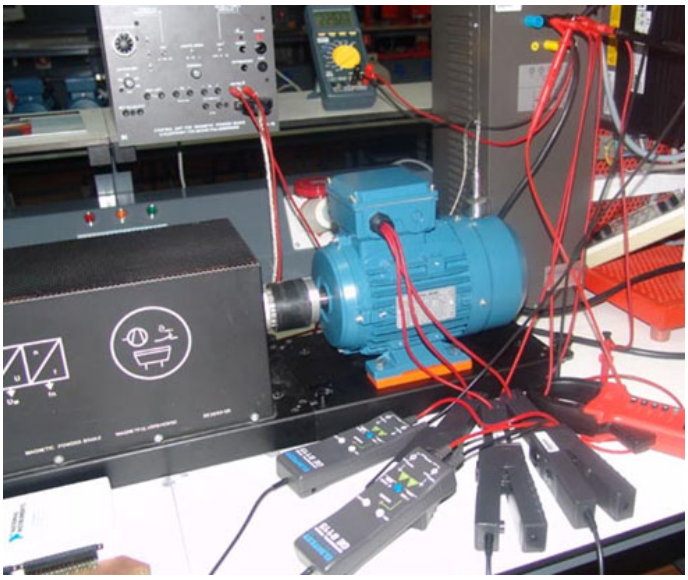
Este anexo apresenta muito resumidamente os testes de campo referentes à análise de frequência em dados adquiridos por um sensor de aceleração.

Para o efeito foram utilizados dois motores, um em bom estado e outro com uma avaria introduzida, tendo sido adquirido o sinal de vibração em três zonas de funcionamento distintas: em vazio, a meia carga e em carga nominal. A Fig. D.1 mostra o exemplo da experimentação feita e, apesar de terem sido adquiridas as correntes, não houve nenhum estudo relativo às mesmas devido a não se poder correlacionar o comportamento de um motor de indução com um gerador eléctrico de um aero-gerador. Em termos de vibração mecânica, constatam-se semelhanças.

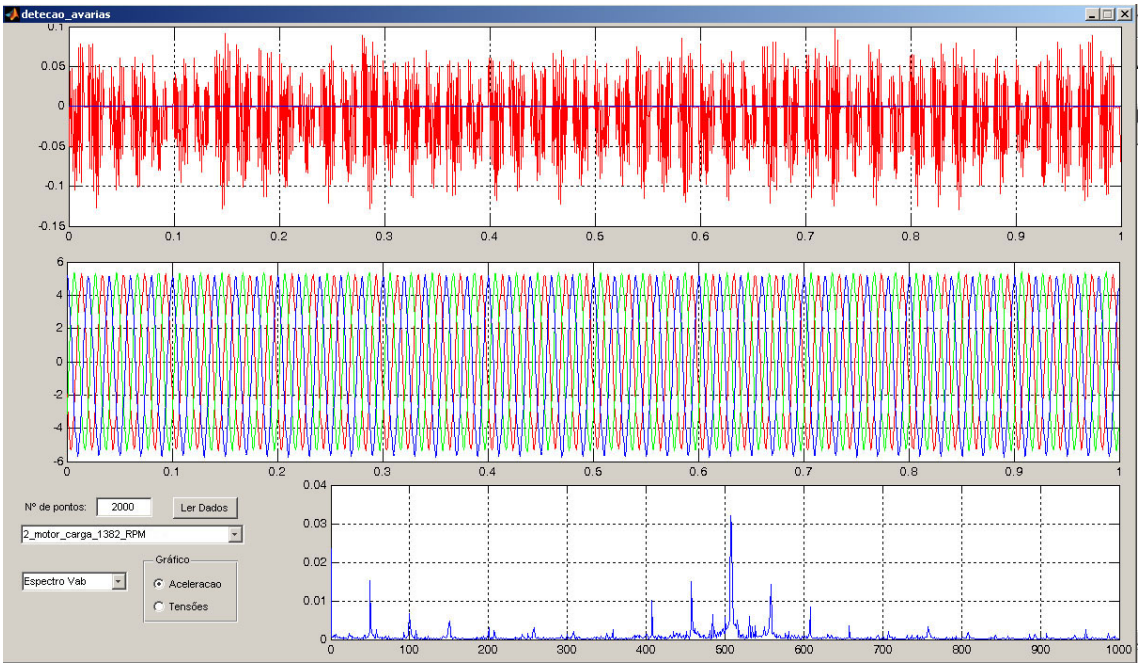
Os gráficos da Fig. D.2 representam o espectro FFT das vibrações recolhidas em cada um dos motores.

As referências [49, 136, 137, 138, 139, 140, 141, 142, 143, 144, 145, 146] sintetizam a globalidade dos métodos de análise de vibrações.

De acordo com o referido no capítulo 6, pág. 94, as frequências onde se devem analisar as amplitudes dependem da carga aplicada [105]. Como a perspectiva do trabalho realizado pressupunha a identificação do instante da avaria, a primeira ideia para concretizar este objectivo passou pela análise das frequências e, em seguida, da verificação temporal de qual a evolução de uma frequência em concreto, através da análise por séries temporais. Estabelecendo os valores limites máximos admissíveis para o crescimento da amplitude espectral, através das séries temporais, pode-se estimar o tempo necessário para que essa amplitude atinja o limiar. Nesta sequência, avançou-se para a construção de um simulador, onde fosse possível indicar quais as frequências que iriam "disparar", evoluindo de forma crescente na sua amplitude. A Fig. D.3 apresenta o simulador, no qual são programadas as frequências que devem evoluir de forma crescente em amplitude, bem como um exemplo tridimensional dessa mesma evolução conseguida através da simulação.

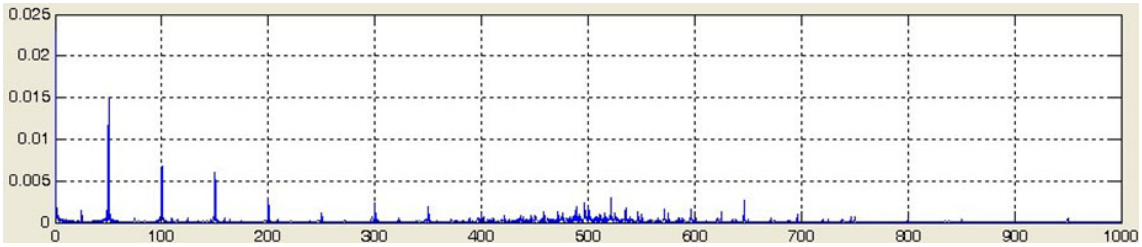


(a) Exemplo de ensaio com um motor trifásico de indução;

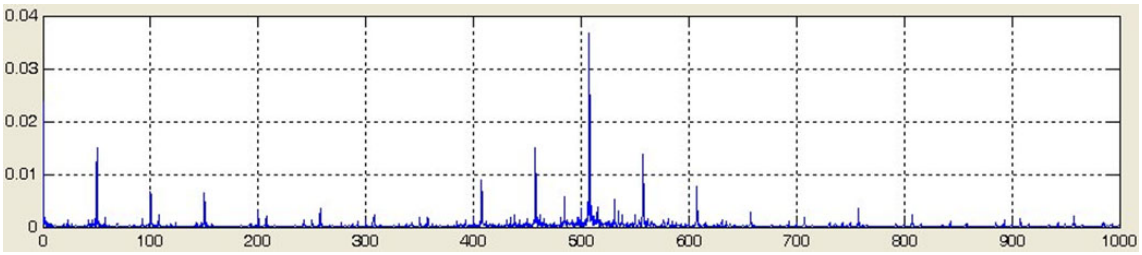



(b) Tratamento dos dados;

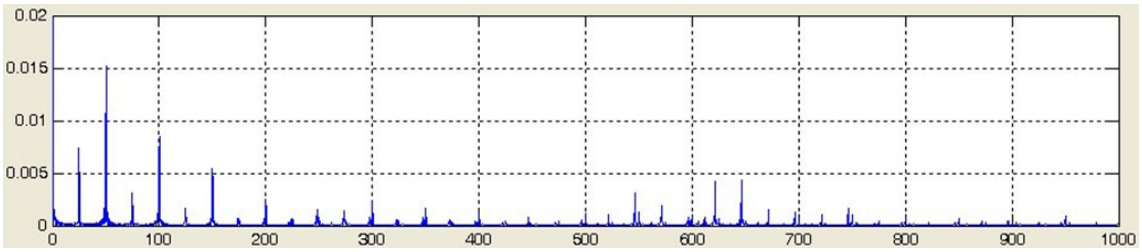
Figura D.1: Aquisição e tratamento.



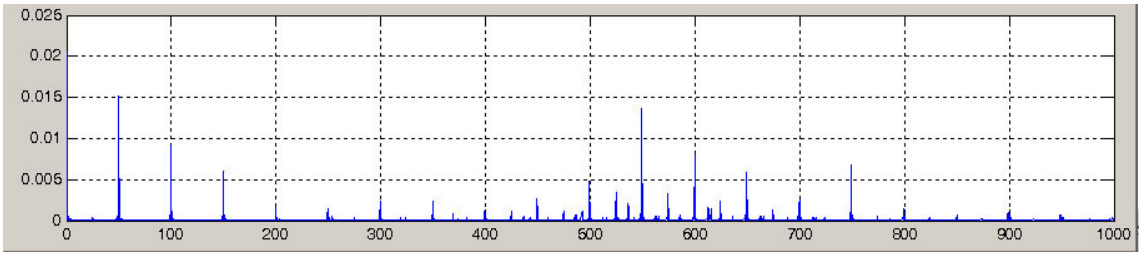

(a) Motor sem avaria. Em vazio;




(b) Motor sem avaria. Plena carga;

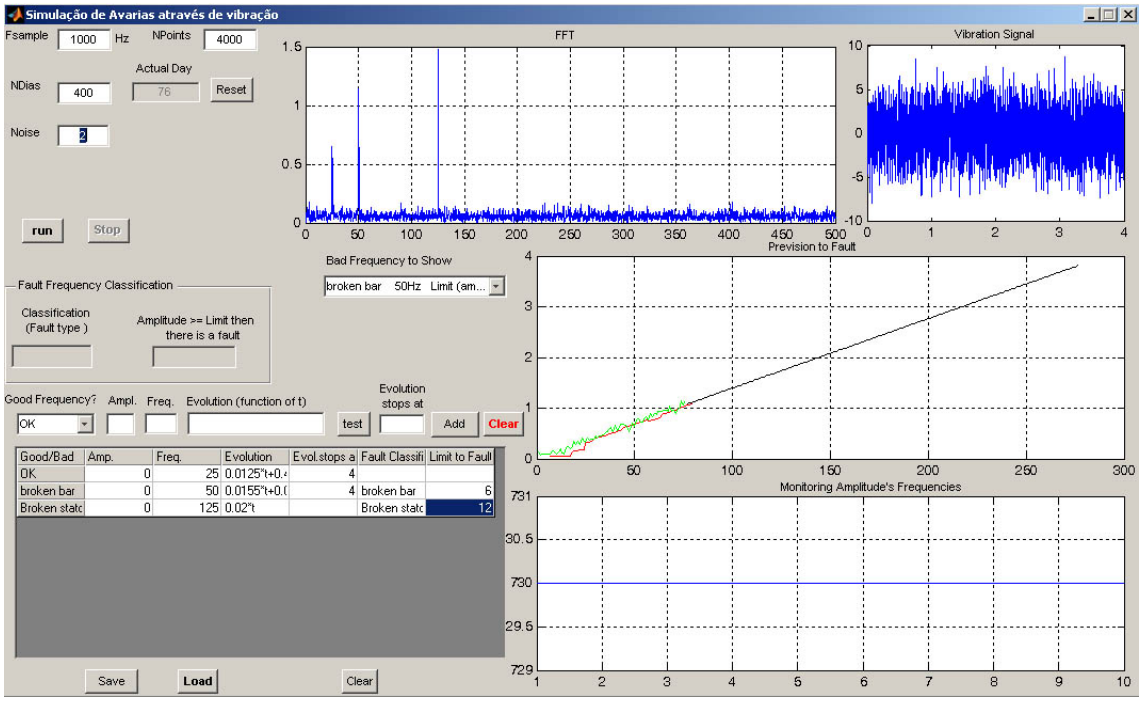



(c) Motor com avaria. Em vazio;




(d) Motor com avaria. 20% da carga;

Figura D.2: FFT do sinal de vibração de dois motores em função da carga aplicada.



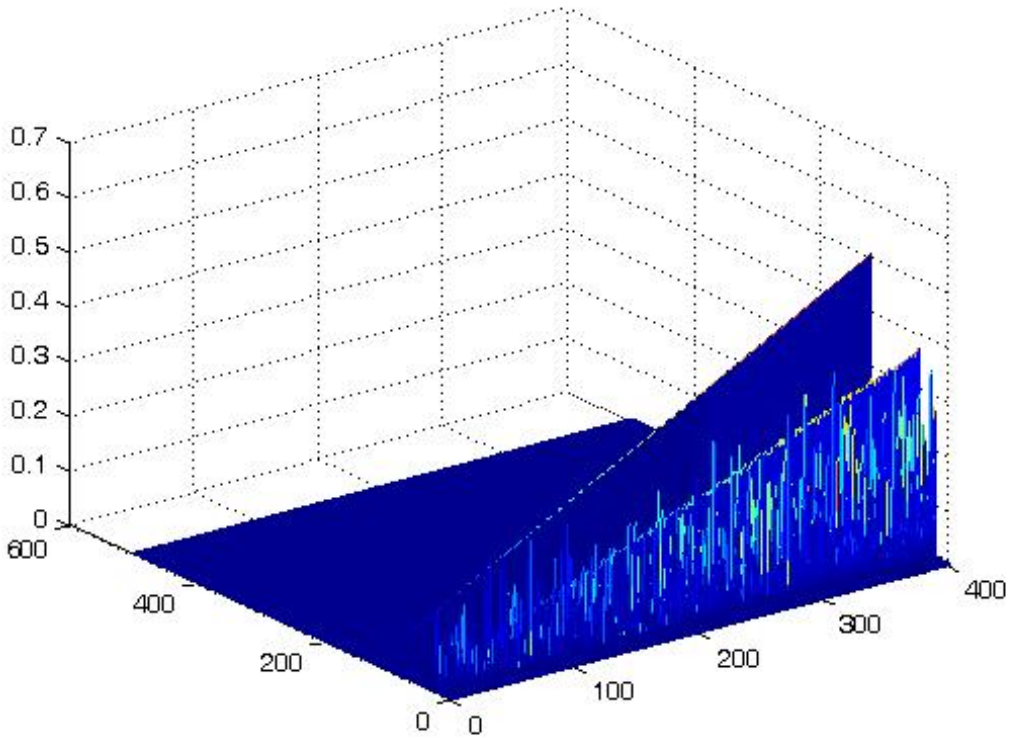




Figura D.3: Exemplo do simulador espectral.

A análise da evolução das frequências é efectuada através de algoritmos baseados em séries temporais, que são descritos no anexo C.

## D.2 Conclusões

Neste anexo apresentou-se uma metodologia para análise de vibrações de motores de indução trifásicos, em consequência da falta de oportunidade de efectuar este estudo no gerador de um aero-gerador. A analogia é em princípio semelhante, quando se utilizam geradores de indução trifásicos, com as respectivas diferenças, relativamente ao tamanho.

A outra metodologia apresentada, designadamente o seguimento temporal da evolução da amplitude de frequências chave foi validada em simulação, não tendo sido possível com os dados reais, em virtude de não existir informação suficiente ao longo do período temporal analisado. No entanto, do ponto de vista da simulação, a metodologia apresenta-se funcionalmente correcta.

# Referências


[1] Principle Power. Turbina eólica flutuante em alto mar, Agosto 2009. Disponível em http://www.principlepowerinc.com/products/windfloat.html e em http://wikienergia.com/~edp/index.php?title=EDP_e_Principle_Power_estudam_eólicas_offshore_em_Portugal, visitado a última vez em 21 de Agosto de 2009.

[2] Rui M.G.Castro. Introdução à energia eólica, energias renováveis e produção descentralizada. Relatório técnico, IST, Edição 4 de Março, 2009. Disponível em http://enerp4.ist.utl.pt/ruicastro/Pedagogia.htm, visitado a última vez em 21 de Agosto de 2009.

[3] Portal São Francisco. Energia eólica. Disponível em http://www.portalsaofrancisco.com.br/alfa/meio-ambiente-energia-eolica/energia-eolica-8.php, visitado a última vez em 21 de Agosto de 2009.

[4] Stirling. Motores de Stirling e explicação do seu funcionamento. Disponível em http://stirlingenergy.com/ e http://carros.hsw.uol.com.br/motores-stirling.htm, visitado a última vez em 21 de Agosto de 2009.

[5] Cleanergy Industries. Sistemas de energia solar utilizando motores de Stirling. Disponível em http://www.cleanergyindustries.com e em http://guillot.emmanuel.free.fr/solar/stirling/photos.php?lang=en, visitado a última vez em 21 de Dezembro de 2009.

[6] Projecto Hydrosol. Produção de hidrogénio a partir da energia solar. Disponível em http://www.hydrosol-project.org/, visitado a última vez em 21 de Agosto de 2009.

[7] WindAtlas. European wind atlas. Disponível em http://www.windatlas.dk, visitado a última vez em 12 de Outubro de 2009.

[8] Fabricante de turbinas domésticas Savonius. Disponível em http://www.helixwind.com, visitado a última vez em 12 de Outubro de 2009.

[9] Fabricante de turbinas domésticas Savonius. Disponível em http://www.re-energy.ca/t-i_windbuild-1.shtml, visitado a última vez em 12 de Outubro de 2009.

[10] Rotor de Lenz. Disponível em http://www.cientificosaficionados.com/foros/viewtopic.php?t=5298, visitado a última vez em 12 de Outubro de 2009.

[11] Exemplos de Turbinas Darrieus. Construção de uma turbina Darrieus. Disponível em http://www.windstuffnow.com/main/darrieus_type.htm, visitado a última vez em 12 de Outubro de 2009.

[12] Turbinas Darrieus. Disponível em http://www.ifb.uni-stuttgart.de/~doerner/eDarrieus.html, visitado a última vez em 12 de Outubro de 2009.

[13] Turbinas de eixo horizontal com rotores de três pás. Disponível em http://www.treehugger.com/files/2007/04/worlds_largest_6.php, visitado a última vez em 12 de Outubro de 2009.

[14] CRESESB. Centro de Referência para Energia Solar e Eólica Sérgio de Salvo Brito, Agosto 2009. Disponível em http://www.cresesb.cepel.br/, visitado a última vez em 21 de Agosto de 2009.

[15] Turbinas eólicas submersas. Disponível em http://www.marineturbines.com, visitado a última vez em 12 de Outubro de 2009.

[16] Joseph T. Foley e Timothy G. Gutowski. Turbsim: Reliability-based wind turbine simulator. *IEEE International Symposium on Electronics and the Environment*, páginas 1–5, May 2008. ISBN 978-1-4244-2272-2.

[17] P. J. Tavner, J. Xiang e F. Spinato. Reliability analysis for wind turbines. *Wind Energy*, páginas 1–18, Jul 2007. ISBN DOI: 10.1002/we.204.

[18] Berthold Hahn, Michael Durstewitz e Kurt Rohrig. *Wind Energy, Reliability of Wind Turbines*. Springer Berlin Heidelberg, February 2007. ISBN 978-3-540-33865-9.

[19] Wikipedia. Ant colony optimization. Disponível em http://en.wikipedia.org/wiki/Ant_colony_optimization, visitado a última vez em 12 de Outubro de 2009.

[20] José L. F. Reis e Filipe Lopes et al. Análise e comparação de algoritmos de cálculo de posições GPS. Em *Quartas Jornadas de Engenharia de Electrónica e Telecomunicações e de Computadores, ISEL, 20-21 Nov*, 2008.

[21] Bobinadora Oliveira, Lda. Disponível em http://www.bobinadoraoliveira.com/frmsetInfTecnica.htm, visitado a última vez em 21 de Dezembro de 2009.

[22] Paul A. Schulte e HeeKyoung Chun. Climate change and occupational safety and health: Establishing a preliminary framework. *Journal of Occupational and Environmental Hygiene, Vol. 6, Nº 9*, páginas 542–554, 2009. ISSN 1545-9624.

[23] Unesco. Case studies on climate change and world heritage (2009 edition), Agosto 2009. Disponível em http://whc.unesco.org, visitado a última vez em 21 de Agosto de 2009.

[24] JET-Joint European Torus. Nuclear fusion research facility, Agosto 2009. Disponível em http://www.jet.efda.org, visitado a última vez em 21 de Agosto de 2009.

[25] Paineís fotovoltaicos. Descoberta célula solar com 40% de eficiência, Julho 2009. Disponível em http://spie.org/x35978.xml?ArticleID=x35978/, visitado a última vez em 21 de Agosto de 2009.

[26] Infinia Corp. Sistemas de energia solar utilizando motores de Stirling, Agosto 2009. Disponível em http://www.infiniacorp.com, visitado a última vez em 21 de Agosto de 2009.

[27] AquaMarine Power. Aproveitamento da energia das ondas do mar, Agosto 2009. Disponível em http://www.aquamarinepower.com/, visitado a última vez em 21 de Agosto de 2009.

[28] StatoilHudro. Turbina eólica flutuante em alto mar, Agosto 2009. Disponível em http://www.statoilhydro.com/en/, visitado a última vez em 21 de Agosto de 2009.

[29] T. W. Verbruggen. Wind turbine operation & maintenance based on condition monitoring. Relatório técnico, ECN Project Number: ECN-C-03-047, April 2003.

[30] P. J. OSullivan et al. Using UPM for real-time multivariate modeling of semiconductor manufacturing equipment. *SEMATECH APC*, páginas 1–18, Nov. 1995.

[31] Triant. Triant's modelware software for monitor equipment health and detect faults, Agosto 2005. Disponível em http://www.triant.com/, visitado a última vez em 21 de Agosto de 2009.

[32] M. A. Drewry e G. A. Georgiou. A review of NDT techniques for wind turbines. Disponível em www.ndt.net/search/docs.php3?id=4550, visitado a última vez em 21 de Agosto de 2009.

[33] Philip J. Morris, Lyle N. Long e Kenneth S. Brentner. An aeroacoustic analysis of wind turbines. *42nd AIAA Aerospace Sciences Meeting and Exhibit*, páginas 1–11, Jan. 2004.

[34] Curtiss-Wright Corporation. Healthcar monitoring for wind turbines with SWANwind, Agosto 2009. Disponível em http://swantech.curtisswright.com/, visitado a última vez em 21 de Agosto de 2009.

[35] David A. Spera. *Wind Turbine Technology: Fundamentals Concepts of Wind Turbine Engineering*. ASME Press (American Society of Mechanical Engineers), Segunda edição, 2009. ISBN 978-0791802601.

[36] P. Caselitz e J. Giebhardt et al. Development of fault detection system for wind energy converters. *EUWEC 96 Gotenborg*, páginas 1–4, Jan. 1996.

[37] P. Caselitz, J. Giebhardt e M. Mevenkamp. On-line fault detection and prediction in wind energy converters. *EWEC 94 - 5th European Wind Energy Conference and Exhibition*, páginas 1–4, Jan. 1994. Disponível em http://www.iset.uni-kassel.de/abt/FB-E/papers/FFE_EWEC_94.pdf, visitado a última vez em 21 de Agosto de 2009.

[38] P. Caselitz e J. Giebhardt. Advanced condition monitoring system for wind energy converters. *Proceedings EWEC'99, Nice, France*, páginas 33–66, Jan. 1999.

[39] P. Caselitz e J. Giebhardt. Advanced maintenance and repair for offshore wind farms using fault prediction techniques. *Berlim*, páginas 1–6, Jan. 2002.

[40] P. Caselitz e J. Giebhardt. Fault prediction techniques for offshore wind farm maintenance and repair strategies. *Madrid*, páginas 1–6, Jan. 2003.

[41] Ron J. Patton, Paul M. Frank e Robert N. Clarke. *Fault diagnosis in dynamic systems: theory and application*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1989. ISBN 0-13-308263-6.

[42] Maintenance World-OnLine. Motor maintenance - scheduled routine care, Agosto 2005. Disponível em http://www.maintenanceworld.com/Articles/relianceelectric/maintenance-troubleshooting-electric-motors.htm, visitado a última vez em 21 de Agosto de 2009.

[43] Vibrametrics. Sensores de aquisição de dados de vibrações, Agosto 2005. Disponível em http://www.vibrametrics.com, visitado a última vez em 21 de Agosto de 2009.

[44] I. E. Alguindigue, A. Loskiewicz-Buczak e R. E. Uhrig. Monitoring and diagnosis of rolling element bearings using artificial neural networks. *IEEE Transactions on Industrial Electronics*, páginas 209–217, Apr. 1993.

[45] GePower. Improved wind turbine condition monitoring using acceleration enveloping, Agosto 2005. Disponível em http://www.gepower.com/system1, visitado a última vez em 21 de Agosto de 2009.

[46] Viriato Marques. *Diagnóstico de falhas em equipamentos baseado em informação difusa oriunda dos técnicos de manutenção*. Tese de doutoramento, Faculdade de Engenharia da Universidade do Porto, Portugal, 2001. Disponível em http://repositorio-aberto.up.pt/handle/10216/11640, visitado a última vez em 3 de Dezembro de 2009.

[47] Zimin Yang. *Dynamic maintenance scheduling using online information about system condition*. Tese de doutoramento, Universidade de Michigan, Ann Arbor, MI 48109 USA 734-764-1817, 2005.

[48] R. Singh, S. O. T. Ogaji e S. D. Probert. Multiple-sensor fault-diagnoses for a 2-shaft stationary gas-turbine. *Elsevier, Applied Energy, Vol. 71, Nº 4*, páginas 321–339, Apr. 2002.

[49] J. Lehtoranta e H. N. Koivo. Fault diagnosis of induction motors with dynamical neural networks. *IEEE International Conference on Systems, Man and Cybernetics, Vol. 3*, páginas 2979–2984, Oct. 2005. ISBN 0-7803-9298-1.

[50] L. B. Jack. Feature selection for ANNs using genetic algorithms in condition monitoring. *ESANN'1999 proceedings - European Symposium on Artificial Neural Networks*, páginas 313–318, April 1999.

[51] Abhinav Saxena e Ashraf Saad. *Genetic Algorithms for Artificial Neural Net-based, Condition Monitoring System Design for Rotating, Mechanical Systems*. Springer Berlin / Heidelberg, Vol. 34, 2006. ISBN 978-3-540-31649-7.

[52] William B. Langdon e Wolfgang Banzhaf. Repeated sequences in linear genetic programming genomes. *Complex Systems, Vol. 15*, páginas 285–306, 2005. Disponível em http://www.cs.uwyo.edu/~wspears/gpe, visitado a última vez em 21 de Agosto de 2009.

[53] RML Technologies Inc. Discipulus, linear-genetic-programming software: How it works, Agosto 2006. Disponível em http://www.aimlearning.com, visitado a última vez em 21 de Agosto de 2009.

[54] Christopher Fogelberg e Mengjie Zhang. VUWLGP - An ansi c++ linear genetic programming package, December 2005. Disponível em http://www.syntilect.com/vuwlgp/, visitado a última vez em 21 de Agosto de 2009.

[55] Larry M. Deschaine, Janardan J. Patel, Ronald D. Guthrie, Joseph T. e Joseph T. Grumski. Using linear genetic programming to develop a c/c++ simulation model of a waste incinerator. Em *Simulation International: Advanced Simulation Technology Conference*, páginas 1–8, 2001.

[56] João Pedro Guerreiro Neto. *Construção Modular de redes Neuronais Recorrentes Analógicas*. Tese de doutoramento, Faculdade de Ciências da Universidade de Lisboa, Campo Grande, 1749-016 Lisboa, Portugal, 2002.

[57] Won-Yong Lee. Subsystem level fault diagnosis of a building's air-handling unit using general regression neural networks. *Elsevier*, 2003.

[58] John Olav G. Tande. Dynamic models of wind farms for power system studies. *Status by IEA Wind R&D, EWEC2004, UK*, páginas 22–25, 2004.

[59] Kjetil Uhlen. Wind farm measurements and modelling. *EWEC2006*, 2006. Disponível em http://www.ewec2006proceedings.info, visitado a última vez em 21 de Agosto de 2009.

[60] Jan Pierik. Development and validation of wind farm models for power system studies. *EWEC2006*, 2006. Disponível em http://www.ewec2006proceedings.info, visitado a última vez em 21 de Agosto de 2009.

[61] Ana Estanqueiro. A wind park dynamic model for power systems studies. *EWEC2006*, Fev. 2006. Disponível em http://www.ewec2006proceedings.info, visitado a última vez em 21 de Agosto de 2009.

[62] Institute of Energy Technology; Alborg University. Uma ferramenta de simulação de turbinas eólicas para Matlab, Dezembro 2005. Disponível em http://www.iet.aau.dk, visitado a última vez em 21 de Agosto de 2009.

[63] Z. Hameed, Y.S. Hong, Y.M. Cho, S.H. Ahn e C.K. Song. Condition monitoring and fault detection of wind turbines and related algorithms: A review. *Renewable and Sustainable Energy Reviews*, 13(1):1 – 39, 2009. ISSN 1364-0321.

[64] Mari Cruz Garcia, Miguel A. Sanz-Bobi e Javier del Pico. SIMAP: Intelligent system for predictive maintenance: Application to the health condition monitoring of a windturbine gearbox. *Computers in Industry*, 57(6):552 – 568, 2006. ISSN 0166-3615. E-maintenance Special Issue.

[65] Y. Amirat, M.E.H. Benbouzid, E. Al-Ahmar, B. Bensaker e S. Turri. A brief status on condition monitoring and fault diagnosis in wind energy conversion systems. *Renewable and Sustainable Energy Reviews*, 13(9):2629 – 2636, 2009. ISSN 1364-0321.

[66] M. Gabriel e Y. Primor. *Maintenance Assisté par Ordinateur*. Masson edição, 1985.

[67] *Links* de *software* de manutenção. Listagem de *Software* de gestão da manutenção, Agosto 2009. Disponível em http://www.plant-maintenance.com/freestuff/index.shtml e em http://www.plant-maintenance.com/maintenance_software_CMMS_A-L.shtml, visitado a última vez em 21 de Agosto de 2009.

[68] J. Torres Farinha. *Uma Abordagem Terológica da Manutenção dos Equipamentos Hospitalares*. Tese de doutoramento, Faculdade de Engenharia da Universidade do Porto, Portugal, 1994. Disponível em http://repositorio-aberto.up.pt/handle/10216/12810, visitado a última vez em 3 de Dezembro de 2009.

[69] Arilde Sutil Gabriel de Camargo. Análise da operação das usinas eólicas de Camelinho e Palmas e avaliação do potencial eólico de localidades no Paraná. Tese de mestrado, Universidade Federal do Paraná, 2005. Disponível em http://www.ppgte.cefetpr.br/dissertacoes/2005/acamargo.pdf, visitado a última vez em 11 de Outubro de 2008.

[70] International Energy Agency. Relatórios técnicos - key world energy statistics, 2009. Disponível em http://www.iea.org e http://www.ieawind.org, visitado a última vez em 21 de Agosto de 2009.

[71] ADENE/INETI, editor. *Energias Renováveis em Portugal*. 2002. ISBN 972-8646-01-1.

[72] European Community. Disponível em http://www.europa.eu, visitado a última vez em 21 de Agosto de 2009.

[73] Giacomo Bizzarri e Gianluca Morini. A life cycle analysis of roof integrated photovoltaic systems. *International Journal of Environmental Technology and Management, Vol. 7, Nº. 1/2*, páginas 134–146, 2007.

[74] Christiana Honsberg e Allen Barnett. University Of Delaware-led Team Sets Solar Cell Record. Disponível em http://www.sciencedaily.com/releases/2007/07/070726210931.htm, visitado a última vez em 21 de Agosto de 2009.

[75] John F. Geisz e Daniel Friedman. Engineering the world's most efficient solar cells. Disponível em http://spie.org/x35978.xml?ArticleID=x35978, visitado a última vez em 21 de Agosto de 2009.

[76] Fraunhofer-Gesellschaft. Lasers are making solar cells competitive. Disponível em http://insciences.org/article.php?article_id=5333, visitado a última vez em 21 de Agosto de 2009.

[77] Sergio Hiroshi Toma. *Química supramolecular e aplicações nanotecnológicas de compostos polipiridínicos de rutênio*. Tese de doutoramento, Universidade de São Paulo, 2007. Disponível em http://www.teses.usp.br/teses/disponiveis/46/46134/tde-24082007-074550/, visitado a última vez em 11 de Outubro de 2008.

[78] George Whitfield. Solar tracking with smart materials. Disponível em http://dmse.mit.edu/madmec/helio.html, visitado a última vez em 14 de Outubro de 2009.

[79] Nanosolar Inc. Nanosolar SolarPly. Disponível em http://www.nanosolar.com/products.htm, visitado a última vez em 21 de Agosto de 2009.

[80] Company 1366tech Inc. Aumento da eficiência das células solares policristalinas. Disponível em http://1366tech.com, visitado a última vez em 21 de Agosto de 2009.

[81] Dale K. Kotter, Steven D. Novack e W. Dennis Slafer. Solar Nantenna Electromagnetic Collectors. *Proceedings of ES2008, Energy Sustainability 2008, August 10-14, 2008, Jacksonville, Florida USA*, 2008.

[82] Universidade de Utah. Sistemas de conversão termo-acústicos. Disponível em http://www.utah.edu/, visitado a última vez em 21 de Agosto de 2009.

[83] Universidade de Massachusetts. http://www.umb.edu/.

[84] História da energia eólica. Disponível em http://www.alternate-energy-sources.com/what-is-wind-energy.html, visitado a última vez em 21 de Agosto de 2009.

[85] WindPower. Windpower, danish wind industry association annual report. ISBN 1397-1697. Disponível em http://www.windpower.org, visitado a última vez em 21 de Agosto de 2009.

[86] Exemplos de Turbinas Savonius. Construção de uma turbina Savonius. Disponível em http://www.macarthurmusic.com/johnkwilson/MakingasimpleSavoniuswindturbine.htm e http://www.electronica-pt.com/files/turbina-eolica.pdf, visitado a última vez em 12 de Outubro de 2009.

[87] Estudo sobre a utilização de turbinas nas cidades. Disponível em http://www.eru.rl.ac.uk/web.htm, visitado a última vez em 12 de Outubro de 2009.

[88] Universidade de Hong-Kong. Estudo de desenvolvimento de turbinas eólicas citadinas, em conjunto com a empresa Motorwave Ltd. Disponivel em http://www.hku.hk/ e http://www.motorwavegroup.com, visitado a última vez em 14 de Outubro de 2009.

[89] Oxford Aviation Training. *080 Principles of Flight,How to Get a PhP*. Jeppesen GmbH, Frankfurt, Germany, Segunda edição, 2001. ISBN 0-88487-277-7.

[90] Global Wind Energy Council. Global wind 2008 report. Disponível em http://www.gwec.net/fileadmin/documents/Global%20Wind%202008%20Report.pdf, visitado a última vez em 14 de Outubro de 2009.

[91] Kendall Correll, Nick Barendt e Michael Branicky. Design Considerations for Software Only Implementations of the IEEE 1588 Precision Time Protocol.

[92] B. P. Lathi. *Modern Digital and Analog Communiction Systems*. Oxford University Press, Terceira edição, 1998. ISBN 1-800-334-4249.

[93] Paul Horowitz e Winfield Hill. *The Art of Electronics*. Cambridge University Press, Segunda edição, 1994. ISBN 0-521-37095-7.

[94] Luciano Junqueira Campos, António Tadeu A. Gomes e Luiz Fernando G. Soares. Sincronismo de Vídeo MPEG-2 VBR Sobre Redes ATM. *Boletim bimestral sobre tecnologia de redes, Vol. 4, Nº 4*, 2000. ISSN 1518-5974. Disponível em http://www.rnp.br/newsgen/0007/art9.html, visitado a última vez em 21 de Agosto de 2009.

[95] Sandia Reports. Wind turbine reliability: Understanding and minimizing wind turbine operation and maintenance costs,2006.

[96] M. M. Khan, M. T. Iqbal e F. Khan. Reliability analysis and condition monitoring of a wind turbine,2005. *Canadian Conference on Electrical and Computer Engineering, Vol. Issue*, páginas 1978–1981, May 2005.

[97] Berthold Hahn, Michael Durstewitz e Kurt Rohrig. *Advanced Maintenance and Repair for OffShore Wind Farms using Fault Prediction and Condition Monitoring Techniques*. contract NNE5/2001/710, 2006.

[98] Alex J. Smola e Bernhard Scholkopf. A tutorial on support vector regression. Relatório técnico, NeuroCOLT Technical Report Series, NC2-TR-1998-030, October 1998. Disponível em http://eprints.pascal-network.org/archive/00002057/01/SmoSch03b.pdf, visitado a última vez em 21 de Agosto de 2009.

[99] Bernhard Scholkopf. Statistical learning and kernel methods. Relatório técnico, Microsoft Research Limited, February 2000. Disponível em http://research.microsoft.com/apps/pubs/default.aspx?id=69756, visitado a última vez em 21 de Agosto de 2009.

[100] B. Scholkopf e A. J. Smola. *Learning with Kernels*. MIT Press, Cambridge, MA, 2002. ISBN 978-0-262-19475-4.

[101] Gert Cauwenberghs e Tomaso Poggio. Incremental and decremental support vector machine learning. *Adv. Neural Information Processing Systems (NIPS*2000)*, 2001. Disponível em http://www.biology.ucsd.edu/~gert/papers/nips00_inc.pdf, visitado a última vez em 21 de Agosto de 2009.

[102] V. Kecman. *Learning and Soft Computing*. MIT Press, Cambridge, MA, 2001. ISBN 978-0-262-11255-0.

[103] J. A. K. Suykens e T. Van Gestel et al. Least squares support vector machines. World Scientific, Singapore, 2002.

[104] Vojtech Franc e Vaclav Hlavac. Statistical pattern recognition toolbox for Matlab. Disponível em http://cmp.felk.cvut.cz/, visitado a última vez em 12 de Outubro de 2009.

[105] Hakan Calis e Abdulkadir Cakir. *Experimental study for sensorless broken bar detection in induction motors*. Energy Convers Manage (2007), doi:10.1016/j.enconman.2007.06.030.

[106] Sanna Poyhonen. Support vector machine based classification in condition monitoring of induction motors. Relatório técnico, Helsinki University of Technology Control Engineering Laboratory.

[107] T. Rothenberg. Univariate time series models. Relatório técnico, Econ 241b, Fall, 2005. Disponível em http://www.ms.unimelb.edu.au/~s620374/timeseries/Rothenberg_ARMA_05.pdf, visitado a última vez em 21 de Agosto de 2009.

[108] George E. P. Box, Gwilym M. Jenkins e Gregory C. Reinsel. *Time Series Analysis: Forecasting and Control*. Fourth edição, July 2008. ISBN 978-0-470-27284-8.

[109] Gláucia de Paula Falco. Técnicas univariadas aperfeiçoadas para a previsão de curtíssimo prazo a partir de dados horários. Tese de mestrado, Pontifícia Universidade Católica do Rio de Janeiro, 2006. Disponível em http://www2.dbd.puc-rio.br/pergamum/tesesabertas/0321196_05_Indice.html, visitado a última vez em 11 de Outubro de 2008.

[110] Ricardo S. Ehlers. Análise de séries temporais. Laboratório de Estatística e Geoinformação, Universidade Federal do Paraná, 2007. Disponível em http://www.icmc.usp.br/~ehlers/stemp/, visitado a última vez em 12 de Outubro de 2009.

[111] Mauri Aparecido de Oliveira. *Aplicação de Redes Neurais Artificiais na Análise de Séries Temporais Económico-Financeiras*. Tese de doutoramento, Universidade de São Paulo, 2007. Disponível em http://www.teses.usp.br/teses/disponiveis/12/12139/tde-31012008-112504/, visitado a última vez em 3 de Dezembro de 2009.

[112] Anderson da Silva Soares. Predição de séries temporais econômicas por meio de redes neurais artificiais e transformada Wavelet: Combinando modelo técnico e fundamentalista. Tese de mestrado, Universidade de São Paulo, 2008. Disponível em http://www.teses.usp.br/teses/disponiveis/18/18152/tde-11042008-111842/, visitado a última vez em 11 de Outubro de 2008.

[113] Maria Cristina Felippetto de Castro. *Predição Não-Linear de Séries Temporais Usando Redes Neurais RBF por Decomposição em Componentes Principais*. Tese de doutoramento, Universidade Estadual de Campinas, 2001. Disponível em http://libdigi.unicamp.br/document/?code=vtls000220226, visitado a última vez em 3 de Dezembro de 2009.

[114] Ming-Wei Chang, Chih-Jen Lin e Ruby C. Weng. Analysis of nonstationary time series using support vector machines. Em Seong-Whan Lee e Alessandro Verri, editores, *Proceedings of SVM 2002*, Lecture Notes in Computer Science 2388, páginas 160–170, New York, NY, USA, 2002. Springer-Verlag Inc. Disponível em http://flake.cs.uiuc.edu/~mchang21/publication/publications.html, visitado a última vez em 21 de Agosto de 2009.

[115] Luzia Vidal De Souza. *Programação Genética e Combinação de Preditores para Previsão de Séries Temporais*. Tese de doutoramento, Universidade Federal do Paraná, 2006. Disponível em http://www.ppgmne.ufpr.br/arquivos/teses/4.pdf, visitado a última vez em 3 de Dezembro de 2009.

[116] Ricardo de A. Araújo, Germano C. Vasconcelos e Tiago A. E. Ferreira. An evolutionary morphological-rank-linear approach for time series prediction. *2007 IEEE Congress on Evolutionary Computation (CEC 2007)*, páginas 4321–4328, Sept 2007.

[117] ZHANG Wei, WU Zhi-ming e YANG Gen-ke. Genetic programming-based chaotic time series modelling. *Journal of Zhejiang University SCIENCE, ISSN 1009-3095*, páginas 1432–1439, 2004. Disponível em http://www.zju.edu.cn/jzus/2004/0411/041118.pdf, visitado a última vez em 11 de Outubro de 2008.

[118] Sara Silva. *Controlling Bloat: Individual and Population Based Approaches in Genetic Programming*. Tese de doutoramento, Universidade de Coimbra, 2008. Disponível em http://cisuc.dei.uc.pt/ecos/dlfile.php?fn=1749_pub_phdsara.pdf&idp=1749, visitado a última vez em 3 de Dezembro de 2009.

[119] Damian Eads, Daniel Hillb e Sean Davis. Genetic algorithms and support vector machines for time series classification. páginas 74–85, Jul. 2002. ISBN 0-8194-4554-1. Disponível em http://users.soe.ucsc.edu/~eads/papers/eads2002.pdf, visitado a última vez em 21 de Agosto de 2009.

[120] Vedat Yorucu. The analysis of forecasting performance by using time series data for two Mediterranean Islands. *Review of Social, Economic and Business Studies, Vol.2*, páginas 175–196. Disponível em http://fbe.emu.edu.tr/journal/doc/2/2Article09.pdf, visitado a última vez em 21 de Agosto de 2009.

[121] Ching-Wu Chu. A comparative study of linear and nonlinear models for aggregate retail sales forecasting. *International Journal of Production Economics, Vol. 86, Nº 3*, páginas 217–231, 2003. ISSN 0925-5273. Disponível em http://www.sciencedirect.com/science/article/B6VF8-48NJ5KD-3/2/522f214de35653b7e8f3908dd32a6070, visitado a última vez em 21 de Agosto de 2009.

[122] William Cook. The traveling salesman problem. Disponível em http://www.tsp.gatech.edu/, visitado a última vez em 12 de Outubro de 2009.

[123] Marco Dorigo. Ant colony optimization. Disponível em http://www.aco-metaheuristic.org, visitado a última vez em 12 de Outubro de 2009.

[124] Jorge Guerreiro, Alípio Magalhães e Manuel Ramalhete. *Programação Linear (Volume I e II)*. McGraw-Hill, 1985. ISBN 972-9241-71-6.

[125] Oliver van Kaick, Ghassan Hamarneh, Hao Zhang e Paul Wighton. Contour correspondence via ant colony optimization. Em *Proc. 15th Pacific Conference on Computer Graphics and Applications (PG'2007)*, páginas 271–280, 2007.

[126] I. Zamboni e L. H. A. Monteiro. Optimization of the topology of electric energy distribution networks by using algorithm inspired on ant behaviors. Em *IEEE Latin America Transactions, Vol. 7, Nº 1*, páginas 85–91, March 2009.

[127] Eric Demaine. MIT Open Courses, Lecture 17: Shortest Paths I: Properties, Dijkstra's Algorithm, Breadth-first Searc. Disponível em http://videolectures.net/mit6046jf05_demaine_lec17, visitado a última vez em 14 de Outubro de 2009.

[128] Eric Demaine. MIT Open Courses, Lecture 18: Shortest Paths II: Bellman-Ford, Linear Programming, Difference Constraints. Disponível em http://videolectures.net/mit6046jf05_demaine_lec18, visitado a última vez em 12 de Novembro de 2009.

[129] Kok Keong Teo, Lipo Wang e Zhiping Lin. Wavelet packet multi-layer perceptron for chaotic time series prediction: Effects of weight initialization school of electrical and electronic engineering. Em *Springer-Verlag Berlin Heidelberg LNCS 2074*, páginas 310–317, 2001.

[130] R. R. Sharapov e A. V. Lapshin. Convergence of genetic algorithms. *Pattern Recognition and Image Analysis, ISSN: 1054-6618, Vol. 16, N. 3*, páginas 392–397, Jul. 2006. Disponível em http://www.springerlink.com/content/1845u6j09t68878m/, visitado a última vez em 11 de Outubro de 2008.

[131] Anna Mikusheva e Paul Schrimpf. MIT Open Courses, 14.384 Time Series Analysis. Disponível em http://ocw.mit.edu/OcwWeb/Economics/14-384Fall-2008/CourseHome/index.htm, visitado a última vez em 14 de Outubro de 2009.

[132] Cambridge University, editor. *The book Numerical Recipes in C: The Art of Scientific Computing*. Terceira edição, 2007. ISBN 978-0521880688. Disponível em http://www.nr.com/, visitado a última vez em 12 de Outubro de 2009.

[133] K. Madsen, H. B. Nielsen e O. Tingleff. Methods for non-linear Least Squares Problems. Relatório técnico, Informatics and Mathematical Modelling, Technical University of Denmark, 2nd Edition, April 2004. Disponível em http://www2.imm.dtu.dk/pubdb/views/edoc_download.php/3215/pdf/imm3215.pdf, visitado a última vez em 21 de Agosto de 2009.

[134] Pedro Lopes. Sobre desenvolvimentos em séries de potências, séries de Taylor e fórmula de Taylor. Relatório técnico, Dept. de Matemética do Instituto Superior Técnico. Disponível em http://www.math.ist.utl.pt/~pelopes/taylor06.pdf, visitado a última vez em 21 de Agosto de 2009.

[135] Markus Braimeier e Wolfgang Banzhaf. *Linear Genetic Programming - Genetic and Evolutionary Computation*. Springer, First edição, 2007. ISBN 0-387-31029.

[136] T. Aroui, Y. Koubaa e A. Toumi. Application of feedforward neural network for induction machine rotor faults diagnostics using stator current. Disponível em http://journal.esrgroups.org/jes/papers/3_4_3.pdf, visitado a última vez em 12 de Outubro de 2009.

[137] P. C. M. L. Filho e R. Pederiva e J. N. Brito. Utilização da lógica nebulosa para detecção defeitos em motores de indução trifásicos. *8º Congresso IberoAmericano de Engenharia Mecânica*, 2007. Disponível em http://www.pucp.edu.pe/congreso/cibim8/pdf/06/06-110.pdf, visitado a última vez em 21 de Agosto de 2009.

[138] Darley F. A. Santiago e Robson Pederiva. Comparison of optimization techniques of neural networks training for faults diagnostic of rotating machinery. *Mecánica Computacional, Vol. XXI*, páginas 1912–1921, 2007. Disponível em http://www.cimec.org.ar/ojs/index.php/mc/article/viewFile/997/945, visitado a última vez em 21 de Agosto de 2009.

[139] Izzet Y. Onel, K. Burak Dalci e Ibrahim Senol. Detection of bearing defects in Three-Phase induction motors using Park's transform and radial basis function neural networks. *Sadhana, Part 3, Vol. 31*, páginas 235–244, Jun 2006. Disponível em http://www.ias.ac.in/sadhana/Pdf2006Jun/235.pdf, visitado a última vez em 21 de Agosto de 2009.

[140] Tian Han, Bo-Suk Yang, Won-Ho Choi e Jae-Sik Kim. Fault diagnosis system of induction motors based on neural network and genetic algorithm using stator current signals. *International Journal of Rotating Machinery, Vol. 2006, Article ID 61690, DOI 10.1155/IJRM/2006/61690*, páginas 1–13. Disponível em www.hindawi.com/getpdf.aspx?doi=10.1155/ijrm/2006/61690, visitado a última vez em 21 de Agosto de 2009.

[141] Jackson Paul Matsuura. Detecção de falhas em sistemas dinâmicos com redes bayesianas aprendidas a partir de estimação de estados. *Instituto Tecnológico de Aeronaútica, 2006*. Disponível em http://www.ele.ita.br/~jackson/files/doc.pdf, visitado a última vez em 21 de Agosto de 2009.

[142] Alexandro Goedtel. Estimador neural de velocidade para motores de indução trifásicos. *Universidade de São Paulo, 2007*. Disponível em http://www.teses.usp.br/teses/disponiveis/18/18153/tde-18122007-204855/publico/TeseAlessandroGoedtel.pdf, visitado a última vez em 21 de Agosto de 2009.

[143] Alessandro Goedtel, Paulo J. A. Serni e Ivan N. da Silva. Uma abordagem neural para estimativa de conjugado em motores de indução. Disponível em http://www.scielo.br/pdf/ca/v17n3/a10v17n3.pdf, visitado a última vez em 21 de Agosto de 2009.

[144] Luís Oscár Araújo Porto Henriques. *Implementação de estratégia de minimização de oscilações de torque e remoção de sensor de posição para um accionamento de relutância variável usando técnica neuro-fuzzy*. Tese de doutoramento, Universidade Federal do Rio de Janeiro, 2004. Disponível em http://www.coe.ufrj.br/~porto/tese/Tese_DSc_Luis_Oscar_Henriques_Ago_04.pdf, visitado a última vez em 3 de Dezembro de 2009.

[145] W. T. Thomson e M. Fenger. Case histories of current signature analysis to detect faults in induction motor drives. *IEEE International, Electric Machines and Drives Conference, Vol. 3*, páginas 1459–1465, 2003. ISBN 0-7803-7817-2. Disponível em http://www.adwel.com/PDF/techpapers/Years/Other/Case%20Histories%20of%20Current%20Signature%20Analysis%20to%20Detect%20Faults%20in%20Induction%20Motor%20Drives.pdf, visitado a última vez em 21 de Agosto de 2009.

[146] Jorge M.Barreto. Introdução às redes neurais artificiais. Disponível em http://www.inf.ufsc.br/~barreto/tutoriais/Survey.pdf, visitado a última vez em 12 de Outubro de 2009.