NTNU

Norwegian University of
Science and Technology

Multiple Sensor Data Analysis, Fusion, and Communication for ULTRASPONDER

Carlos Sergio Gutiérrez Perera

Master of Science in Electronics
Submission date: September 2009
Supervisor: Ilangko Balasingham, IET

Norwegian University of Science and Technology
Department of Electronics and Telecommunications

Problem Description

It is believed that monitoring blood pressure inside the heart may give vital information to correctly diagnose and provide treatment for chronic heart failure patients. The EU Project ULTRASPONDER (In vivo Ultrasonic Transponder System for Biomedical Applications), develops a novel transponder technology based on ultrasonic energy scavenging and wireless communication with pressure sensors implanted deep inside the heart cavity. The sensors communicate with a control unit implanted just underneath the skin, where external communication is based on radio frequency.

This thesis includes a part of the study intended to be performed in the ULTRASPONDER, where the main areas of interest are to study how to combine signals from the pressure sensors, ECG electrodes, and other type of sensors, and to understand the physiological and biological changes. Furthermore it is desired to transmit such information in a resource constraint manner in terms of energy, processing power, etc, from the control unit implanted in the patient to an external device (PC), which may have large amount of resources. Hence, the candidate will perform a theoretical analysis of the signals by considering information content, correlation among signals, etc, intended for data fusion with the aim of reducing the amount of information, or bits/sample, which would be transmitted.

A semi-specific ECG signal processing will be carried out, considering monitoring applications. Furthermore, signal processing for communication will be studied, where efficient source-channel coding and modulation can be considered.

The report should include detailed literature survey and performed computer simulation results.

Assignment given: 17. February 2009
Supervisor: Ilangko Balasingham, IET

Multiple Sensor Data Analysis, Fusion, and Communication for ULTRASPONDER

Carlos Sergio Gutiérrez Perera
Norges teknisk-naturvitenskapelige universitet i Trondheim
Universidad Politécnica de Valencia

2009

A mi madre,
de quien aprendí el amor por la lectura, la música
y la naturaleza. Entre otras muchas cosas.

A mi abuela,
de quien aprendí mis primeras nociones de inglés
en el asiento trasero de un coche de domingo en
domingo, y luego de los libros de Muzzy.

A mi abuelo,
in memoriam,
de quien tomé mi interés por la medicina,
y en cuya biblioteca descubrí años más tarde
un pequeño nuevo mundo para mí.

A mi hermano,
por ser siempre mi hermano.

A Cristina e Iván,
con quienes vengo aprendiendo, poco a poco,
a hacernos adultos y que la vida va en serio.

Acknowledgements

First of all, I would like to express my deep gratitude to all the people that make the ERASMUS exchange programme possible. The opportunity they give us in invaluable. In this context, I would like to thank the staff of the Office of International Relations in Gløshaugen campus, in NTNU, and the OPII (Oficina de Programas Internacionales de Intercambio) in Vera campus, in UPV. Most specially, of course, a million thanks to the people of SRE-APIT for their dedication and helpful tips, suggestions, etc.

I would also like to show my appreciation to all the people that reinforce the ERASMUS experience, like studental associations such as ESN or ISU, which I am honored to have worked within and alongside. Their actions help students to make the most of a long stay in a foreign country, enabling a more interesting, pleasant, enriching and fun experience.

I wish to express my profound acknowledgement and gratitude to Professor Ilangko Balasingham of Gruppen for Signalbehandling, Institutt for elektronikk og telekommunikasjon, NTNU, supervisor of this thesis, for his guidance and helpful advice related to the work involving the thesis as well as for his support regarding my personal obligations with my home university.

Similar feelings go to my many teachers in UPV who taught me during these years and became such an important part of all this process. Significantly, my deep thanks to those who made me discover a concrete interest inside the vast field of telecommunications engineering for the area of biomedicine and biomedical signals and images processing.

At the personal level, I cannot pass without declaring my most sincere gratitude to the many new friends I have met in this wonderful city of Trondheim, whom I expect to keep hearing from in the future years. Without their priceless company, nothing at all would have been the same. Specially heartfelt regards go to Hildrun Brendler, Heiko Luckmann, Valentina Parra, Ørjan Olsen, Seth Davis, Antonis Karapetrides, Michelle Karevoll...

The list of people whom I must be thankful to could be never-ending, since each person we meet touches us or affects us in some way that shapes what or who we become. Therefore, thanks to everybody who has been important for me and has made possible in one or other way, my road until here. It would be unfair to forget mentioning some/anybody, and certainly I do not forget anybody. But mentioning all is unfeasible, so I will focus on those who have

supported me the most in the achievement of my degree and the realization of this thesis.

Allow me to continue in Spanish to convey to my people my appreciation and thanks for everything.

Mi especial agradecimiento:

a Leyre Cendrero, por refrescarme la memoria sobre las prácticas de ASIB.
a Cristina Manzano, por recomendarme venir a Trondheim, y por las fresitas y lacitos durante el estudio veraniego hasta altas horas de la noche, años ha.
a Paula Mallén, por hacerme Erasmus con su motivación, ánimos y, por supuesto, con el impagable regalo de Pityië.
una vez más, a Cristina Messalino, por ese modelo de constancia, que aunque me empeñe, nunca he conseguido llegar a imitar. Por los años juntos, y ahora una amistad fuerte y duradera.
a Iván Vilar, por el inmenso apoyo logístico y emocional. Por la banda sonora, los descubrimientos, los recados, las idas y venidas y la incondicionalidad. Y por creer en mí incluso (mucho) más que yo mismo.
a Marta Abásolo y Marga Reyes, por enseñarme que los cambios pueden ser buenos, y por tantas cenas y visitas. Y por el queso, el jamón y el vino: sabor a hogar.
a Alf, por las impagables charlas frikis; a Alberto, por las sonrisas y las manos echadas; a Inés, por la complicidad y las postales; a Victor y Álvaro, por no perder el contacto a pesar de la distancia.
y, por supuesto, por último pero de ninguna manera menos importante, a mi familia, por el inmenso apoyo de todo tipo durante mi carrera, por el amor, las enseñanzas, la paciencia y todo lo demás.

Also, now that I come to think about it, many thanks:

to Richard O'Brien and Tim Curry for the music and the show;
to Terry Pratchett, Orson Scott Card, J.R.R. Tolkien and Jim Henson for the fantasy;
to Zhang Yimou, Kim Ki-duk, Lee Myung-jin and Hayao Miyazaki for the visual poetry;
to Pedro Almodóvar, José Saramago and Naguib Mahfouz for the incredible reality; all of them have somehow helped me in the tough task of keeping some sanity, so forgetting them in this point would be unforgiveable.

This has been a long long way, started years ago in Valencia, and finished here in Trondheim. It has been full of experiences and learning. But as wise men say, the end is not more than just a new beginning.

So perhaps, this point is just the end of the actual beginning...

Many more experiences and learning lie ahead, awaiting...

Abstract

This Thesis covers a part of the study comprised in the ULTRASPONDER (In vivo Ultrasonic Transponder System for Biomedical Applications) Project.

The main area of interest is to study how to combine different signals which can imply an improvement onto the diagnostic information carried by the ECG records. It is believed that monitoring blood pressure inside the heart may give vital information to correctly diagnose and provide treatment for chronic heart failure patients. Moreover, heart rate variability analysis has proved to be one of the most important risk predictors in detecting ventricular tachycardias and fluttering.

With this focus, the Thesis provides a solid background on cardiac anatomy and physiology, uncommon in many engineering texts, in order to understand the biological changes that affect the waveforms, for then moving to the performance of a theoretical and statistical study in order to find correlations, redundancies, or new information content in the signals intended to cohabit in the ULTRASPONDER control unit, namely, signals from the intra-cavity pressure sensors, ECG electrodes and other type of sensors, as well as heart rate time-series.

Because this control unit, implanted underneath the patient's skin, must handle several different signals and transmit clinically relevant information in a power constraint manner to an external device, which may have much larger amount of resources, all signal processing performed in the context of the control unit must be kept under a reasonable limit that permits to efficiently extract information about the patient's health without decreasing the device's lifetime.

We have implemented two time-domain QRS complexes detection systems, two simple beat classification algorithms based on beat-to-beat segmentation and template correlation, and some HRV measures as fundamental elements of ECG signal processing. Detection performance is analyzed from a critical point of view, considering several not so common parameters, such as Q_α and MCC, which collect much more information than the usual sensitivity and predictivity assessments.

A closed-loop DPCM system was chosen for the encoding and compression tasks, experiments showing its validity for ECG and blood pressure signals, although advising against its usage for HR time series. Compression performance is analyzed in terms of compression ratio attained and distorsion

introduced. A novel measure called "compressibility quotient" (CQ) is presented as an indicator of the balance between theoretical compression limits marked by the sample entropy and actual compression obtained with a concrete scheme, in terms of the tradeoff CR-distorsion. A strong correlation between signal-to-noise ratio and CQ was found, implying that this measure might have some relevance for analyzing real compression possibilities under some quality criteria.

The approaches followed in this Thesis, particularly regarding the theoretical study and data fusion comments, are valid for the ECG, blood pressure and heart rate signals considered, without detriment to be likewise applied to new signals that might become of interest in the future years. When new sensors are implemented to provide distinct signals, a theoretical study can include them to find out their usefulness and relation to the ones already considered. Data fusion should then be reviewed to assess the validity and convenience of the communication system for the new set of significant signals.

Sinopsis

Esta Tesis trata una parte del estudio comprendido en el Proyecto ULTRASPONDER (In vivo Ultrasonic Transponder System for Biomedical Applications).

El principal área de interés es el estudio acerca de cómo combinar diferentes señales que puedan suponer una mejora sobre la información diagnóstica contenida en los registros de ECG. Se considera que monitorizar la presión sanguínea en el interior de las cavidades cardiacas puede proporcionar información vital para diagnosticar correctamente y proveer tratamiento adecuado a pacientes con fallo cardíaco crónico. Adicionalmente, el análisis de variabilidad del ritmo cardíaco ha demostrado ser uno de los predictores de riesgo más importantes en la detección de taquicardias y fibrilación ventricular

Con este enfoque, la Tesis proporciona un sólido marco de referencia sobre la anatomía y fisiología cardíacas, infrecuente en textos de ingeniería, para entender los cambios biológicos que afectan a las formas de onda, para posteriormente pasar a realizar un estudio teórico y estadístico con el fin de encontrar correlaciones, redundancias, o nuevo contenido de información en las señales que se prevé cohabiten en la unidad de control ULTRASPONDER, es decir, señales provinientes de los sensores de presión intracavitarios, los electrodos de ECG y otro tipo de sensores, así como series temporales de ritmo cardíaco.

Puesto que dicha unidad de control, implantada bajo la piel del paciente, debe manejar varios tipos de señales diferentes y transmitir en un marco de potencia limitado la información clínicamente relevante a un dispositivo externo, que puede contar con unos recursos mucho mayores, todo el tratamiento de señal que se haga en el ámbito de la unidad de control debe ser controlado a un nivel razonable que permita la extracción eficiente de información sobre el estado de salud del paciente, sin disminuir la vida útil del dispositivo.

Se han implementado dos sistemas de detección de complejos QRS en el dominio temporal, dos algoritmos de clasificación de latidos basados en segmentación latido a latido y correlación con plantillas, y algunas medidas de variabilidad del ritmo cardíaco como elementos fundamentales de procesado de señal de ECG. Los resultados de la detección se analizan desde un punto de vista crítico considerando varios parámetros poco comunes, como Q_α y el MCC, que recogen mucha más información que las evaluaciones habituales basadas en sensibilidad y predictividad.

Se eligió un sistema DPCM de bucle cerrado para las labores de codificación y compresión, mostrando los experimentos realizados su validez para señales de ECG y presión sanguínea, si bien su uso sobre las series de ritmo cardíaco resulta desaconsejable. Se presenta una nueva medida llamada "cociente de compresibilidad" (CQ, del inglés *compressibility quotient*) como indicador del balance entre los límites teóricos de compresión impuestos por la entropía de una señal y la compresión actual obtenida empleando un esquema específico, en términos del compromiso entre tasa de compresión y distorsión. Se halló una fuerte correlación entre la relación señal-a-ruido y el CQ, lo cual significa que esta medida pudiera tener cierta relevancia para analizar las posibilidades de compresión real bajo ciertos criterios de calidad.

Los enfoques seguidos en esta Tesis, en particular en lo que respecta al estudio teórico y los comentarios sobre fusión de datos, son válidos para las señales de ECG, presión sanguínea y ritmo cardíaco considerados, sin detrimento de que puedan ser aplicados de forma similar a nuevas señales que pudieran suscitar interés en los próximos años. Cuando se implementaran nuevos sensores que proporcionaran señales distintas, un estudio teórico debería incluirlas para hallar su utilidad y relación con las ya consideradas. La fusión de datos debería ser revisada para evaluar la validez y conveniencia del sistema de comunicaciones para ese nuevo conjunto de señales significativas.

Contents

List of Figures

List of Tables

Nomenclature and Abbreviations

A/D	analog to digital
ABP	arterial blood pressure
AM	amplitude modulation
ANS	autonomic nervous system
AppEn	Approximate Entropy
AR	auto-regressive
ART	arterial blood pressure
AV	atrioventricular
b.p.m.	beats per minute (also found as bpm)
BP	blood pressure
c.r.v.	continuous random variable
CNS	central nervous system
CO	cardiac output
CQ	compressibility quotient
CR	compression ratio
CVP	central venous pressure
d.r.v.	discrete random variable
DC	direct current
DFA	detrended fluctuation analysis
DP	diastolic pressure
DPCM	differential pulse code modulation

ECG	electrocardiography, or electrocardiogram
EMG	electromyogram
ePAD	estimated pulmonary artery diastolic pressure
FAQ	Frequently asked questions
FFT	Fast Fourier Transform
FIR	finite impulse response
FN	false negatives
FOI	first-order interpolator
FOP	first-order predictor
FP	false positives
FPR	false positive rate, or false alarm rate
HR	heart rate
IDE	integrated development (or design) environment
IVS	interventricular septum
LPC	linear predictive coding
MAP	maximum a posteriori (estimation methods)
MAP	mean arterial pressure
MAX	maximum amplitude error
MCC	Matthews correlation coefficient
P_{pulse}	pulse pressure
PAP	pulmonary arterial pressure
PCG	phonocardiogram
PP	positive predictivity
PRD	percent root-mean-square difference
PSD	Power Spectral Density
PSD	power spectral distribution
r.v.	random variable
rms	root mean square error
ROC	receiver operating characteristic
SA	sinoatrial

SAN	sinoatrial node
Se	sensitivity
ShanEn	Sample Entropy
ShanEn	Shannon Entropy
SNR	signal-to-noise ratio
SP	systolic pressure
Sp	specificity
SQ	scalar quantizer
SR	sinus rythm
SVR	systemic vascular resistance
TWA	T wave alternans
URL	Uniform Resource Locator
USDZQ	uniform scalar dead-zone quantizer
VCG	vectorcardiography, or vectorcardiogram
WCT	Wilson's central terminal
WDD	weighted diagnostic distortion (measure)
ZOI	zero-order interpolator
ZOP	zero-order predictor
Ultrasponder	In vivo Ultrasonic Transponder System for Biomedical Applications

Part I

Introduction and Background

I do not imagine that electrocardiography is likely to find any very extensive use in the hospital. It can at most be of rare and occasional use to afford a record of some rare anomaly of cardiac action.

Augustus Desiré Waller, 1911

Chapter 1

Introduction

1.1 Motivation

As described in the Abstract, the ULTRASPONDER control unit implanted underneath the patient's skin that this research work is focused on, must be capable of handling several kinds of morphologycally different signals that are all obtained by dedicated sensors and describe different aspects of the patient's health state or organism operation. Thus, ECG signals, arterial or venous blood pressure records, measures of ventricular displacement or other related parameters, and hypothetically some other measures of diverse nature, for instance derived calculations that can be done with the mentioned sources as a starting point, have to cohabit the unit for obtaining as much information as possible about the patient's health conditions and significant heart activity performance parameters.

However, for maximizing the lifetime of the device's batteries and hence avoiding continuous resurgery or other means of re-energizing it, a matter of capital interest is to minimize the amount of information transmitted via a microwaves frequency link to the outside.

This consideration leads to the fact that the information provided by the control unit must be the most efficient as possible in terms of data provision for the physicians. A primary approach to this is that some simple signal processing, which has to be not too energy-consuming because of the energy constraints, can be done in the control unit itself, and just the most important results transmitted to the outside, instead of radiating the full range of data records acquired. The external system, intendedly a dedicated computer, can then perform a much more complex signal processing if required, since in that scenario the energy consumption is not a critical restricting parameter.

A further step in the matter allows us to think of avoiding any redundant transmission, in the sense of not dedicating resources to transmit a set of different signals if they don't contribute with any new information considered to be notable. In this direction, a theoretical study of the signals will be carried

out with the aim of finding some correlations between them that could justify the decision of transmitting reduced amounts of information, or only the most important portions of the records.

1.2 Approaches

In order to find the significance of a signal in terms of the information carried, the theoretical analysis should consist of some statistical calculations that can show how impredictible is a signal itself, and how dissimilar to cohabitant signals it is. The basic information measure employed in a wide field of engineering applications is entropy in its different forms.

Redundancies or dependence relations between the signals of interest must exist, since there is a physiological entrainment between them in their origin. Thus, we try to find mathematical indicators of these relations by means of correlation coefficients, mutual information, conditional entropy, etc.

The findings of this study suggest an appropriate communication model in which data redundancy, inherent to the signals we are considering, is exploited in order to get an efficient source coding scheme that yields high compression ratios without introducing significant distorsion in the reconstructed signal. A DPCM system is chosen for this purpose.

Furthermore, as the ULTRASPONDER control unit can provide some signal processing capabilities, though in a heavily power-constrained scenario, diagnostically meaningful analysis and even critical monitoring applications can be implemented. Typically, all high energy consuming processing and calculations will be performed by an external system, namely, a computer or medical device, connected via microwaves to the control unit, but it is useful indeed to include some basic functions which can continuously retreive and store important information about the patient's health or even inform the medical personnel if appropriate.

Implantable devices can offer this permanent monitoring during the patients' daily life. In this sense and for the aim of the ULTRASPONDER project, several aspects can be interesting, namely: basic QRS complexes detection systems, beats and arrhythmias classifiers, hemodynamic parameters monitoring, heart rate variability analysis, etc.

1.3 Structure of this Thesis

The body matter of this Thesis is structured as follows:

Part I includes this introductory Chapter and Chapter 2, intended to provide a fundamental background on the biomedical area that this work focuses on, with special emphasis on the particularities that ECG signal processing implies. In it, some anatomic and physiologic rudiments about the heart are

presented to give a solid understanding of the rest of the matter, explaining the morphology of the ECG signal, some blood pressure records and an introduction to heart rate variability.

Part II, the central matter of this Thesis, covers the methods, tools and algorithms employed in the work and proposed communication system. It comprises Chapters 3, 4 and 5.

Chapter 3 focuses on the mathematical tools based on information theory and others that we use to carry out the theoretical and statistical analysis of the relevant signals, results of which justify and set the basis for the direction taken in the communications processing performed further on. It also includes a brief Section covering the numerical computation environment used for all the calculations needed, MATLAB, and the text processing tools and typesetting system employed for producing this Thesis, LaTeX.

Chapter 4, one of the main contents of this work, deals with the ECG processing performed. In Section 4.1 we specify the sources employed to obtain the signals used throughout the work, a capital aspect as every algorithm has been tested and all processing has been done on them. Section 4.2 gives some commentaries about necessary pre-processing that must be done on the digitized ECG in order to adapt it to our purposes. Section 4.3 describes two time-domain QRS complexes detection algorithms for filtering the signal and producing treatable samples for the communications oriented postprocessing and 4.5 gives some insight about beats and arrhythmias classification systems. Evaluation of detection performance is covered in Section 4.4 and at last, 4.6 presents several different mehtods for HRV analysis.

Chapter 5 describes the closed-loop DPCM communication system employed. An introduction on general data compression is provided in Sections 5.1 and 5.2, with 5.3 covering specific direct ECG data compression schemes. The remaining Sections 5.4, 5.5 and 5.6 deal with several necessary aspects and components of a DPCM system, namely the structure and operation of the coder and the decoder subsystems, the prediction filter design and the quantization and coding. Lastly, in Section 5.7 we introduce a novel measure called CQ, compressibility quotient, which collects info about theoretical compression limits and actual performance results in a practical system.

Part III collects the results of statistical studies, signal processing and communication simulations treated along Part II, reported in Chapter 6. Discussion of these results is carried out in Chapter 7, and a conclusion to the work comes in Chapter 8.

The enclosed Appendices include a quick reference on statistics in App. A, an overview on polynomial predictors and interpolators in App. B, some results of the information measures introduced in Chapter 3 moved to App. C, and the M-code of all capital functions programmed for this work and some auxiliary routines, listed in App. D.

We should first endeavor to better understand the working of the heart in all its details, and the cause of a large variety of abnormalities. This will enable us, in a possibly still-distant future and based upon a clear insight and improved knowledge, to give relief to the suffering of our patients.

Het Tele-cardiogram
Willem Einthoven, 1906

Chapter 2

Background

In this Chapter we will cover the subjects related to the background of this Thesis. We will first introduce some rudiments of the heart's anatomy and electrophysiology, and then move to an explanation of the different types of signals that the control unit must handle as described in the Introduction (see page 3), with a main focus on the ECG since it will be our reference signal.

Section 2.1 covers all that must be known about the heart for the objectives of this work: its basic anatomy and reasoned electric behaviour, which is indispensable to understand the ECG waveform. Section 2.2 deals with the ECG signal itself, presenting the normal healthy ECG under sinus rythm and introducing some of the most common abnormalities that can be found in the ECG records, covering as well how are these records obtained. Other signals or important parameters to be handled by the ULTRASPONDER control unit, such as blood pressure and heart-rate are discussed in Sections 2.3 and 2.4 respectively.

After reading this Chapter one should be able to understand all technical terms and expressions as well as have a good insight in the cardioelectric frame in which this work is based on.

2.1 The Heart's Anatomy and Conduction System

To better understand the standard ECG waveform, it is convenient to carry out a preliminary study of the heart's electrophysiology, for which a brief introduction or review to the basic aspects of the heart's anatomy is needed. Once the reasons for which the ECG looks like it does are known, we will be in a favorable position to discern healthy ECGs from pathological or abnormal ones. With that basis, and applying on some basic knowledge of electrocardiography we would then be able to classify these abnormailities, sorting out the reason of the observed difference from a healthy ECG and trying to locate the failure in the cardiac

system. This Section describes the generalities we need to know about the human heart's anatomy in order to conduct our later study on its electrical activity registered in the ECG.

2.1.1 Rudiments of Anatomy. The Heart and the Circulatory System.

The heart is a pulsating biological pump that carries out the mission of keeping the blood flux circulating through the blood vessels at the necessary pressure and flow at any moment. There are four chambers forming it, two upper and smaller called atria[1], and two lower, bigger, and more muscled, called ventricles, pairwise connected by the *atrioventricular valves* (tricuspid and mitral, between the right and left atrium and ventricle, respectively).

Being the heart's task to keep the blood flowing correctly at all times all through the body blood vessels, a brief description of the course followed by the blood is convenient.

The blood flows uninterruptedly entering the heart through the right atrium, coming from the superior and inferior *vena cavae*, which collect the deoxygenated blood that comes from the cells of the whole body. From there it is pumped down to the right ventricle passing through the *tricuspid valve*, from where it leaves the heart through the *pulmonary semilunar valve*, directed to the *pulmonary artery*. The blood then reaches the lungs, where it is purified by a passive process of gas exchange called diffusion: the carbon dioxide is dropped off whereas oxygen is picked up. After that, the blood returns to the heart, entering the left atrium coming from the *pulmonary veins*. It then travels to the left ventricle flowing through the *mitral valve* mentioned above, and is furtherly pumped through the *aortic semilunar valve* to the *aorta*.

The aorta forks dividing the blood between major arteries which supply the upper and lower body. The blood travels in the arteries and to the smaller *arterioles*, then finally to the tiny *capillaries* which feed each cell. In the combustion processes that take place in the cells, the oxygen carried by the blood is partially exhausted, while carbon dioxide is produced. The deoxygenated blood, polluted with this CO_2, then travels to the *venules*, which coalesce into veins, then to the inferior and superior venae cavae and finally back to the right atrium where the journey began.

Figure 2.1 shows the ramifications of the human circulatory system, including arteries, arterioles, capillaries, venules, veins, and of course the heart. Figure 2.2 shows schematically the anatomy of the heart, depicting the elements mentioned above and some others that we will shortly refer to. Note that the left ventricle (right side of the figure) is bigger than the right ventricle. It is furthermore stronger, due to its mission of pumping the blood to all body parts, in comparison to the less demanding task of pumping blood to the lungs,

[1] Singular, atrium. Plural, atria, although atriums can also be accepted. From the Latin, *ātrium*, entry hall or open area in the center of an ancient Roman home. The atria of the heart are also called the auricles, from the Latin *auricula*, meaning little ear, for their resemblance to a dog's floppy ears. [1]

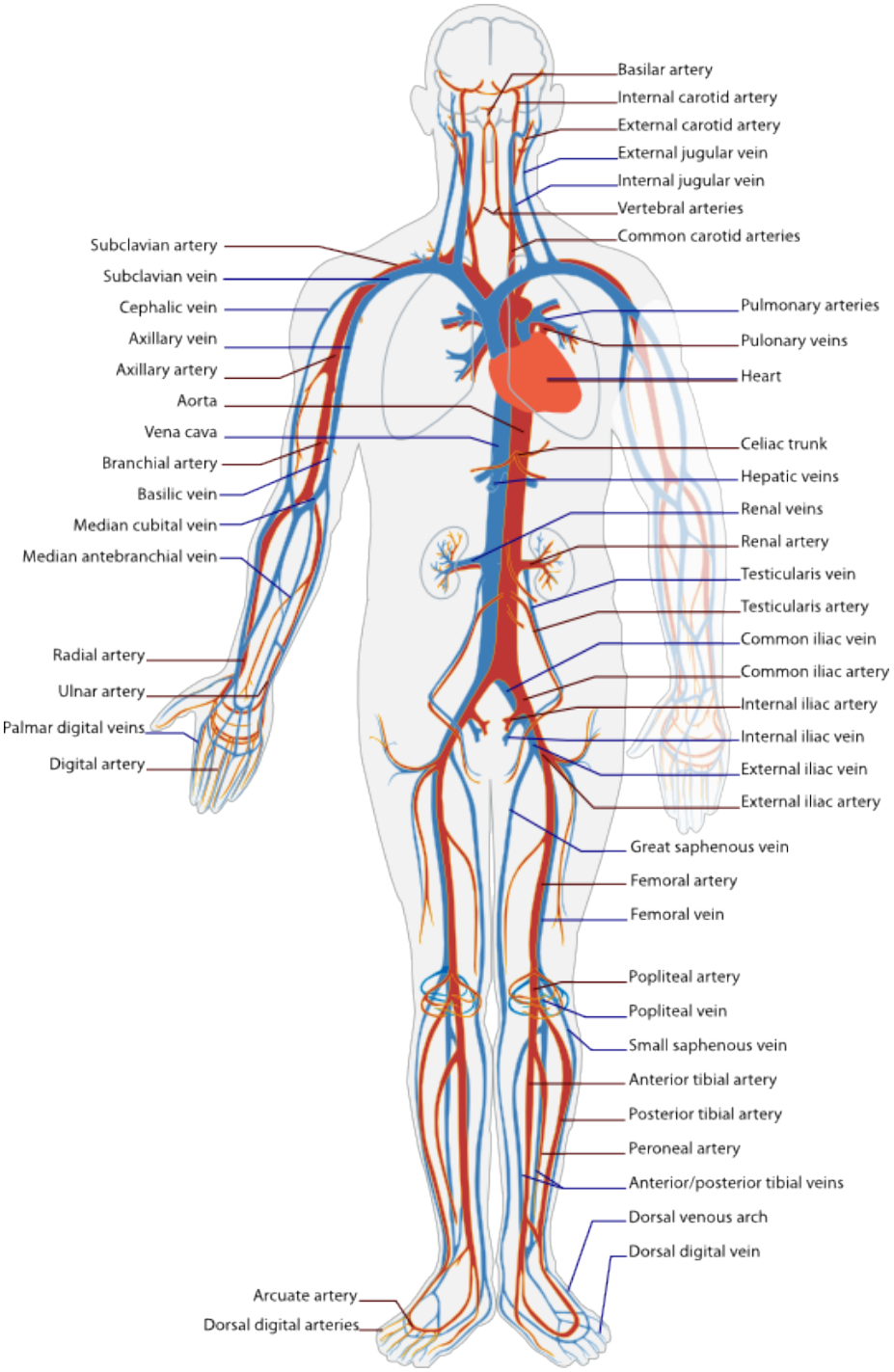


Figure 2.1. Simplified human circulatory system. Arterial system, whose mission is transporting oxygenated blood to cells all over the body, is depicted in red. Venous system, transporting deoxygenated blood polluted with CO_2 is depicted in blue. Reproduced from: [2].

acomplished by the right ventricle. As a result, the left lung is smaller than the right lung.

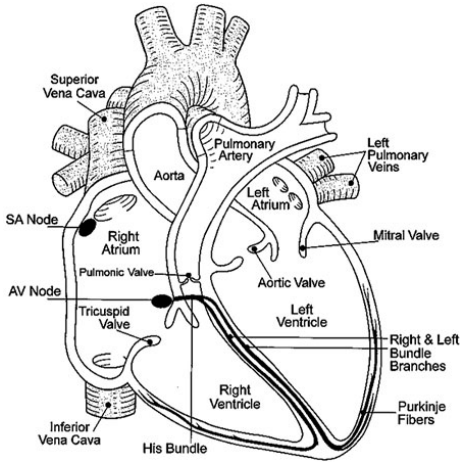


Figure 2.2. Basic heart anatomy showed schematically. Reproduced from: [3].

2.1.2 The Cardiac Cycle

For understanding the existing relation between the ECG waves and the cardiac activity, it is now pertinent to explain the phases of the *cardiac cycle*, which is the succession of the events related to the blood flow or pressure occuring from the beginning of one heartbeat to the beginning of the next ([4]). These events are traditionally ordered, and hence, the cardiac cycle divided, in two stages: *diastole* and *systole*.

When the diastole starts, the semilunar valves (pulmonary and aortic) close, and the AV valves (tricuspid and mitral) open, so that the blood flows rapidly from the atria to the ventricles, thanks to a pressure difference that the valves maintain. In this stage the whole heart is relaxed. When the ventricles are almost full, a small amount of blood still enters the heart coming directly from the veins. This substage is called *diastasis*. Then, the *atrial systole* occurs: the atria contract so that the last blood they contain flows to the ventricles. After that, the diastole is finished and the systole, second stage of the cycle, takes place.

During the systole, there is an *isovolumic ventricular contraction*, this is, without a noticeable variation in volume, while the AV valves close to avoid any blood regurgitation to the atria. The pressure in the ventricles grows rapidly until it reaches a point in which the pulmonary and aortic valves are opened, moment in which the *ejection* starts and the blood is pumped to the pulmonary artery and the aorta. The emptied flux decreases exponentially ([5]) and the pressure in the ventricles diminishes consequently. The ventricles keep contracting until the blood is totally pumped out, and at the same time the atria, in a relaxation phase, are filled with blood. The ventricles then go into the phase of *isovolumic relaxation* and the closure of the semilunar valves

prevent the blood to travel back to the ventricles, which would otherwise occur due to the fact that the pressure in them is lower than the arterial pressure. When the decrease of the ventricular pressure is enough, the AV valves open, completing the cycle and allowing the blood filling the atria to be pumped down again. Under normal circumstances, each cycle takes approximately one second.

Thus, and just to clarify, the atrial systole (contraction) takes place in the end of the heart diastole, while the ventricles are relaxed. And it is during the systole of the ventricles, being the atria in relaxation, that the blood is pumped out of the heart. Commonly, the heart systole is also referred to as ventricular systole.

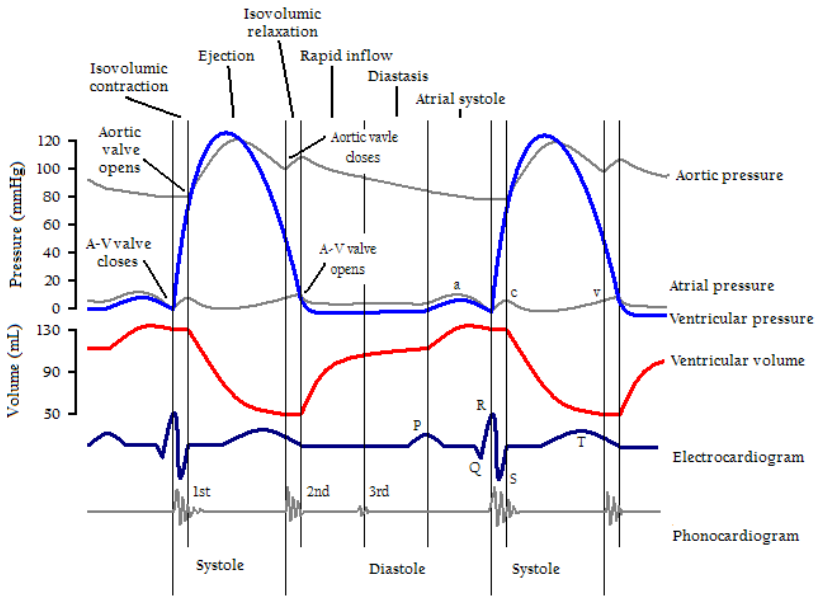


Figure 2.3. Cardiac cycle events occuring in the left ventricle. Two complete cycles are illustrated. The alignment of the different signals allows to see the relation between them. Aortic, atrial and ventricular pressure, ventricular volume, ECG and PCG are displayed. Reproduced from: [2].

Throughout the cardiac cycle, the blood pressure increases and decreases as we have remarked. But this is not the unique event to be recorded. The closure of the different valves produce some noises, similar to clicks or snaps, than can be heard by auscultation. The first sound, or S1, results from reverberation within the blood associated with the sudden block of reverse blood flow due to closure of the AV valves, i.e. mitral and tricuspid, at the beginning of systole. The second sound, or S2, is caused similarly by the sudden block of reversing blood flow due to closure of the semilunar valves, i.e. pulmonary and aortic, at the end of systole (beginning of diastole).

Figure 2.3 shows the cardiac cycle events occuring in the left ventricle, depicting these changes in aortic, atrial and ventricular pressure mentioned above, the derived sounds recorded in the phonocardiogram (PCG), and the changes in ventricular volume. An ECG signal is superimposed; we will learn

shortly why it has that shape and why each observed wave coincides with each of the phases of the cardiac cycle.

Note that the observation of these parameters - pressure measures, volume changes measures, ECG recording, and auscultation or phonocardiogram recording, can all shed light on a correct healthy heart behaviour, or on the other hand, reveal some kind of malfunctioning or faulty performance, since they are all flags of particular events or stages in the cardiac cycle that should operate in an interleaved fashion in order to ensure an optimal performance.

2.1.3 The Heart's Electrical Conduction System

The heart is mainly constituted by a muscular tissue referred to as *cardiac muscle* or *myocardium*[2], which has some particular facts that are important to know, since they allow the heart's operation.

This muscle is composed by short fibers that exhibit cross striation, sharing some similarities in terms of structure, mechanism of contraction, metabolism, etc., to both skeletal muscle and smooth muscle, the other two major types of muscle found in the human body, but having special properties that make it unique and different from both. Its cells are capable, apart from contracting, of propagating electric impulses, or *action potentials*, like the neurons that constitute the nerves, although this electrical propagation responds obviously to a different mechanism. Moreover, the musculature of the atria, as well as the one of the ventricles, behaves as a whole set.

The four basic properties that characterize the cardiac muscle are ([6]):

Automaticity Unlike the rest of the muscles in the body, which are ultimately controlled by the nervous system and excited by the motor nerves, cardiac muscle is able to generate its own stimulus, being the *sinoatrial node*[3] (see Figure 2.2) the normal origin of these stimuli. That is the reason why this fusiform node, located on the right atrium next to the entrance of the superior vena cava, is often referred to as the heart's primary pacemaker, or natural pacemaker.

Contractility The myocardium fibers respond with a maximal efficiency systole to the arrival of a stimulus, no matter the magnitude of the stimuli always that it exceeds the muscle excitation threshold. Unlike the other types of muscles, a greater stimulus does not yield a bigger contraction of the fibers. The cardiac muscle develops what is called myogenic contraction, meaning that it is the muscle itself that initiates the contraction, and not any external stimulus coming from active innervation.

Excitability The ability of cells to respond to stimuli is a basic function of life. The basis for the excitability of cells is their ionic distribution, and the distribution of ions and molecules is determined by transport mechanisms associated with their plasma membrane structure. The fibers in the heart

[2] From the Ancient Greek $\mu\upsilon\sigma$ (mys), muscle, and $\kappa\alpha\rho\delta\iota\alpha$ (kardia), heart.
[3] Abbreviated SA node or SAN, also called the sinus node.

show a special response to excitation that make them particular, contracting by means of depolarization and repolarization of their cell membranes.

Conductibility The heart muscle is capable of, apart from being excited and in turn contract, transmitting the electrical impulses, fact which is the basis for its performance as a whole with physiological unicity. The heart conducting system will be shortly explained in detail.

Whereas the SA node is the origin of the stimuli that triggers the contraction of the heart under normal circumstances, as stated above, all of the myocardium's cells possess the ability of generating electrical impulses that could eventually trigger cardiac contraction. The SA node normally initiates it simply because it generates these action potentials slightly faster than other areas with pacemaker capability, at a rate of approximately 60-80 beats per minute in a healthy adult human. Like all muscle cells, the cardiac myocytes[4] have *refractary periods* following contraction during which additional contractions cannot be triggered even if new stimuli arrive, so the rate imposed by the SA node governs the overall performance if operating properly. In this situation the heart operates at what is called a sinus rythm.

The fact that the cardiomyocytes constitute a whole set from the point of view of the stimulus electrical conduction, provokes that when a single cell is reached by it, the excitation is spread rapidly among all the others, utilizing for the contraction the total available energy of the whole set of cells, resulting in the high efficiency systole mentioned. This is the reason for that a stimulus of a bigger level does not result in a bigger contraction of the heart, unlike the other types of muscles, in which each fiber is independent of the rest, and hence, in the presence of a greater excitation, a bigger amount of fibers are stimulated and the visible effect is a bigger contraction. In relation to the refractary period, if during a previously triggered systole a second stimulus is received, no additional contraction will be unleashed since there is no energy available for the process in any cell of the muscle, so the event will simply be ignored.

So far, about the excitability of cardiac cells, suffice it to say that if a threshold-exceeding stimuli reaches a resting myocyte, the later will get active discharging the action potential and helping conducting the excitation along the muscle, becoming then non-excitable until the resting status is achieved again. During the diastole, the muscle progressively recovers its excitability as energy is stored in the cells. The later a new stimuli is produced, the easier it will be that it generates a full systole. It could happen that an abnormal stimuli were received quite before the end of the diastole, but late enough for the cells to be partially energyzed to be contracted; in this case, the subsequent extrasystolic contraction may result in diverse abnormalities in the ECG waveform. These will be discussed in time[5].

[4]Muscle cells. From the Greek, $\mu\upsilon\sigma$ (mys), muscle, and $\kappa\upsilon\tau o\varsigma$ (kyto), cell. Cardiac myocytes are specifically referred to as cardiomyocytes.

[5]We will learn that the ECG inspection and analysis of the relations of extrasystoles with the overall cardiac rythm is very useful to determine if a given area of the heart is beating automatically or not.

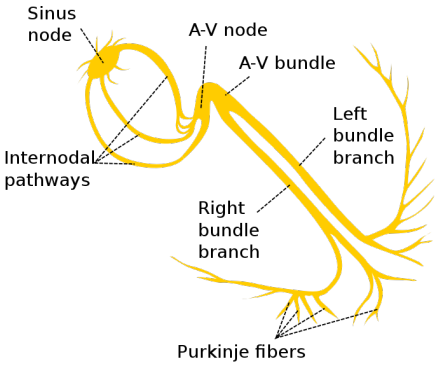


Figure 2.4. Isolated conduction system of the heart. Right and left bundle branches are the ramifications of the Bundle of His. Reproduced from: [2].

The heart's electrical conduction system can be observed in Figure 2.2 back in page 10 with the rest of the heart's main anatomic elements, or in more detail in Figure 2.4, where the rest of the elements have been deleted and only the electric conduction system with its main paths are depicted isolated.

The action potentials generated at the SA node are propagated covering the area of the right atrium throug the *internodal pathways*, causing the atria to contract in their way, until the *atrioventricular node* (or AV node)[6]. There, a delay is produced to provoke the necessary temporal difference between the atrial and ventricular systole in order to ensure not to start the ventricles contraction until the atria have been totally emptied. This delay is approximately 0.12 s.

An important property that is unique to the AV node is decremental conduction[8], which makes that the more frequently the node is stimulated the slower it conducts. This property protects the ventricles from excessively fast rate response to atrial arrhythmias, such as atrial fibrillation or flutter, to which we will refer again later. The AV node is normally the only electric connection between the atria and the ventricles.

Description of the anatomy of SA and AV nodes, as well as the particularities of the atrioventricular specialized conduction system can be found in much further detail in [9].

If by any reason the activity of the SA node was interrupted, pathologically altered, or the conduction of its electrical impulses along the internodal pathways was blocked, the automaticity mechanism of the myocardium would trigger the heart contraction even though, taking as a secondary pacemaker the AV node, which beats at a rate of 40-60 times per minute. In opposition to normal sinus rythm (see Page 13), this situation results into what is known as ectopic beats or ectopic contractions[7]. This situation can be held normally if the occurrance is occasional - it may even be triggered by overstimulation

[6] Also (rarely) referred to as the Aschoff-Tawara node, named after K.A.L. Aschoff and S. Tawara, author of the classic *Das Reizleitungssystem des Säugetierherzens* (see [7]).

[7] From the Greek, $\epsilon\kappa$ (ec), out, and $\tau o \pi o \varsigma$ (topos), place. Meaning that they arise from the wrong part of the heart muscle, when they should be generated normally in the SA node.

from some drugs such as caffeine and many other different reasons, but chronic occurrence can be life-threatening if the coordination of the muscle is lost and fibrillation occurs.

Nevertheless, the SA node is influenced by the stiumlation of the vagus or parasympathetic nerves which cause a decrease in its rate thereby decreasing the heart rate, and the sympathetic nerves causing an increase in its rate and hence increasing the heart rate. These cardiac nerves do not function triggering contractions ordered by the nervous system, because as it has been remarked before, the heart regulates autonomously its own performance (again, myogenic contractions), but they do help modulating the rythm and rate of contraction, intensifying or weakening the SAN's activity according to metabolic demand.

The AV node is electrically connected to a collection of specialized cells called *bundle of His*[8], which conduct the impulses much faster than typical cardiac fibers, forking into the three *bundle branches*: right, left anterior and left posterior bundle branches[9], that run along the *interventricular septum*[10]. The bundles give rise to thin filaments known as *Purkinje fibers*[11] that, located in the inner walls, innervate the ventricular muscle and are responsible to distribute the electric impulses to it. Because of their specialization to rapidly conduct impulses, these fibers ensure that all the cells of the ventricular muscle are excited almost in unison, providing a strong contraction carried out in a coordinated fashion, in which if everything works correctly, the left ventricle contracts slightly before the right ventricle. It takes about 0.03-0.04 s for the impulses to travel from the bundle of His to the last cells of the ventricular muscle.

Purkinje fibers also have the ability of automacity - if not overriden they generate action potentials, at a slower rate than the SA node or other ectopic pacemakers, but they can thus serve as the last resort when all other pacemakers fail. Whereas the natural resting rate of the SA node is 60-80 beats per minute (b.p.m.) and the AV node's is 40-60 b.p.m., the cells of the *bundle of His* can generate 30-40 stimuli in a minute, and the even slower cells of the *Purkinje fibers* may generate 20-30 b.p.m. if their activity is not overriden.

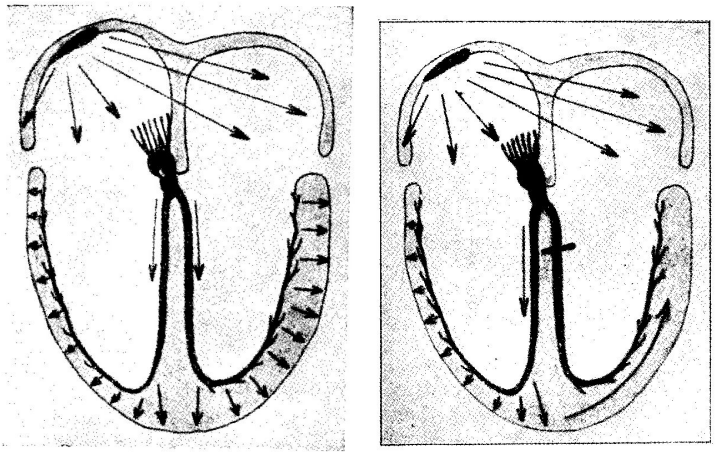
Figure 2.5 shows the path followed by the stimuli in the normal healthy case and in a *left bundle branch block*, a relatively common disease that causes one of the bundle branches to conduct the stimuli very poorly, abnormally increasing the tissue impedance and the natural delay. In that case, the stimuli have to travel along an alternative longer path in order to excite all the cells in the ventricles.

[8] Named after Wilhelm His, Jr., physician and cardiologist, not to be confused with Wilhelm His, Sr., anatomist and professor, author of many works on physiology, embriology and anatomy and inventor of the microtome.

[9] Much of the literature often distinguishes only between right bundle branch and left bundle branch, but this is not totally accurate, since the left ventricle receives the stimuli it needs for a strong and efficient contraction from the double-branched structure in which the left bundle branch further forks. These two sub-branches are alse called *fascicles*.

[10] Abbreviated IVS, is the stout wall separating the ventricles from one another.

[11] Named after Jan Evangelista Purkinjĕ, anatomist and physiologist.



(a) Normal conduction path

(b) Alternative conduction path in a LBBB

Figure 2.5. Conduction path of the stimuli in (a) the normal healthy case and (b) left bundle branch block (LBBB). Reproduced from [6].

Turning back to the issue of disorder of the stimuli generation in the SA node, these *arrhythmias* can occur as both decreases (sinus *bradycardia*[12]) or increases (sinus *tachycardia*[13]) in the normal rythm. These dysfunctions result in an abnormal heartbeat and its occurrence can be observed in the ECG, as we will study in Section 2.2.

Cardiac output is the volume of blood pumped by the heart per minute. For an average size and weight adult (70 kg) at rest this would be about 5 l/min. During severe exercise it can increase to over 30 l/min, although not in the unfit.

The decreased heart rate can cause a decreased *cardiac output* resulting in several pathological symptoms, and may also lead to ectopic beats as explained above. If ectopic pacemakers take over for a prolonged period and the heart rate falls below 35-40 b.p.m. the situation can cause complications due to a deficit of blood flow to vital organs. Sinus bradycardia is, however, not always a pathological symptom. People who regularly practice sports and whose trained hearts can pump enough blood to supply the whole body in each contraction without diminishing the oxygenation process, may have persistent sinus bradycardia, without major negative consequences[14].

On the other hand, sinus tachycardia may lead to abnormal stimuli generation in diverse locations of the myocardium which may in turn provoke extrasystolic contractions and severe complications. Other derived problem may be a fall in cardiac output too, due to the markedly reduced ventricular filling time.

[12] From the Greek $\beta\rho\alpha\delta\upsilon\kappa\alpha\rho\delta\iota\alpha$, bradykardía, heart slowness.

[13] From the Greek $\tau\alpha\chi\upsilon\varsigma\kappa\alpha\rho\delta\iota\alpha$, tachyskardía, heart acceleration.

[14] Miguel Indurain, Spanish cyclist who won the Tour de France in five successive years, had a resting heart rate of 28 beats per minute and could increase his cardiac output to 50 litres per minute and his heart rate to 220 beats per minute.

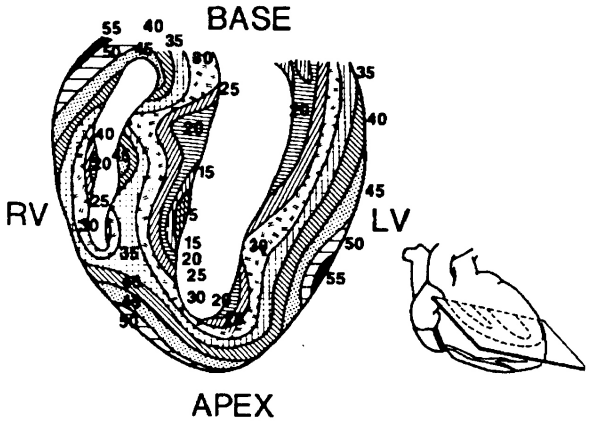


Figure 2.6. Action potential propagation isochrones in the myocardium. Reproduced from [5].

Figure 2.6 shows the action potential propagation isochrones in the myocardium for several time instants.

Summary

To summarize this first introductory Section we will briefly recapitulate the most important aspects we have discussed about the heart's anatomy and the operation of its conduction system.

- The heart's main mission is to keep the blood flowing along the blood vessels under adequate conditions. The cardiovascular system is formed by a double circuit:
  - Oxygenated blood leaves the lungs through the pulmonary veins arriving to the heart's left atrium. It flows then to the left ventricle, from where it leaves the heart through the aorta, which forks and branches into the arteries, reaching the cells around the body. The oxygen is then used in the cells for combustion processes, and the blood is loaded with CO_2 and other waste products.
  - Deoxygenated blood is collected by the veins all along the body, and led finally through the vena cavae into the right atrium, from where it is pumped into the right ventricle, and then further to the lungs through the pulmonary artery. It is purified in the lungs as the waste products are exchanged by oxygen, and then starts the journey again where it began.
- The features of the myocardium, or muscle heart, are such that it contracts triggered by stimuli that it generates itself. Moreover, whenever excited, the contraction occurs affecting the whole physiological area, unlike other kind of muscle types in which different fibers contract sepparately.

- Under normal conditions, the SA node is the area in charge of generating the electrical stimuli that trigger the heart's contraction and hence govern the heart rate. It is thus called the natural pacemaker. These stimuli, or action potentials, travel from the SA node to the AV node provoking the atrial systole (contraction), suffer a delay in the AV node that ensures the atria are fully empty, and are then rapidly conducted along the bundle of His and its ramifications onto the Purkinje fibers, which innervate and excite the ventricular muscle, provoking the coordinated ventricular systole. This situation is known as sinus rythm.
- Abnormal electrical activity in the heart is called cardiac arrhythmia. Arrhythmias can be of different kind and nature. A very simplified typology may be as follows:
  - Should the SA node fail or the conduction paths be blocked, other areas of the heart, preferably elements of the conduction system, will operate as subsidiary pacemakers and control the heart contraction. This is known as ectopic rythm, for the impulses are generated in wrong parts of the cardiac muscle. Occasional ectopic beats are common.
  - Bradycardias are slow heart rythms, less than 60 b.p.m. in rest. They rarely produce symptoms if HR does not descend below 50 b.p.m.
  - Tachycardias are fast heart rythms, more than 100 b.p.m. in rest.
- The electrical activity of the heart, which ultimately controls its contraction and hence performance, can be studied by means of the ECG. Any abnormality that occurs in it will be reflected in the ECG.

2.2 The ECG Signal

One of the extracellular records that contains more information about the functioning of a vital organ is in fact the *electrocardiogram* (commonly abbreviated ECG, or EKG[15]), which importance for clinical diagnosis is essential in a variety of ambits.

Out of the many clinically registered *extracellular biopotentials*, the ECG is the unique one that, because it possesses some more or less fixed guidelines which do not vary much from one individual to another and has a cuasi-periodic caracter, is able to offer a valuable information about cardiac pathologies by itself ([5]). Of course, there is diversity among ECGs and abnormalities occur, but due to the common characteristics that a healthy ECG should show, cardiologists are able to determine if the heart's operation is correct or not merely by observation. Another big advantage of it is that surface recordings can be obtained by means of *noninvasive techniques*. Furthermore, digitized ECGs can be computationally processed and features analyzed in order to obtain automated diagnosis. Obviously this is the foundation in which this Thesis and much related work is based on.

[15] From German, Elektrokardiogramm.

From an electric point of view, the surface ECG is simply the recording at skin level of the extracellular biosignals generated by the conduction and muscular fibers of the heart. The device used to record ECG signals is called *electrocardiograph*[16].

In this Section we will describe how ECGs are obtained, which characteristic features they possess, and how abnormalities in the conduction system of the heart can be detected by inspecting the records. Subsection 2.2.1 gives a brief overview on the history of electrocardiography, focusing on Einthoven's string galvanometer, 2.2.2 describes the obtention of ECG recordings, and 2.2.3 presents the typical ECG waveform in sinus rythm giving some data about morphology, duration and level of the different waves. Later in Section 2.6 the typical noise sources and artifacts encountered in the ECG are discussed.

2.2.1 Background of Modern Electrocardiography

Modern electrocardiographs have evolved much since the first capillary electrometers and string galvanometers were invented in the early 20th century.

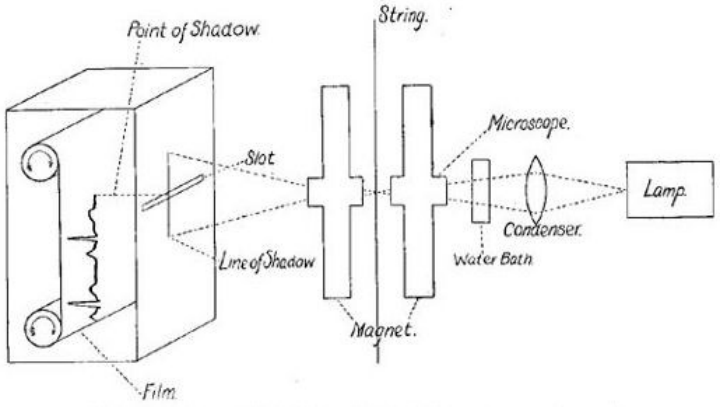
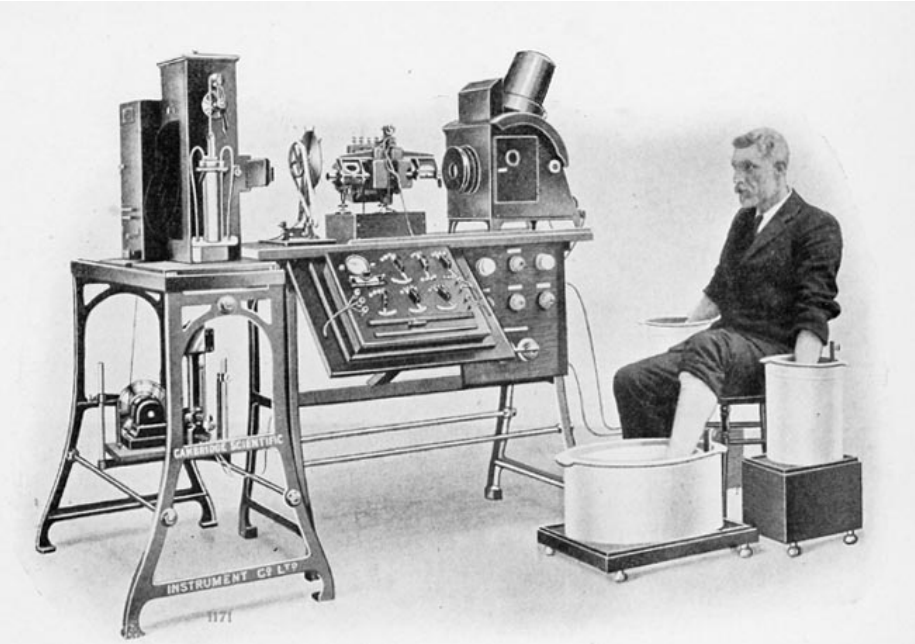


Figure 2.7. Einthoven's string galvanometer's recording scheme.

Willem Einthoven's immensurable contribution to electrocardiography allowed to accurately record human electrocardiograms by first time in 1902. His original *string galvanometer*, much more accurate than Lippmann capillary electrometer and Waller's galvanometer, required water cooling for the powerful electromagnets, needed five operators and weighed some 600 lb. Patients had to sit with both arms and left leg in separate buckets of a saline solution which acted as electrodes to conduct the electric current from the skin's surface to a silver-coated quartz filament. This filament was acted upon by powerful electromagnets positioned either side of it, which caused sideways displacement of the filament in proportion to the current carried due to the electromagnetic field. That movement was heavily magnified by a microscope and projected

[16] From the Greek, $\eta\lambda\epsilon\kappa\tau\rho o\nu$ (elektro), originally meaning amber, nowadays meaning related to electricity, $\kappa\alpha\rho\delta\iota\alpha$ (kardia), heart, and $\gamma\rho\alpha\mu\mu\alpha$ (grama) or $\gamma\rho\alpha\varphi\epsilon\nu$ (grafo), written or to write.

through a thin slot onto a moving photographic plate or film on which the point of shadow writes in a continuous curve ([10]). Figure 2.7 shows schematically this combined current-dependant electromagnetic deflection and optic system for obtaining the record of an ECG.



PHOTOGRAPH OF A COMPLETE ELECTROCARDIOGRAPH, SHOWING THE MANNER IN WHICH THE ELECTRODES ARE ATTACHED TO THE PATIENT, IN THIS CASE THE HANDS AND ONE FOOT BEING IMMERSED IN JARS OF SALT SOLUTION

Figure 2.8. Early commercial electrocardiographic device manufactured by The Cambridge Scientific Instrument Co. in the 20s decade, based on Einthoven's string galvanometer.

Einthoven obtained and published series of classic rythms and arrhythmias, describing the electrocardiographic features of many cardiovascular disorders. Figure 2.9 shows some of the first records obtained with his galvanometer. He also described a system whereby hospital recordings could be transmitted over a mile of telephone wires to his laboratory for analysis purposes[17], thus becoming one of the fathers of telemedicine with this telemetric cardiac monitoring system. He was awarded the Nobel prize in Physiology or Medicine in 1924 for his work.

In Figure 2.10 we can see the evolution of the ECG recordings from Lippmann capillary electrometer to Einthoven's string galvanometer. The upper and middle portions of this figure are from Einthoven's work *Die galvanometrische Registrirung des menschlichen Elektrokardiogramms, zugleich eine Beurtheilung der Anwendung des Capillar-Elektrometers in der Physiologie.* Archiv fur die Gesammte Physiologie des Menschen und der Thiere, 99:473, 1903. The exact

[17]Full description of the procedure in his classic paper *Het tele-cardiogram* or *Le telecardiogramme* [11], originally, The telecardiogram, 1906.

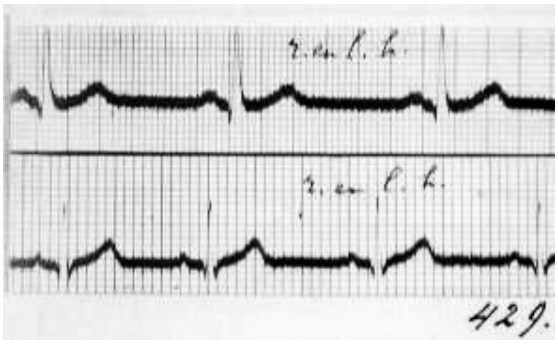


Figure 2.9. ECG recordings obtained by Einthoven's electrocardiograph.

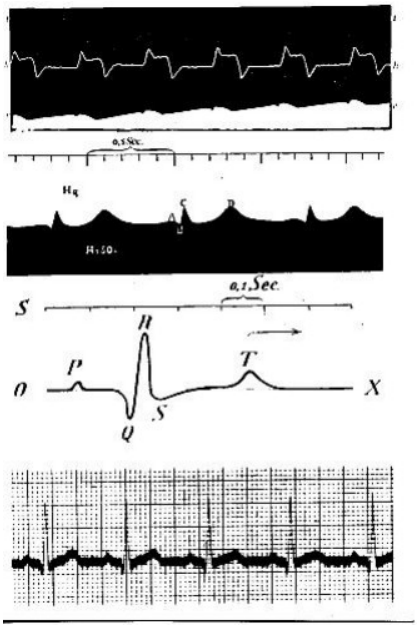


Figure 2.10. Evolution of the first ECG recordings. The upper record was made using Lippmann capillary electrometer. The middle record is a "corrected curve", mathematical model developed by Einthoven that contains more waves than previous recordings. The lower record was made using Einthoven's string galvanometer.

source for the lower portion of this figure is unknown for it was not shown in the original figure published in 1903. It did appear in Fishman A.P. and Richards D.W.'s *Circulation of the Blood: Men and Ideas*. New York, Oxford University Press, 1964, p 295. Figure 2.8 shows one of the first commercial electrocardiographic devices based on Einthoven's string galvanometer produced by The Cambridge Scientific Instrument Co.[18] in the decade of the 20s.

Though the basic principles of these devices' era are still in use today, advances have permitted a shift from cumbersome laboratory apparatus to compact electronic systems that usually include computerized interpretation of the electrocardiogram.

Nowadays, compact and portable devices can be used to continuosuly monitor the electrical activity of the heart 24 hours a day during long periods of time without causing significant discomfort to the patients. These devices are called Holter monitors[19] or ambulatory electrocardiographic monitors. One is shown in Figure 2.11.

These Holter monitors commonly employ two or three bipolar leads, named with the prefix "CM", using between five and seven electrodes placed on the patient's chest. Their extended recording period is very useful for detecting occasional cardiac arrhythmias that do not manifest during hospital monitoring and would be very difficult to identify otherwise. Periodic communication with medical personnal or storage systems are also included among their capabilities.

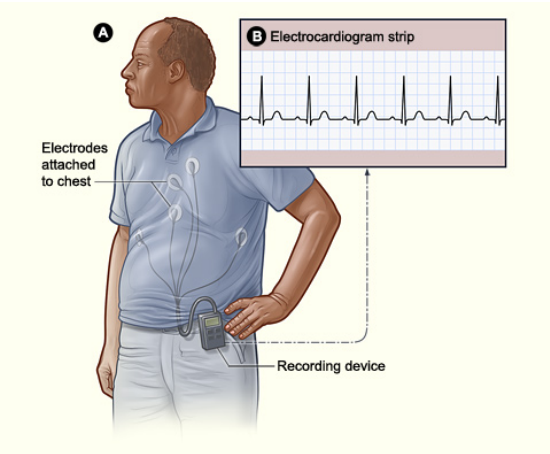


Figure 2.11. Holter monitor for continuous ambulatory ECG recording. Source: National Heart Lung and Blood Institute.

Readers interested in further information about the history of electrocardiographic devices are refered to the abundant literature on the matter (see, among others, [10, 12, 13, 14]).

[18] In the time, headed by Horace Darwin, Charles Darwin's youngest son.
[19] Named after Norman Jefferis Holter, biophysicist who invented these portable devices.

2.2.2 Surface ECG Recording: Leads

Some of Einthoven's primal methods for the recording of the ECG are still in use nowadays as we have said before. This is quite clear in the placement of electrodes for the surface ECG recording, as well as in the leads configurations.

In electrocardiography, the word *lead* refers to the electric signals transmitted and received between two electrodes. There are two types of leads: unipolar and bipolar. *Bipolar leads* are those in which one of the electrodes is considered the positive pole and the other is the negative pole. *Unipolar leads* are those which only have one true pole (the positive pole) and the reference is taken from a composite pole made up of signals from a series of other electrodes.

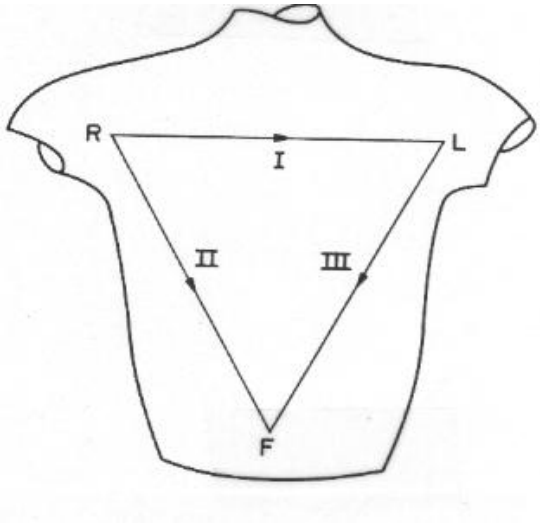


Figure 2.12. Einthoven's Triangle

Einthoven placed the three electrodes for his ECG records on the left wrist, right wrist, and left ankle of the patient, creating as shown in Figure 2.12, what is called *Einthoven's triangle*. Between them, the first three leads, called the *limb leads* (they are bipolar leads) are established as:

Lead I, the signal between the (negative) RA electrode placed on the right arm and the (positive) LA electrode placed on the left arm.

Lead II, the signal between the (negative) RA electrode placed on the right arm and the (positive) LL electrode placed on the left leg.

Lead III, the signal between the (negative) LA electrode placed on the left arm and the (positive) LL electrode placed on the left leg.

Nowadays the electrodes are set not strictly on the wrists, but along the arms, evenly on the right and left side. Same case with the ankle: the leg electrode is placed not only there, but anywhere along the leg, avoding bony prominences. The average point of these three leads is called *Wilson's central terminal* (WCT), and it approximates ground level. The law that describes the relation between these leads is:

$$I + III = II$$

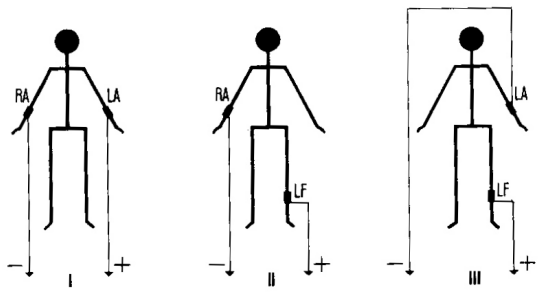


Figure 2.13. Bipolar ECG leads I, II and III. Notice that LF (left foot) is used as a synonymous of LL (left leg). Reproduced from [15].

Einthoven's triangle is still applied in modern day electrocardiography, but more electrodes and leads have been included to register a bigger amount of information. Figure 2.13 shows Einthoven's bipolar derivations I, II and III, commonly referred to as standard limb leads.

Augmented limb leads are derived from the same electrodes as limb leads I, II and III, but are intended to emphasize concrete aspects of the ECG waveform. They view the heart from different angles, or vectors (this is the basic concept for VCG, vectorcardiography), having as the negative electrode a modification of *Wilson's central terminal*. They are, thus, unipolar leads:

Lead aVR, or "augmented vector right" has the positive electrode on the right arm (RA) and the negative electrode is a combination of the left arm electrode (LA) and the left leg electrode (LL), which augments the signal strength of the positive electrode on the right arm.

Lead aVL, or "augmented vector left" has the positive electrode on the left arm (LA) and the negative electrode is a combination of the right arm electrode (RA) and the left leg electrode (LL), which augments the signal strength of the positive electrode on the left arm.

Lead aVF, or "augmented vector foot" has the positive electrode on the left leg (LL). The negative electrode is a combination of the right arm electrode (RA) and the left arm electrode (LA), which "augments" the signal of the positive electrode on the left leg.

These augmented limb leads aVR, aVL and aVF, together with the limb leads I, II and III, for the basis of the hexaxial reference system. The laws that describe the relations between the leads are:

$$aVR = -(I + II)/2$$
$$aVL = I - II/2$$
$$aVF = II - I/2$$

Figure 2.14 depicts the *augmented limb leads*, also known as Goldberger's unipolar *peripheral leads*, aVR, aVL and aVF, together with Wilson's classic approach of leads VR, VL and VF, taking the WCT point as the negative electrode.

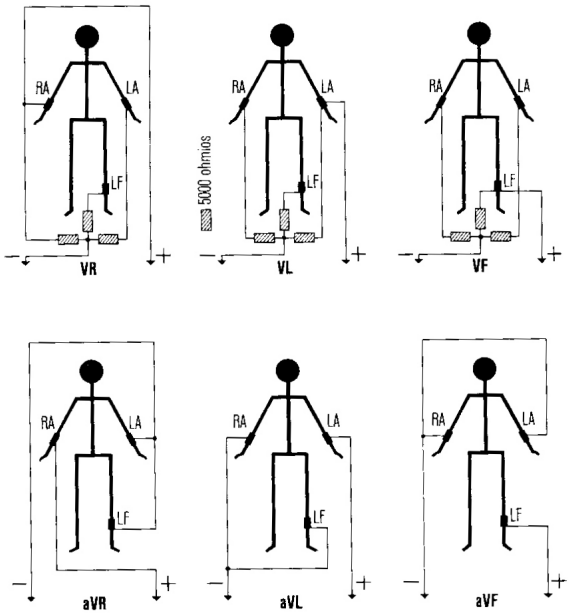


Figure 2.14. Unipolar ECG peripheral leads. Wilson's VR, VL and VF leads on the top row, and Goldberger's aVR, aVL and aVF on the bottom. Reproduced from [15].

Other six electrodes were historically introduced to complement the information obtained with electrodes RA, LA and LL. These are electrodes V1, V2, V3, V4, V5 and V6, placed directly on the chest and basis for the *precordial leads*. Because of their proximity to the heart, they do not require augmentation. The placement of the electrodes is as follows:

V1, placed on the fourth intercostal space to the right of the sternum.
V2, placed on the fourth intercostal space to the left of the sternum.
V3, placed between electrodes V2 and V4.
V4, placed on the fifth intercostal space in the midclavicular line[20].
V5, placed horizontally even with V4, but in the anterior axillary line[21].
V6, placed horizontally even with V4 and V5 in the midaxillary line[22].

Figure 2.15 shows the placement of these electrodes. From them, the *precordial leads* are defined taking each of the electrodes V1 to V6 as the positive pole, and having *Wilson's central terminal* as the negative pole.

A tenth electrode, RL, placed evenly with LL but on the right leg, is added for being used by the ECG recorder as a ground reference. As such, it plays no part in the formation of any lead.

[20]The imaginary line that extends down from the midpoint of the clavicle.
[21]The imaginary line that runs down from the point midway between the middle of the clavicle and the lateral end of the clavicle.
[22]The imaginary line that extends down from the middle of the patient's armpit.

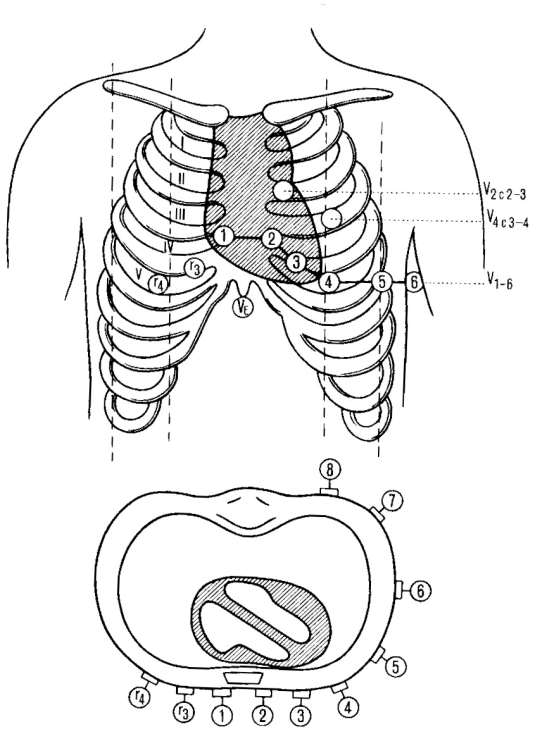


Figure 2.15. Placement of electrodes for precordial leads. Reproduced from [15].

The three limb leads, the three augmented limb leads, and the six precordial leads form together the *12-lead ECG* that is nowadays commonly recorded in hospitalary environments. Of course, for ambulatory monitorings simpler schemes such as 5-leads or 3-leads configurations are used. Each lead has a different shape and shows some details more accurately and hides others since they monitor distinct areas of the heart, but the biggest amount of information can be observed in the full 12-lead recordings. Figure 2.16 shows a 3 seconds excerpt of a typical 12-lead ECG record.

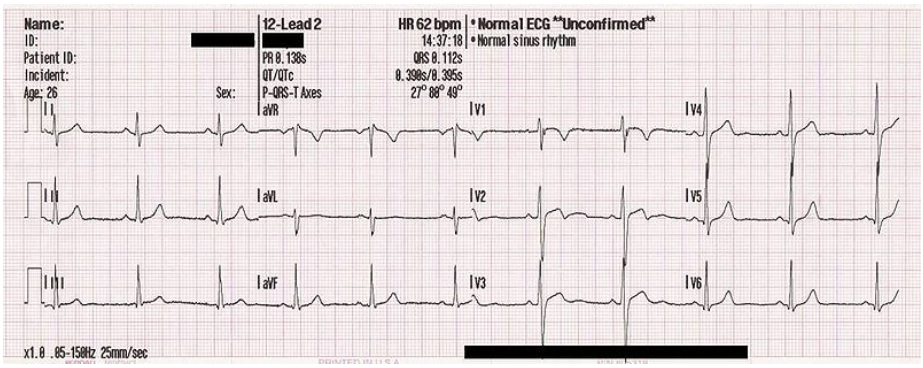


Figure 2.16. Excerpt of a 12-lead ECG of a 26-year-old male in normal sinus rythm. Reproduced from: [2]

2.2.3 Normal Sinus Rythm and Features of ECG Waves

To end this Section about the ECG signal, we present the typical waveform of the records obtained from a healthy heart operating in sinus rythm (SR).

Three major waves of electric signals appear on the ECG, each one showing a different part of the heartbeat. The first wave is called the *P wave*, and shows the electrical activity of the heart's two upper chambers (atria). The second and largest wave is called the QRS wave or *QRS complex*, since it is in turn composed by the *Q*, *R*, and *S waves*, and it shows the electrical activity of the heart's two lower chambers (ventricles). The third wave is the *T wave* and shows the heart's return to the relaxation state. They all appear on the schematic representation of an ECG shown in Figure 2.17. The baseline voltage of the electrocardiogram is known as the *isoelectric line*, typically measured as the portion of the tracing following the *T wave* and preceding the next *P wave*.

By studying the shape and size of the waves, the time between each wave, and the rate and regularity of beating, a doctor can obtain very useful information about the heart, its rhythm and health.

The *P wave* is, as said, the result of atrial depolarization. In this phase the main electrical vector is directed from the SA node towards the AV node, and spreads from the right atrium to the left atrium. Thus, the *P wave* is upright in leads II, III, and aVF (since the general electrical activity is going toward the positive electrode), and inverted in aVR (since it is going away

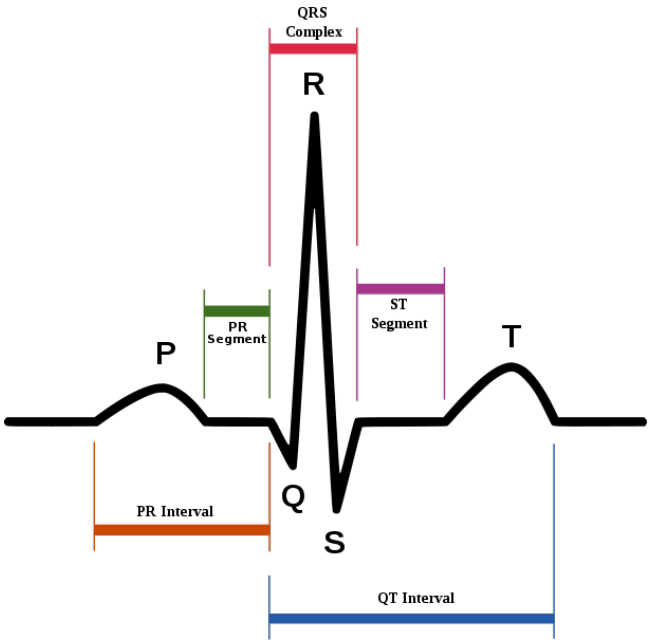


Figure 2.17. Schematic representation of normal ECG signal in sinus rythm. Reproduced from [2].

from the positive electrode for that lead). A *P wave* must be upright in leads II and aVF and inverted in lead aVR at least to designate a cardiac rhythm as *sinus rhythm*. The shape and duration of the *P waves* may indicate atrial enlargement. Its absence may indicate atrial fibrillation, whereas a saw tooth formed *P wave* may indicate atrial flutter. Weak mono or biphasic deflections without time expansion are normal and caused by light desynchronization in the atrial excitation. The maximum duration of *P wave* is 100 ms and normal amplitude is between 0.1 and 0.3 mV.

The *QRS complex* responds to the depolarization of the ventricles. As these contain much more muscle mass than the atria, the *QRS complex* is considerably larger than the *P wave*, lasting between 80 and 120 ms. A duration above 100 ms usually indicates pathological behavior. We must state that not every *QRS complex* contains a *Q wave*, an *R wave*, and an *S wave*, definitely not in every lead. But by convention, any combination of these waves can be referred to as a *QRS complex*, having different notations to facilitate correct interpretation of ECG recordings, as shown in Figure 2.18. Usually, lowercase and capital letters are used depending on the relative size of each wave and the deflection direction.

The duration, amplitude, and morphology of the *QRS complex* are very useful in diagnosing cardiac arrhythmias, conduction abnormalities, ventricular hypertrophy, myocardial infarction, electrolyte derangements, and other disease states ([16]). The amplitude of complexes vary in the different leads. Since the most important activity of the heart is the contraction of the ventricles, described in this waveform, the *QRS complex* is the quintessential element of

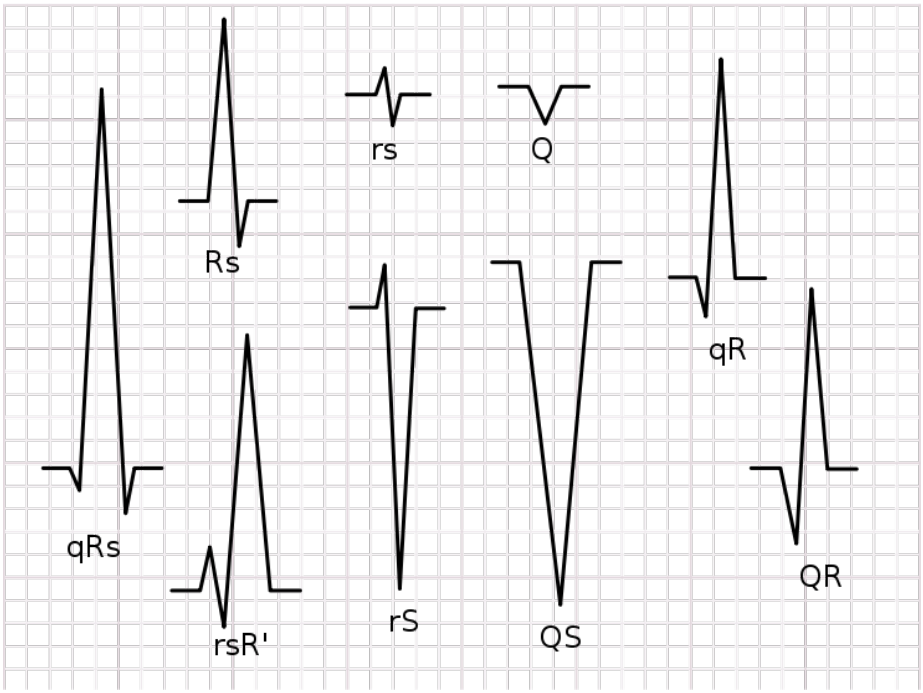


Figure 2.18. Nomenclature for different morphologies of QRS complexes. Reproduced from [2].

the ECG. Furthermore, its magnitude obscures the atrial repolarization wave, which resembles an inverse P wave but cannot be seen in normal ECGs since it is buried by the *QRS complex*, fact that makes it quite difficult to study atrial faulty operation at least in the time domain.

Q waves can be considered normal (physiological) or pathological, when they have a height of 25% or more than that of the partner *R wave* and/or have a width greater than 40 ms. Normal *Q waves*, when present, represent depolarization of the interventricular septum, and can be appreciated in leads I, aVL, V5 and V6. Abnormalities in the conduction system (for example, excessive delay in the bundle of His or Purkinje fibers) cause evidently widened *QRS complexes*.

The *T wave* represents the repolarization of the ventricles. The interval from the beginning of the *QRS complex* to the apex of the *T wave* is referred to as the *absolute refractory period*. The last half of the *T wave* is referred to as the *relative refractory period* or vulnerable period, for it is then when although not being fully recovered, the cardiomyocytes store enough energy for a premature contraction to be triggered if a stimuli is received (turn to Page 13 for explanation).

In most leads, the *T wave* is positive; however, it is negative in lead aVR. Lead V1 may have a positive, negative, or biphasic *T wave*. In addition, it is not uncommon to have an isolated negative *T wave* in leads III, aVL, or aVF. Its amplitude is commonly between 1/8 and 2/3 of that of *R wave*. Inverted *T*

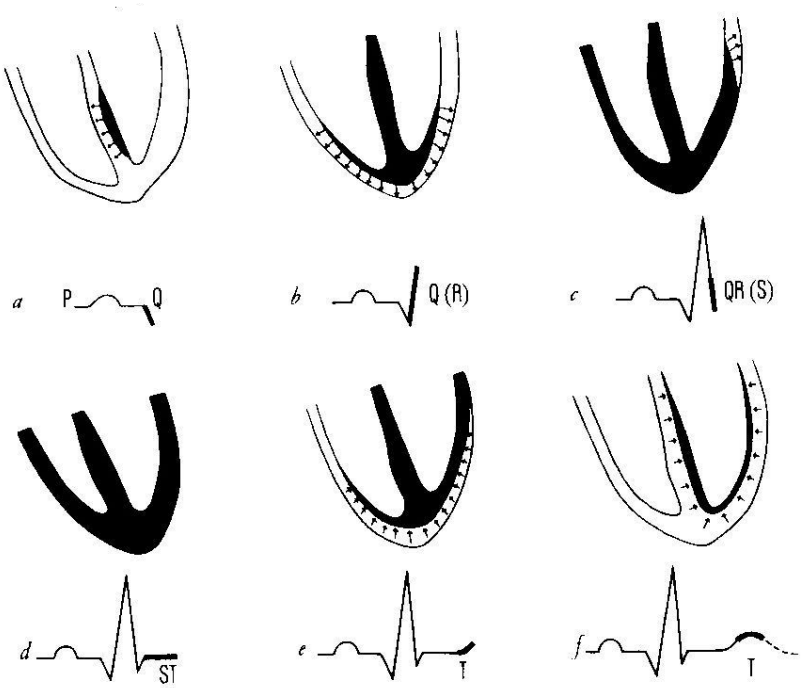


Figure 2.19. Ventricular depolarization and repolarization: formation of the QRS complex and the T wave in the ECG. Reproduced from [15].

waves when unexpected in leads, can be a sign of coronary ischemia, Wellens' syndrome, left ventricular hypertrophy, or central nervous system disorder. Tall or tented symmetrical *T waves* may indicate hyperkalemia, or even signs of acute myocardial infarction if the *T wave* is hyperacute, whereas flat *T waves* may indicate coronary ischemia or hypokalemia. When a conduction abnormality (e.g., left bundle branch block, paced rhythm) is present, the *T wave* should be deflected opposite the terminal deflection of the *QRS complex*. This is known as appropriate *T wave discordance*. In the last years the phenomenon of *T wave alternans* (TWA) has revealed to be one of the most promising clinical indexes to estimate risks of malignant arrhythmias and sudden cardiac death, and many studies have been conducted on methods to analyze TWA, such as [17].

Figure 2.19 shows the process of depolarization and subsequent repolarization of the ventricles along with the ECG registered in those moments: the *QRS complex* and the *T wave*. Atrial repolarization is buried on the *QRS complex* and cannot be noticed in the ECG.

A small *U wave* appears in some ECG recordings, after the *T wave*. *U waves*, the only remaining enigma of the ECG, are thought to represent repolarization of the papillary muscles or Purkinje fibers, but their meaning is not fully unravelled yet (see [18]). They have normally the same polarity as the *T wave* and less than one-third its amplitude. *U waves* are usually best seen in the right precordial leads especially V2 and V3. See them in Figure 2.20.

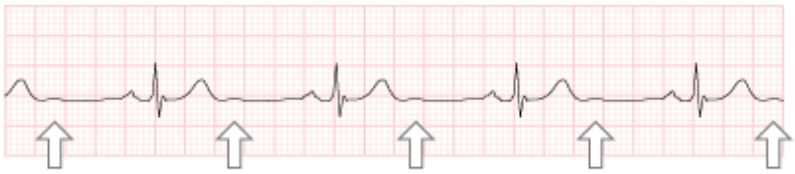


Figure 2.20. U waves seen in an ECG record

The *PR interval* describes the atrioventricular conduction time, and is defined between atrial activation onset and ventricular activation onset. It can vary between 120 and 200 ms depending on the heart rate. It tends to be an isolectric interval, so it is used as a reference line for examining the *ST segment*. The *PR segment* is measured from *P wave* offset.

The *ST segment* comprises the time between *S wave* offset and *T wave* onset and corresponds to the total ventricular depolarization phase. It can appear slightly above or below the isoelectric level depending on the lead and rate activity.

The *QT interval* covers the time of ventricular systole, and is measured from *Q wave* onset until *T wave* offset. Its duration depends heavily on the heart rate, as shown in Figure 2.21 with maximum and minimum boundaries, and can be enlarged or shortened in pathological situations. It is related with the *R-R interval* (usually taken as a measure of the duration of a cardiac cycle, although the *P-P interval* is also used), or equivalently, with the heart rate, but the dependence is complex. Classically, the next expression, known as Hegglin and Holzmann formula, was accepted:

$$QT = 0.39\sqrt{RR \pm 0.04}$$

Figure 2.21 is plotted from it. But several corrections have been proposed over the years in different studies, yet none of them providing a closed and universal valid solution for the interrelation in every case ([16, 19]).

Lastly, the *TP segment* corresponds to the diastole phase and is measured from the offset of *T wave* until the onset of the next *P wave*. Bradycardia affects the ECG in general, widening the waves and the isolectric segments, whereas tachycardia shortens them with increasing heart rate. If the heart rate is above 140 bpm, the *P wave* might get difficult to distinguish from the *T wave* of the previous heartbeat, and the *TP segment* disappears, situation which may be confused with a paroxysmal supraventricular tachycardia or atrial flutter with a 2:1 block.

Many more details can be found in classic references such as [20, 15, 6], but the insight provided in this Section broadly suffices for the objectives of this Thesis.

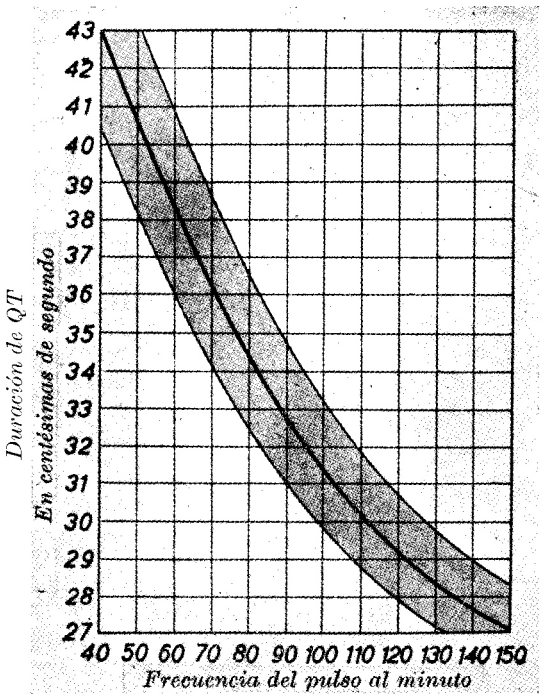


Figure 2.21. QT interval duration (vertical axis, in hundredths of second) as a function of heart rate (horizontal axis, in beats per minute). Reproduced from [6].

2.3 Blood Pressure Records

Blood pressure (BP) is the pressure exerted by circulating blood on the walls of blood vessels, constituting one of the main vital signs. We have already referred to it in Section 2.1, seeing in Figure 2.3 that maximum arterial pressure occurs at the beginning of the cardiac cycle during ventricular systole, whereas minimum pressure occurs during the diastole phase when the ventricles are filled with blood, but we now give an insight in its measurement and recording.

The pressure of the circulating blood decreases as it travels away from the heart through arteries and returns to it through veins.

The predominantly used unit for blood preasure measurement is the millimeter of mercury (mmHg), and measurement can be done by means of *non-invasive techniques* such as palpation, auscultation and others, which are simpler, quicker, less unpleasent and painful for the patient, but less accurate, or by *invasive techniques* usually involving direct measurement of arterial pressure by placing a cannula needle in an artery. Of course invasive techniques are reserved to hospitalary environments. The digitized waveforms we will use in this work all come from this type of methods.

Venous pressure (CVP, for central venous pressure), measured in a vein or in the atria, is less commonly used than arterial pressure and is usually reserved to intensive care medicine, and its common values are around 5 mmHg in the right atrium and 8 mmHg in the left atrium, whereas systolic arterial pressure (SP) moves between 90 and 119 mmHg and diastolic pressure (DP) is in the range of 60 to 79 mmHg. Occurance of lower values are called *hypotension*, whereas higher values are referred to as *hypertension*.

In relation to ECG Holter monitors described in 2.2.1, blood pressure home monitoring is also a common practice since it can help avoiding misdiagnosis and reduce the incidence of "white coat" syndroms. Continuous hemodynamic monitors have been developed over the last years providing increased capabilities and measurement accuracy for outpatient monitoring, often implanted simultaneously with or as an integrated part of pacemakers or cardioverter-defibrillators, or for research purposes ([21, 22, 23]).

Figure 2.22 shows a record from the MGH/MF Waveform Database[23] containing three ECG leads and blood pressure records along with respiratory impedance and airway CO_2 waveforms. Waveforms labelled ART, PAP and CVP represent respectively arterial pressure, pulmonary arterial pressure and central venous pressure. We can see the correspondence of the peaks and minimum values of arterial pressure and the different events of the cardiac cycle conveyed in the ECG records according to what has been explained above.

It is common to use ePAD (*estimated pulmonary artery diastolic pressure*), defined as the right ventricular pressure at the time of pulmonary valve opening, which occurs at the time of maximal dP/dt, as an approximation to actual pulmonary artery pressure, since a strong correlation has been shown to exist between them ([24]) and ePAD is easier to obtain (implantation of long-

[23] See Section 4.1.

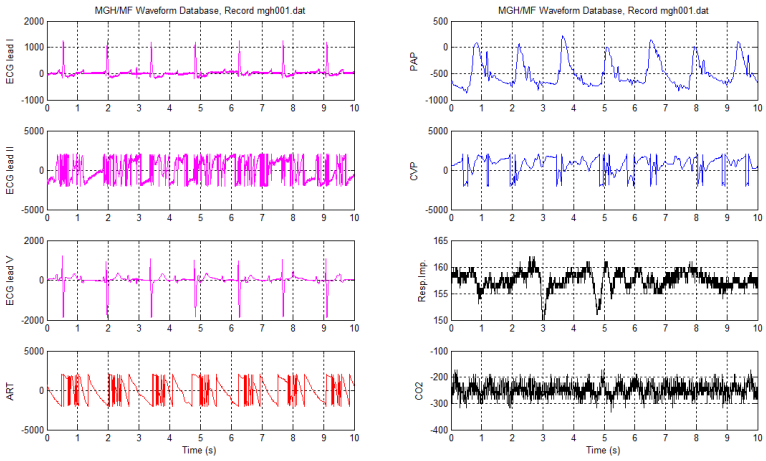


Figure 2.22. Details of blood pressure measurements in record mgh001 from the MGH/MF Waveform Database

term pressure sensors in the right ventricle as a part of a permanent pacing lead has been reported without relevant complications, whereas the potential obstacles to long-term implantation of measurement devices in the pulmonary artery are not insignificant, according to [24]. See Figure 2.23 for details.

Some straightforward calculations can be done from the digitized waveforms in order to obtain clinically relevant information. The *cardiac output* (Q, or CO), is the volume of blood being pumped by the heart, in particular by a ventricle, in a minute. It can be calculated multiplying the heart rate by the stroke volume. An average cardiac output would be 5 L/min for a human male and 4.5 L/min for a female. The *systemic vascular resistance* (SVR) is the resistance to flow that must be overcome to push blood through the peripheral circulatory system. Average values are 900-1200 dyn·s/cm^5, or equivalently, 90-120 MPa·s/m^3. From these two parameters, the *mean arterial pressure* (MAP) over a cardiac cycle is defined as:

$$MAP = (CO \cdot SVR) + CVP$$

being CVP the central venous pressure. MAP can be estimated from measurements of the systolic and diastolic pressures while the heart rate is constrained to normal range as:

$$MAP \approx P_{dias} + \frac{P_{sys} - P_{dias}}{3}$$

The pulse pressure can be simply calculated from the difference:

$$P_{pulse} = P_{sys} - P_{dias}$$

In this Thesis we will analyze the underlying relations between blood pressure records and ECG signals by means of several mathematical tools. This physiologically originated entrainment has a significant repercussion on the information shared by the different waveforms.

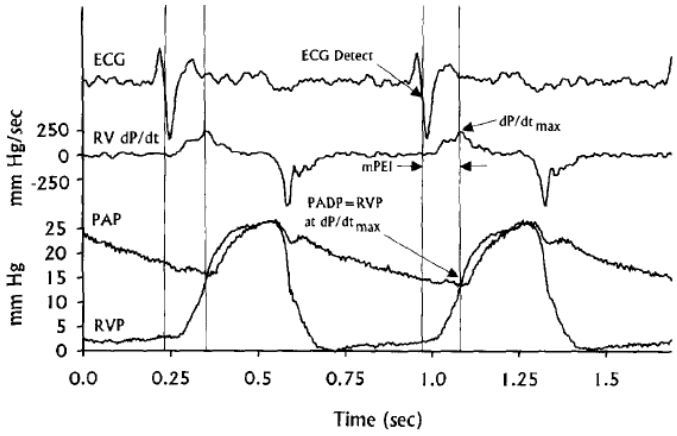


Figure 2.23. Schematic illustration of QRS complex detection, maximal first derivative of right ventricular (RV) pressure (dP/dt_{max}) and modified preejection interaval (mPEI) and the hypothesis that pulmonary artery diastolic pressure (PADP) equals right ventricular pressure (RVP) at maximal dP/dt. Reproduced from [24].

2.4 Heart Rate

The *heart rate* (HR) is simply the frequency with which the cardiac cycle occurs. Implied in its definition is the fact that it of course adapts, just as the operation of the heart does, to changes in the body's need for oxygen, thus being slower during rest or sleep periods and turning faster in exercise, moments of excitation or other situations that require a big blood supply, as ordered by the autonomic nervous system (ANS). However, this variation is gradual to a certain extent under normal circumstances.

It can be seen in Figure 2.3 that the duration of a cardiac cycle may be measured by determining the instants in which a variety of events take place on several signals of different nature which are all entrained, but the most common and easiest way of measuring it in automated ECG and heart rate monitors is by calculating the *R-R period*, defined as the interval between subsequent maxima of *R waves* in the ECG signal.

The measurement of *R-R intervals* is shown in Figure 2.24, and *heart rate* is defined as its inverse:

$$\text{HR} = \frac{1}{\text{Duration of R-R Interval}} \tag{2.4.1}$$

HR is usually given in beats per minute (b.p.m. or bpm). As stated in Section 2.2.3, the normal resting HR is around 60 bpm.

Heart rate variability (HRV) is a very interesting parameter to study. While the rhythmic beating of the resting heart was in the past believed to be monotonously regular, we nowadays know that the rhythm of a healthy heart is

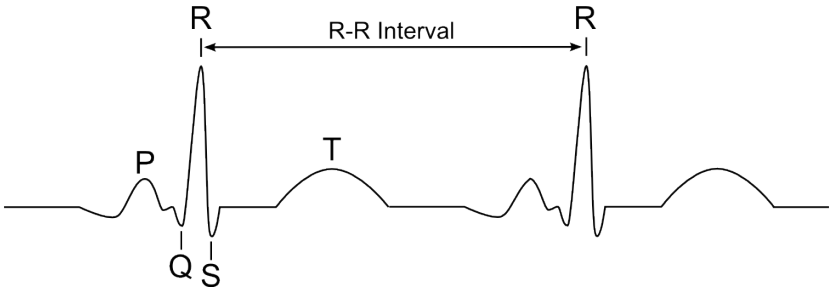


Figure 2.24. Measurement of R-R intervals

actually surprisingly irregular, even under resting conditions and no apparent situations of stress or emotional alteration ([25]. Also worth to review: [26]). These moment-to-moment variations in *heart rate* are easily overlooked when the average rate is calculated over a long period by simply dividing the number of beats per the time interval considered.

In [25], this explanation is provided: "The normal variability in heart rate is due to the synergistic action of the two branches of the ANS, which act in balance through neural, mechanical, humoral and other physiological mechanisms to maintain cardiovascular parameters in their optimal ranges and to permit appropriate reactions to changing external or internal conditions. In a healthy individual, thus, the heart rate estimated at any given time represents the net effect of the parasympathetic (vagus) nerves, which slow heart rate, and the sympathetic nerves, which accelerate it. These changes are influenced by emotions, thoughts and physical exercise. Our changing heart rhythms affect not only the heart but also the brain's ability to process information, including decision-making, problem-solving and creativity. They also directly affect how we feel. Thus, the study of heart rate variability is a powerful, objective and noninvasive tool to explore the dynamic interactions between physiological, mental, emotional and behavioral processes."

There are several general effects that provoke changes in the HR ([27]):

i) respiratory sinus arrhythmia mediated by respiration, and controlled by parasympathetic activity. This activity is responsible for variations of the HR in 2-5 seconds intervals;
ii) the blood pressure regulation, which contributes to HRV in 10 second rythms;
iii) sympathetic system activity, which induces changes with a periode above 20 seconds;
iv) variations in the minutes and hours range are influenced by the neurohumoral oscillations in the circulating blood, by circadian rythms or rapid eye movement phases during sleep.

Apart from natural variation due to the ANS activity, arrhythmias cause abnormalities in the heart rythm and rate, accelerating or slowing it down, and tachycardia is considered to be a HR of more than 100 bpm and bradycardia one of less than 60 bpm.

Due to its big interest as a significantly accurate mortality predictor, HRV has been widely studied as an indicator of the entrainment between brain and heart, as well as of cardiac performance after severe cardiac diseases and operations. A number of different methods and approaches has been proposed for measuring it, some of which will be discussed in Section 4.6.

2.5 Bioelectric signals features

The process of acquiring signals from the real, continuous world and transforming them into digitized samples, is commonly not conceived as an isolated task, but it has an associated signal preprocessing aimed to unmask every significant detail contained in the signal under consideration to make it available and ready for the actual analysis or processing desired.

In the context of biomedical signals this is even more significant if possible, since these signals gather some characteristics that make this preprocessing absolutely necessary if some coherent analysis is going to be derived. Firstly, the magnitude of bioelectric signals is intrinsically small, and often comparable to that of the noise that can affect them. In second place, in this kind of acquisition there is usually a considerable amount of equipment involved, fact that ensures an important amount of interference that could lead to measure errors. Also, in hospitals and laboratories there are abundant electric appliances and secondary equipment not directly related with the acquisition process, but that can also contribute to the impurity of the records retrieved. Last, but not least, the frequency range of biomedical signals can span over a relatively big interval in which significant noise sources can interfere.

Table 2.1 shows some of the characteristic features of the most commonly employed bioelectric signals. We see that the ECG signal has an approximate dynamic range of 1 - 10 mV and that its frequency range spans over 0.05 - 100 Hz. Note that this means that any other signal or noise of comparable magnitude whose spectral content overlaps or falls into the ECG's, and can somehow interfere the measure by entering through the electrodes or making itself present in some other way, will affect the quality of the record.

This issue is discussed in more detail in the following Section, which describes the specific noise and interference sources that affect the ECG signal.

Because the ECG signals retrieved from the public databases (see Section 4.1) have been mostly obtained in a real hospital context they are all subject to the considerations described above. This, in terms of digital processing, means that they are affected by different artifacts that can hide relevant information to the processing system. Thus, a previous conditioning step is required before using them in an analysis with warranties as we will discuss in Section 4.2, where we will study the methods used to prevent these interferences from having an impact on the behaviour of ECG processing systems.

Classification	Acquisition	Frequency Range	Dynamic Range	Comments
Action potential	Microelectrodes	100 Hz - 2 kHz	10 μV - 100 mV	Invasive measurement of cell membrane potential
Electroneurogram (ENG)	Needle electrodes	100 Hz - 1 kHz	5 μV - 10 mV	Potential of a nerve bundle
Electroretinogram (ERG)	Microelectrode	0.2 - 200 Hz	0.5 μV - 1 mV	Evoked flash potential
Electro-oculogram (EOG)	Surface electrodes	dc - 100 Hz	10 μV - 5 mV	Steady corneal-retinal potential
Electroencephalogram (EEG)				
Surface	Surface electrodes	0.5 - 100 Hz	2 - 100 μV	Multichannel (6-32) scalp potential
Delta range		0.5 - 4 Hz		Young children, deep sleep and pathologies
Theta range		4 - 8 Hz		Temporal and central areas during alert states
Alpha range		8 - 13 Hz		Awake, relaxed, closed eyes
Beta range		13 - 22 Hz		
Sleep spindles		6 - 15 Hz	50 - 100 μV	Bursts of about 0.2 to 0.6 s
K-complexes		12 - 14 Hz	100 - 200 μV	Bursts during moderate and deep sleep
Evoked potentials (EP)	Surface electrodes		0.1 - 20 μV	Response of brain potential to stimulus
Visual (VEP)		1 - 300 Hz	1 - 20 μV	Occipital lobe recordings, 200 ms duration
Somatosensory (SEP)		2 Hz - 3 kHz		Sensory cortex
Auditory (AEP)		100 Hz - 3 kHz	0.5 - 10 μV	Vertex recordings
Electrocorticogram	Needle electrodes	100 Hz - 5 kHz		Recordings from exposed surface of brain
Electromyography (EMG)				
Single-fiber (SFEMG)	Needle electrode	500 Hz - 10 kHz	1 - 10 mV	Action potentials from single muscle fiber
Motor unit action potential (MUAP)	Needle electrode	5 Hz - 10 kHz	100 μV - 2 mV	
Surface EMG (SEMG)	Surface electrodes			
Skeletal muscle		2 - 500 Hz	50 μV - 5 mV	
Smooth muscle		0.01 - 1 Hz		
Electrocardiogram (ECG)	Surface electrodes	0.05 - 100 Hz	1 - 10 mV	
High-Frequency ECG	Surface electrodes	100 Hz - 1 kHz	100 μV - 2 mV	Notchs and slus waveforms superimposed on the ECG

Table 2.1. Features of the most commonly used bioelectric signals. Note the small level they all have in general, in the range of tenths of μV to hundreds of mV. Also note the wide frequency range covered by the different signals.

2.6 Noise and interference sources present in the ECG

In order to know how to design the preprocessing system and which tasks to accomplish in this previous step it is necessary to know which are the actual artifacts (noise sources or interferences) that affect the ECG, which are their origins, and if it is actually possible to avoid or remove them. A commonly accepted classification for these artifacts is as follows ([17]):

Physiologic origin noise, subclassified in:

- **Muscular activity.** The activity of the muscles which are close to the electrodes is also captured by them. Thus, the ECG is affected by the EMG[24] with transient interferences in the order of milivolts and typically between 0 and 10 kHz. This effect can be reduced by trying to keep the patient steady while the record is being done, but this is not always possible, like in Holter monitoring (see page 22) or in stress tests, which necessarily imply dynamic movement.
- **Noise due to the movement of the electrodes.** If the electrodes are not properly fixed or the patient moves, two different kinds of variations appear, as a consequence of the change of impedance in the interphase skin-electrode:
  - **Baseline shifting.** Slow variations in the baseline level mainly due to the patient breathing an derived alterations. This produces a phenomenon similar to a soft AM modulation (0.15 - 0.3 Hz).
  - **Movement artifacts.** Quick variations produced by harsh movements of the electrodes. These produce severe level shift of the baseline with exponential falls. The effect can be smoothed reducing the skin's impedance and using special electrodes.

External origin interferences which can be of diverse nature:

- **Line induced interference.** Because of the severely "hostile" environment in which ECG digitation is usually done (electrical appliances, equipment, lighting...), this is typically an important factor. It is worth to mention that different countries have different line frequencies. Whereas in Europe and most countries of Asia and Africa the frequency is 50 Hz, in the majority of North and South America it is 60 Hz[25].
- **Radioelectric interferences.** Produced by equipment commonly found in hospitalary environments such as electric bisturies. These interferences are in the high-frequency range, so they can be deleted by an adequate filtering scheme.

Electronic noise introduced by the acquisition equipment itself. These noise sources are within the acquisition system so no screening approaches are valid; only a good design of the equipment (isolation, noise control, appropriate choice of components, etc.) can help to reduce their effect.

[24] Electromyogram. Muscular electric activity.

[25] Japan is a special case, for it has a mixed electric system: the Eastern regions (Tokyo, Kawasaki, Sapporo, Yokohama, and Sendai) use a line frequency of 50 Hz, whereas the Western regions (Okinawa, Osaka, Kyoto, Kobe, Nagoya, Hiroshima) operates at 60 Hz. See `http://en.wikipedia.org/wiki/Mains_power_systems` for further reference and details on power systems all around the world.

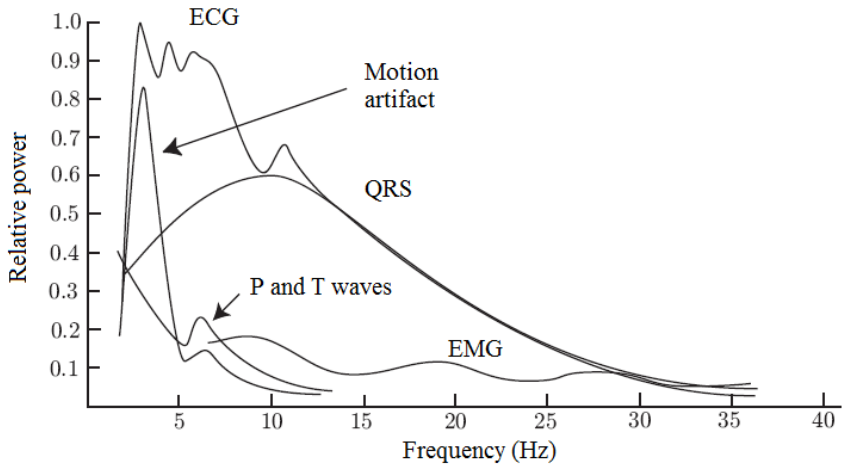


Figure 2.25. Power spectral distribution of the normal ECG waveform and main artifacts. Reproduced from [31].

Figure 2.25 shows the PSD of the main artifacts along with the ECG characteristic waves and the overall ECG waveform. The frequency location of the artifacts makes clear that band-pass filtering is a basic and fundamental step for any ECG signal processing. Also, the relatively distinct location of atrial and ventricular electric activities can make it possible to study them sepparately. In this Thesis we will use this advantage to select the information of QRS complexes as explained in Section 4.3, but other filtering schemes can be equally applied to study different parts of the ECG, in special, atrial activity ([28, 29, 30]).

The baseline artifacts due to appearence of respiratory patterns in the ECG led to the idea that the records might carry significant information of two kinds: electrical cardiac activity and respiration-induced modulation. Methods for derivating respiratory signals from ECG recordings, useful for apnea identification, have actually been developed for applications in which the ECG, but not respiration, is routinely monitored, as presented in [32, 33].

Some literature ([34, 35]) is completely devoted to the study of origins and effects of these noises, and covers a wide quantity of methods dedicated to minimize their presence in the ECG. A good example is [36]. In this Thesis we will focus on the software algorithms employed, inviting the interested reader to consult the cited References for further details.

Part II

Methods and Communication Model

> Those who assert that the mathematical sciences say nothing of the beautiful or the good are in error. For these sciences say and prove a great deal about them; if they do not expressly mention them, but prove attributes which are their results or definitions, it is not true that they tell us nothing about them. The chief forms of beauty are order and symmetry and definiteness, which the mathematical sciences demonstrate in a special degree.
>
> Aristotle from Estagira, *XIII.1078a33*

Chapter 3

Mathematical Analysis and Development Tools

This first introductory Chapter of Part II mainly provides a brief review of the mathematical elements of information theory that we need to know in order to analyze the relations and dependencies among the signals we are considering in this work. Short remarks are given in Section 3.1, and readers interested in more details are referred to Shannon's classic work: [37].

We also provide a quick reference in statistics in Appendix A, which constitutes a short compilation of some of the basic statistics applied in the vast field of engineering, obviously focusing on the parameters used in this work. Readers without a solid knowledge of statistics are recommended to review it shortly, before reading this Chapter or the results presented in Chapter 6.

Section 3.2, that closes this Chapter, has been included to cover a mention to the development tools utilized to carry out the research, simulations, tests and calculations that the work summarized in this Thesis comprises, as well as the environments employed to produce the results and the document of the Thesis itself.

3.1 Mathematical Analysis

3.1.1 Information and Entropy

Based on Shannon's information theory, we will use in this work several magnitudes to measure the amount of information contained in a signal, or the information that one signal gives provided another, or the mutual dependence exsting in between them.

In this sense, we will consider discrete signals as numerical sequences that represent the outcome of random variables and apply all the mathematical theory developed over these concepts.

First, the concept of *information* itself must be introduced. Even intuitively, one can think that the more uncertain is the outcome of a random variable, namely, an uncommon symbol composed by different values of bits transmitted along a communication channel, the occurance of an earthquake in a region in which no quakes usually occur, or the appearence of the combination "kzj" in English written text[1], the bigger information the message contains, and opposedly, if the outcome of a random variable is likely to be predicted with accuracy, it contains less information.

Mathematically, this is called *self-information*, which is a measure of the information content associated with the said outcome of a r.v. and depends solely on the probability distribution of the variable: the smaller the probability of an event, the larger the *self-information* associated with the fact of that event happening. It is thus a measure of the unexpectability of an event, or of the "surprise" that the event produces, in opposition to its expectability, likelihood or probability of appearence in redundant data sources.

Let us consider a discrete random variable X which can take a series of values $x_1, x_2, .., x_N$, each of which with a probability $p(x_1), p(x_2), .., p(x_N)$ described by its probability mass function. The *self-information* $I(x_i)$ associated with outcome x_i is defined as:

$$I(x_i) = \log\left(\frac{1}{p(x_i)}\right) = -\log\left(p(x_i)\right) \tag{3.1.1}$$

For binary variables the logarithms are calculated with base 2 and the unit of information is the *bit*. If natural logarithms are used (base e) the unit is the *nat* and for decimal logarithms (base 10), it is the *hartley*.

The *entropy* $H(X)$ of the d.r.v. X is the expected value of its self-information over its range of possible values:

$$H(X) = E_X\left[I(X)\right] = \sum_{x \in X} p(x) I(x) \tag{3.1.2}$$

[1] Many studies have been done over the years to analyze entropy and redundancy of different languages, using as dataset huge collections of modern and classic novels. It is a known fact that letter "E" is the most common in general English speech. Of course this is a particularity of English, and is not shared by other European languages.

Introducing in 3.1.2 the definition of $I(X)$ given in 3.1.1 and considering the range of the variable X, we can rewrite:

$$H(X) = \sum_{i=1}^{N} p(x_i)I(x_i) = -\sum_{i=1}^{N} p(x_i)\log p(x_i) \tag{3.1.3}$$

In the case that some of the events is absolutely unlikely to happen, $p_j = 0$ for some j, the value of the corresponding term is assumed to be 0, which is consistent with:

$$\lim_{p\to 0^+} p\log p = 0$$

An important property of *entropy* is that its value is maximized when all events in the event space (or range of the variable in the discrete case) are equiprobable:

$$p(x) = 1/N \quad \forall x_i \quad \Rightarrow \quad H(X) = \log N$$

Moreover, if X is a binary discrete random variable, as it is the case of the result of a bit transmitted over a channel, or the result of a coin toss, and we call p to the probability of getting one result and $q = (1-p)$ the probability of the opposed result, the binary entropy function becomes:

$$\begin{aligned} H_b(X) = H_b(p) &= -p\log p - (1-p)\log(1-p) \\ &= -p\log p - q\log q \end{aligned} \tag{3.1.4}$$

If the d.r.v. is furthermore unbiased, the maximum value of entropy $H(x) = 1$.

Note that the entropy is not always an integer number, not even for the discrete case it is an integer number of bits.

Suffice it to say in this point that the entropy of an information source, that is to say, the uncertainty over its probability distribution, is a capital parameter to know how easy or difficult it can be to compress the information generated by that source. The more redundant the source is, the easier it will be for compression methods to exploit this lack of "surprise" about the upcoming data.

3.1.2 Other Quantities of Information

If *entropy*, that is, the information content of a random variable, points how easily message data can be compressed, one would like to have a measure of how to find the communication rate accross a channel by analyzing the common information shared between two random variables.

This measure is called *mutual information*, and it expresses the amount of information that can be obtained about one random variable by observing another with which the former shares some dependency.

Before giving a mathematical definition, we need to define two other concepts: *joint entropy* and *conditional entropy*.

The *joint entropy* $H(X,Y)$ of two discrete random variables X and Y is merely the entropy of their pairing: (X,Y). This implies that if X and Y are statistically independent, their *joint entropy* is the sum of their individual *entropies*.

$$H(X,Y) = E_{X,Y}\left[I(X,Y)\right] = -\sum_{x,y} p(x,y)\log p(x,y) = H(Y,X) \tag{3.1.5}$$

The *conditional entropy* (or *equivocation*) $H(X|Y)$ of d.r.v.X, given d.r.v. Y, quantifies the remaining uncertainty of X after observing the values of Y. It is the result of averaging (expected value of) $H(X|Y=y)$ over the range of possible values of Y:

$$\begin{aligned} H(X|Y) = E_Y\left[H(X|y)\right] &= -\sum_{y\in Y} p(y)\sum_{x\in X} p(x|y)\log p(x|y) = \\ &= -\sum_{x,y} p(x,y)\log\left(\frac{p(x,y)}{p(y)}\right) \end{aligned} \tag{3.1.6}$$

This can be rewritten in terms of entropies as:

$$H(X|Y) = H(X,Y) - H(Y) \tag{3.1.7}$$

Then, given the information revealed by the observation of variables X and Y at the same time, $H(X,Y)$, and the uncertainty observed in X for concrete realizations of Y, $H(X|Y)$, the *mutual information* (or *transinformation*) between X and Y, noted $I(X;Y)$ measures the amount of information that can be obtained about X by observing Y, and is defined as:

$$I(X;Y) = E_{X,Y}\left[SI(x,y)\right] = \sum_{x,y} p(x,y)\log\left(\frac{p(x,y)}{p(x)p(y)}\right) \tag{3.1.8}$$

where $SI(x,y)$ is the *specific mutual information* and measures the association between variables X and Y. Rewriting the expression in terms of entropies:

$$\begin{aligned} I(X;Y) &= H(X) - H(X|Y) = \\ &= H(X) - H(X,Y) + H(Y) = \\ &= H(Y) - H(Y|X) = I(Y;X) \end{aligned} \tag{3.1.9}$$

Intuitively, being $H(X)$ a measure of uncertainty about the r.v. X, then $H(X|Y)$ is a measure of what Y does not clear up about X, which is the "amount of uncertainty remaining about X after Y is known". Thus, $I(X;Y)$ equals to the "amount of uncertainty in X which is removed by knowing Y", which of course, provided that there is a certain relation of dependency between X and Y, is the same as the "uncertainty in Y removed by knowing "X", making $I(Y;X) = I(X;Y)$.

Thus, we can estimate the mathematical relation that exists between a pair of signals by computing these measures of the information revealed about one by observing the other, or the information that the pairing provides, etc.

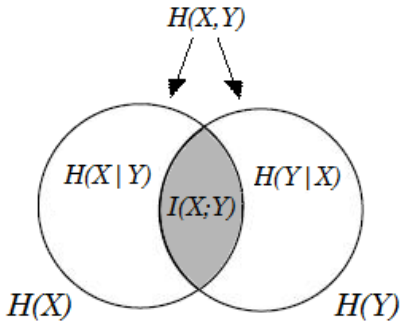


Figure 3.1. Information diagram showing the relations between entropies and mutual information.

Figure 3.1 shows the Venn diagram summarizing the relations among these concepts for two random variables X and Y. Marginal *entropies* $H(X)$ and $H(Y)$ are represented as sets, which union is the *joint entropy* $H(X,Y)$ and which intersection is the *mutual information* $I(X;Y)$. *Conditional entropies* are calculated by subtracting the appropriate marginal *entropy* from the *joint entropy*, being $H(X|Y) = H(X,Y) - H(Y)$ and $H(Y|X) = H(X,Y) - H(X)$.

3.1.3 Estimation of Shannon Entropy and Other Complexity Measures

Over the years, researchers have developed many methods and measures to estimate the complexity of time series, based on the concept of Shannon's entropy and other information and sequences distance measures. The study of heart rate dynamics is a good example of the many fields in which these approaches have been applied, in addition to the methods explained in Section 4.6.

In [38] and other works by the same authors, we can read a multiscale approach to quantifying the complexity of physiological time series (MSE, multiscale entropy analysis), while [39] and [40] use sample entropy to estimate HR variability. Approximate entropy (ApEn) is studied in [41]. In [42] a model based on stochastic feedback is presented. [43] describes estimation of Shannon entropy by recurrence plots involving computation of Lyapunov exponents.

In [44] we find estimation methods for marginal, joint and conditional entropy rates, oriented to linguistics as well as biomedical applications. The authors of [45] use a Lempel-Ziv complexity measure as a base to their ventricular tachycardia detection system.

The amount of literature covering ECG, EEG and other physiological time-series complexity is very abundant. Readers interested in these dinamic studies are referred to the specific literature provided or general reviews on the matter.

3.1.4 Correlation

In several points of this work, we need to analyze the similarities between two discrete sequences or sets of numbers, and more concretely, if there exists a linear relationship between them. Several coefficients measure the degree of correlation, that is, the strength and direction of that hypothetical dependence.

To determine such a possible relation between signals $x(t)$ and $y(t)$, and how big might this relation be, the *cross-correlation coefficient* ρ_{xy} is employed, defined as:

$$\rho_{xy} = \frac{Cov(x,y)}{\sigma_x \sigma_y} \tag{3.1.10}$$

where σ_x and σ_y are respectively the variances of $x(t)$ and $y(t)$, and $Cov(x,y)$ is the *covariance* of signals $x(t)$ and $y(t)$.

By employing the *autocovariance* of a signal, one can examine how similar is the signal in comparison to time-shifted versions of itself, computing the *autocorrelation coefficient* $\rho_{xx}(\tau)$ as:

$$\rho_{xx}(\tau) = \frac{K_{xx}(\tau)}{\sigma_x^2} \tag{3.1.11}$$

where σ_x is again the variances of $x(t)$ and $y(t)$, and $K_{xx}(\tau)$ is the autocovariance of $x(t)$.

3.1.5 Kurtosis

In probability theory and statistics, *kurtosis*[2] is a shape measure of the peakedness or bulging of the probability distribution of a real-valued random variable. It studies the frequency concentration around the mean and the central area of the distribution. Higher kurtosis means more of the function variance is due to infrequent extreme deviations, as opposed to frequent modestly-sized deviations for the case of low kurtosis.

Classically, *kurtosis coefficient* was defined as:

$$\beta_2 = \frac{\mu_4}{\sigma^4} \tag{3.1.12}$$

where μ_4 is the fourth moment about the mean and σ is the standard deviation.

Nowadays, kurtosis is more commonly calculated dividing the fourth cumulant by the square of the second cumulant, which can be expressed as:

$$\gamma_2 = \frac{\kappa_4}{\kappa_2^2} = \frac{\mu_4}{\sigma^4} - 3 \tag{3.1.13}$$

where the minus 3 is added for correction so that the kurtosis of the normal distribution is equal to zero. This parameter is often referred to as *excess kurtosis*.

[2] From the Greek $\kappa\upsilon\rho\tau o\varsigma$ (kurtos), bulging.

The *k-th moment about the mean*, or *k-th central moment* of a real-valued random variable X is defined as:

$$\mu_k = E\left[(X - E[X])^k\right] \tag{3.1.14}$$

Note that the zero-th central moment μ_0 equals 1, the first central moment μ_1 equals 0, and the second central moment μ_2 is the variance, usually noted σ^2. See Appendix A for further reference.

In the application to ECG signals, kurtosis provides a useful statistical information: a higher value indicates less frequent but bigger variations in the signal, such as could be the case of normal sinus rythm, whereas a low value of kurtosis shows that the variations are many more but of smaller level, which could be the case of a tachycardic episode or even atrial flutter. Thus, it can be a capital parameter in monitor systems and arrhythmia classification if computed in real-time.

For calculating the kurtosis of a data set of N samples we will use:

$$\gamma_2 = \frac{\mu_4}{\sigma^4} - 3 = \frac{\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^4}{\left(\frac{1}{N}\sum_{i=1}^{N}(x_i - \bar{x})^2\right)^2} - 3 \tag{3.1.15}$$

3.2 Development Environments

3.2.1 Numerical Computation and Signal Processing

All the mathematical and statistical analysis, and signal processing comprised in the work that this Thesis covers has been done in MATLAB.

MATLAB is a numerical computing environment as well as a programming language, invented by Cleve Moler and developed by The MathWorks, specialized in matrix manipulation, plotting of functions and data, implementation of user designed algorithms, creation of graphic or textual interfaces, etc. It is widely used for a variety of very different applications in the engineering field, since a big number of "toolboxes" including programs, functions and scripts for specific applications are developed by The MathWorks and offered as part of the software, and also programmed, packaged and shared by personal users.

The version of the environment employed was 7.4.0 (R2007a).

The implementation in MATLAB code of the algorithms employed in this work are included in Appendix D for consult purposes.

3.2.2 Text Processing Tools

Although it has little to do with the practical realization itself of the engineering problems approached and tried to solve covered in this Thesis, we believe that it

is also important to mention, though briefly, which tools have been used in the redaction of the actual document of the Thesis and maybe justify the reasons for that election, for this process of redaction is also a relevant and hard part of any scientific work, and specially in the case of long documents such as thesises, that ultimately determines much of its flavor. That is the aim of the present Subsection.

For several reasons such as easiness in the management of structured documents, high quality in the equations edition system, gratuitousness, lack of dependance in propietary text processing suites, almost universal compatibility, widespread application in the scientific community and thesises writing context, and, in general, appreciation for high quality text composition systems, this Thesis has been redacted using LaTeX.

LaTeXis a document markup language and document preparation system created by Leslie Lamport for the TeXtypesetting system created by Donald Knuth. Its current version as of the date of writing this Thesis is LaTeX 2_ε. The distribution employed has been MiKTeX 2.7 for Windows and the text editor, TeXnicCenter, a free and open source IDE specially designed to work with MiKTeX. The PDF viewer chosen was Adobe Reader 8.

BibTeX, created by Oren Patashnik and Leslie Lamport, was used as the reference and bibliography management software, being also one of the major attractives for having decided to write the Thesis in LaTeX.

Several sources have been consulted as reference material about the usage of TeXcommands, LaTeXcapabilities and BibTeXformats, many from the Internet in the form of FAQ compilations, specialized forums, blogs, etc., and some in the shape of documentation and books. Some of the latest have been included in the References Chapter ([46, 47, 48, 49]). Of course, we have also consulted the manuals of some of the packages[3] employed, specially the ones not included in the standard MiKTeX distribution. It is worth mentioning that this Thesis is the very first contact of its author with the LaTeXsystem.

Below there is a list of some of the not-so-common packages employed in this work, with brief accompanying descriptions of what each of them is used for.

caption Package for typesetting full-custom captions in which the user can control styles, fonts, delimiters, margins, placement, etc.

epigraph Relatively cryptic package for typesetting epigraphs. It has been used for attaching interesting quotations related to the matter at the beginning of each Chapter.

makeidx Standard package for creating indexes. It uses the MakeIndex program. It has been used in this Thesis to produce the alphabetical Index of terms found in the end of the work.

[3] Packages in LaTeXare sets of code that can be included in a document in order to make available some commands and features that would not be otherwise. Also, user created packages help extending LaTeXstandard capabilities. Any package can be added to the distribution if not included by default, and then loaded in a document or project by issuing a specific command.

moreverb Extension of the `verbatim` package that provides verbatim transcriptions, usually for typesetting code, with some additional facilities. It has been used in this Thesis to transcript the code in Appendix D.

nomencl Package for producing lists of symbols or acronyms, using the capabilities of the MakeIndex program. Other packages such as `acronym`, `glossaries` or `glosstex` have similar functionalities. It has been used in this Thesis to produce the abbreviations list found in the frontmatter.

paralist Package that redefines the listing environments, giving the possibility of typesetting compact lists, lists as paragraphs, or lists embedded inside paragraphs. Simply cosmetic use.

pdflscape Package that adds PDF support to the environment `landscape` of package `lscape`, allowing the user to compose sideways pages to show rotated elements. Specially useful to include long tables which are wider than long, and fit much better in a landscape than in a portrait orientation.

todonotes Package that lets the user mark things to do later, in a simple and visually appealing way, fully customizable. This information can be viewed as margin notes. It also allows to print a list of all the to-do notes included in the document. It is indeed very useful in the redaction of long and complex documents in which the user may be working in several different sections simultaneously, or for marking pendant things to review, rephrase, etc.

url Allows correct typesetting and hyphenation of URLs.

– As for you, my galvanized friend, you want a heart. You don't know how lucky you are not to have one. Hearts will never be practical until they can be made unbreakable.
– But I still want one.

Dialogue between the Wizard of Oz and the Tin Woodsman, from the film *The Wizard of Oz*, 1939

Chapter 4

ECG Processing

In this Chapter we will present the various methods and algorithms utilized in the signal processing that the work exposed in this Thesis comprises.

As we discussed in Section 2.2, the ECG is a fundamental signal with a significant relevance for clinical diagnosis. Computerized processing of digitized ECG signals allows a variety of different applications. For instance, just to mention some: *i*) detection of ventricular late potentials in intervals commonly considered to be isoelectric can be a sign of defective repolarization ([36]); *ii*) classification of isolated abnormal heart beats and comparison to known healthy patterns can lead to identification of abnormal waveform morphologies which can in turn be sign of life-threatening pathologies; *iii*) QRS complexes detection can serve as the basis for the automated calculation of the heart-rate, and its variability over time; or *iv*) atrial fibrillation is an arrhythmia of which many features are still unknown and evolution cannot be predicted by mere visual inspection of ECG recordings, but ECG samples processing can provide data about the fibrillation rate, spectral components, etc., which can be used to determine the nature of the episode. As we will see in the following sections, the QRS complex is the fundamental wave for almost all automated ECG analysis.

All the different applications and aims require specific processing approaches. For the purposes of the ULTRASPONDER Project, we will focus on:

- Implementing a QRS complexes detection system, which will identify and enclose each cardiac beat.
- Building an alignment scheme which provides an accurate reference point in each beat and allows for additional joint analysis.
- Designing a characterization system for the detected complexes that allows classifying them and detecting abnormal beats.
- Complementary processing such as noise and interference reduction to increase the accuracy of all other processes.

- Additional postprocessing, such as heart rate calculation for monitorization purposes.

4.1 Signals Sources

The signals employed for the analysis and all the processing done in this work have been obtained from the PhysioBank Archive[1], which is a large collection of "well-characterized digital recordings of physiologic signals and related data for use by the biomedical research community", freely offered in its website to download and use. Altogether with PhysioToolkit and PhysioNet, it is part of the Research Resource for Complex Physiologic Signals "created under the auspices of the National Center for Research Resources of the National Institutes of Health" as described in [50].

PhysioBank Archive compiles many types of different databases, including biomedical signals representative from both "healthy subjects and patients with a variety of conditions with major public health implications".

Out of the enormous signals collection offered by PhysioBank, the following sets categorized in the Multi-Parameter Databases and ECG Databases were considered for their interest in this work[2]:

MGH/MF Waveform Database: Collection of 250 recordings each containing 3-lead ECGs, ABP, PAP, CVP, respiration, and airway CO_2 signals from patients in critical care units; some recordings include intra-cranial, left atrial, ventricular and intra-aortic pressure waveforms. The records represent a broad spectrum of physiologic and pathophysiologic states. Individual recordings vary in length from 12 to 86 minutes, and in most cases are about an hour long.

Fantasia Database: ECG and respiration recordings two hours long each, with beat annotations from 20 young and 20 elderly subjects, all healthy, in sinus rhythm during a resting state while watching the movie *Fantasia* (Disney, 1940) to help maintain wakefulness. Half of the recordings also include (uncalibrated) continuous noninvasive blood pressure signals.

MIT-BIH Arrhythmia Database: Collection of 48 half-hour excerpts of 2-channel ambulatory ECG recordings, obtained from 47 different subjects. Twentythree of these recordings were chosen at random from a set of 4000 24-hour ambulatory ECG recordings collected from a mixed population of inpatients (about 60%) and outpatients (about 40%); the remaining twentyfive recordings were selected from the same set to include less common but clinically significant arrhythmias that would not be well-represented in a small random sample.

MIT-BIH Noise Stress Test Database: Collection of 12 half-hour ECG recordings and 3 half-hour recordings of noise typical in ambulatory ECG

[1] `http://www.physionet.org/physiobank/`.

[2] Descriptions are taken from the specified entries in PhysioBank Archive.

recordings. The noise recordings were made using physically active volunteers and standard ECG recorders, leads, and electrodes. The three noise records were assembled from the recordings by selecting intervals that contained predominantly baseline wander, muscle (EMG) artifact, and electrode motion artifact. Electrode motion artifact is generally considered the most troublesome, since it can mimic the appearance of ectopic beats and cannot be removed easily by simple filters, as can noise of other types

MIT-BIH ECG Compression Test Database: Unannotated database collecting 168 short ECG recordings (20.48 seconds each) selected to pose a variety of challenges for ECG compressors, in particular for lossy compression methods.

Some of these databases have been employed just in the training phase of our algorithms, whereas others have also been used for experiments performance reports, mainly the MIT-BIH Arrhythmia Database and the MGH/MF Waveform Database.

4.2 Noise and Interference Reduction

Unless otherwise specified in the concrete documentation, most signal collections publicly available have been acquired and digitized in the U.S.A. and hence the power-line induced interference occurs at a frequency of 60 Hz, unlike the case of hospitalary environments in Europe in which this fundamental frequency is 50 Hz as explained in Section 2.6.

This is indeed the case of records obtained from the MIT-BIH Databases and others described in the previous Section. Thus, the rejection filters designed must take this fact into account.

The ECG PSD including common artifacts, shown in Fig. 2.25, suggests that band-pass filtering might be enough to reduce noise and interference levels significantly in most applications in which the overall ECG has to be considered, as it is the case in our work.

Baseline drifts can also be cancelled easily either by high-pass filtering, or by envelope subtraction, usually preferred.

For the purposes of our processing, different approaches will be used for noise and artifacts removal, usually specifically designed in accordance to the concrete detection or classification system. Readers interested in different methods are suggested to see the literature referred to in Section 2.6.

4.3 Detection of QRS Complexes

4.3.1 Introduction

Since the QRS complex is the most characteristic waveform set of the cardiac cycle, reflecting the electrical activity of the heart during the depolarization of the ventricles, the duration of its waveform as well as its shape can provide very important information about the heart's performance. Therefore, an essential component of any ECG signal processing system is the block performing QRS onset and offset detection.

This is a complex problem due to the variability in morphology of the waveform itself, both in healthy and abnormal ECGs, as well as the presence of interferent sources overlapping in the same frequency span (see Figure 2.25 in Section 2.6). Moreover, P and T waves can occasionally appear altered in magnitude and frequence, resembling the QRS complex and running the risk of being confused and missidentified.

As a result of the complexity of the problem, many different approaches have been proposed over the years. In [51] we find a general classification of QRS detection algorithms sorted into three categories: *i*) *non-syntactic*, employing techniques from classical signal processing such as matched filters, threshold triggers, template correlation, frequency domain analysis, etc. as well as heuristic techniques; *ii*) *syntactic*, employing techniques from the syntactic pattern recognition field; and *iii*) *hybrid*, which approach the detection as a mixture of the non-syntactic and syntactic methods, often incorporating techniques from the artificial intelligence field.

Also, a comprehensive description of the principles underlying different approaches can be found in [52], where algorithms are classified as *i*) based on signal derivatives and digital filters; *ii*) wavelet-based, including approaches with filter-banks; *iii*) neural network approaches; and *iv*) additional methods, including adaptive and matched filters, hidden Markov models, genetic algorithms, techniques based on mathematical morphology operators, other transform-based detectors, syntactic methods, and *maximum a posteriori* (MAP) estimation methods. Evaluation and comparison of performance of the different algorithms is as well provided.

It is important to state that all algorithms, implemented in hardware or software, described in the literature fall in one of two big groups ([53]): either they focus on stablishing a very accurate reference point, only possible in highly favorable acquisition conditions where a high signal-to-noise ratio is achieved, or else they try to detect the QRS complex without emphasizing on precision in the definition of the onset, permitting detection in less favorable situations.

In this Thesis we will focus on *non-syntactic detectors*, which are capable of attaining a greater processing speed due to their bigger structural simplicity, and hence are applicable in agreement to the real-time constraints of Ultrasponder. Readers interested in *syntactic detectors* are referred to the review done in [54]. A genetic algorithm for QRS detection is described in [55]. A

MAP estimation scheme and a detector system based on it are presented in [56]. A detector based on the dyadic wavelet transform is described in [57], and [58] analyses and compares several QRS detection algorithms based on first-derivative schemes. For a general overview and some insights on many different methods, readers may turn to [52]. Even, in a relatively recent study, [59], a less common beat detection algorithm for pressure signals was presented, based on finding the location of percussion peaks in intracranial pressure signals and the systolic peak in arterial blood pressure and pulse oximetry signals. This is an interesting approach, exploiting the characteristic waveforms of some pressure signals and using a bank of bandpass filters and several preprocessing techniques worth to furtherly develop.

4.3.2 Description of the Detection Systems

Most software non-syntactic QRS detectors follow the algorithmic structure showed in Figure 4.1 ([52]). There are two stages performing different tasks: in first place there is a *preprocessing feature extraction stage*, including *linear filtering* and some kind of *nonlinear transformation*, and in second place there is a *decision stage* that involves *peak detection* and a set of *decision logic*. This block diagram will help us understand the function of each part of the proposed algorithms.

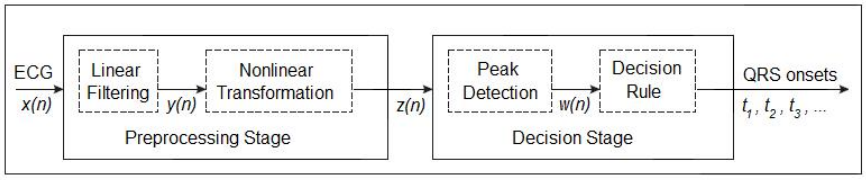


Figure 4.1. Block diagram of a QRS complexes detector. The system receives the input of ECG signals, and provides the estimated instants of presence of QRS complexes as output.

The mission of the *preprocessing stage* is providing a signal, $z(n)$ that is more suitable for application of the decision rule than the ECG signal itself, $x(n)$.

Typical frequency components of QRS complexes range from about 10 Hz to about 25 Hz. Therefore, almost all QRS detection algorithm use a *bandpass filtering* scheme prior to the actual detection in order to attenuate other signal components and other artifacts and noise sources that interfere at higher frequencies. This required bandpass filtering can be constructed by a cascade combination of a high-pass filter that reduces the pressence of P- and T-waves as well as baseline drift, usually implemented as a differentiator, and a low-pass filter that removes artifacts, or as a band-pass filter itself. Some approaches even consider the application of two cascaded bandpass filters, or two low-pass filters. For details on different filtering methods, consult [52].

Thus filtered ECG signals, $y(n)$, have to undergo a transformation to obtain a positive peak that represents each QRS complex. Again, multiple

approaches have been explored in the literature, ranging from simple square power modules to discrete Hilbert transform ECG envelope representation. See [58] for a comparison of different methods. Because the existence of high magnitude P-waves, abrupt stepwise fluctuation of the base line or other reasons, can produce fault peaks virtually all algorithms use additional decision rules for reduction of *false positive* detections. Specific solutions can be found in [60] and others.

Once $z(n)$ is obtained, which is a signal containing the candidate peaks in the temporal positions in which some fiducial points occur, most commonly the R-wave because it's the easiest to detect, the *peak detector* will typically compare the signal with some amplitude threshold, which can be fix or adaptive, and apply a *logic decision rule* in order to discriminate the peaks corresponding to actual QRS complexes to those which are spureous, obtaining subsequently $w(n)$ and lastly, the succession of time instants in which QRS complexes are reported to be present. Algorithms with adaptive thresholds that are updated after each beat detection based on some criteria, such as those described in [61] or [62], yield the best performance results. Time considerations are also taken into account in this stage: the duration of the QRS is constrained as mentioned in Section 2.2.3, and refractary periods, discussed in Page 13, mandatorily prevent the appearence of two subsequent QRS complexes in a too short interval. Thus, the R-R period cannot be suddenly shortened by a great factor, and this fact can help rejecting peaks from further consideration if they appear in an interval smaller than 200 ms approximately. On the other hand, R-R period cannot increase stepwise either, so if no new peak is detected in a reasonable interval after a previous one, retrospective approaches commonly known as *search-back* techniques can retrieve initially missed complexes by adapting the threshold, thus reducing *false negatives*.

Two different detection algorithms are proposed in this Thesis: one, based on Engelse and Zeelenberg's classic method ([63]), whose approach was to filter the signal in a particular way and then check the number of crossings over a threshold for discriminating QRS complexes from noisy features; and another, based on Pan and Tompkins' algorithm ([64]), in which the signal is prefiltered, derived, squared powered and integrated by means of mobile windows, and then searched comparing its values to an adaptive threshold which is updated under certain criteria, and taking into account physiological restrictions that make the algorithm search back if some conditions are met.

Readers may refer to the original articles for a complete description of the algorithms. MATLAB code of the routines programmed (`zeelenberg.m` and `tompkins.m`)is shown in Appendix D. Section 6.2 covers a detailed analysis of the linear filtering schemes employed in these two methods, and 6.3 deals with the detection performance itself.

4.4 Evaluation of Detection Performance

The usage of software QRS detection algorithms requires objective evaluation of the detection performance. It usually does not suffice to inspect visually

the QRS onsets/offsets determined by an algorithm superimposed to the ECG recording and assess the correct detection of QRS complexes based on coincidence with the waveforms timing. Consider the case of noisy ECGs, or severe arrhythmias such as fibrillating rythms or flutter, in which no coherent conclusion can be extracted. Even in the case of healthy ECGs, automated systems cannot rely in visual inspection for performance evaluation.

However, ECG databases annotated by professional cardiologists are available, which are ultimately the most certain source of information for assessing the correction or not of automated systems performance. This is the case of several publicly available databases used throughout this work, as noted in Section 4.1. Hence, it is possible to compare the automated detection performed by designed algorithms[3] and the annotation previously carried out by cardiologists, which is assumed to be undoubtfully truthful and correct.

Of course manual comparison is possible in the training stage of newly designed systems and algorithms, but impractical in real time applications, in which a highly fair performance must be granted and detection must keep continuously running.

In this context, measures of performance are defined based on the count of *FP* (false positives) and *FN* (false negatives), which are respectively the incorrect detections and incorrect miss-detections reported by an automated system, in discrepancy with the cardiologists' criterion. *TP* (true positives) and *TN* (true negatives) are the correct detections and correct non-detections calculated by the system, in agreement to the cardiologists' criterion.

In some literature, *TP* are called *hits* and *TN* are referred to as *correct rejections*, whereas *FP* and *FN* are called respectively *false alarms* and *misses*.

TN are not commonly employed in performance reports, since a detection system is supposed not to introduce wrong miss-detections, but to provide the most *TP*, although on the other hand it is likely to miss a certain amount of real QRS complexes if its discernment fails, which is the reason for *FN*. *FP* arise when the system is too sensitive and rejection criteria are not incorporated, hence confusing other waves or noise fluctuations with a QRS complex which is actually not present.

Measures of performance can be used for objective quality assessment as well as for comparison purposes between different detection systems if they follow a common definition, and experiments are carried out on the same signal databases as suggested in [52]. With that aim, the Standard defined in ANSI/AAMI EC57 recommends reporting performance results in basis of the following two parameters:

$$Se = \frac{TP}{TP + FN} \quad \text{sensitivity} \tag{4.4.1}$$

$$PP = \frac{TP}{TP + FP} \quad \text{positive predictivity} \tag{4.4.2}$$

[3] As we will see, this approach can also be valid for evaluation of beat classification performance.

Sensitivity, also called *hit rate* or *true positive rate*, is defined as the ability of a system to perform detection correctly, being expressed as the quotient of the positively detected cases over the total amount of cases (in this context, QRS complexes), whereas *positive predictivity* is the probability that a positive prediction is correct.

Another very common parameter is *specifity*, sometimes confused with *positive predictivity* in medical statistics. *Specifity* is the ability to identify the absence of the goal pattern (a QRS complex) correctly., and is given by the expression:

$$Sp = \frac{TN}{TN + FP} \quad \text{specifity} \tag{4.4.3}$$

The *false alarm rate*, or *false positive rate* is defined as:

$$FPR = \frac{FP}{TN + FP} = 1 - Sp \quad \text{false positive rate} \tag{4.4.4}$$

Evidently, the better the performance of the detecting system, the greater these parameters, which can range theoretically from 0 to 1. It is also common to find them given as percentages, in which case the perfect situation would be a *sensitivity*, *specifity* and *positive predictivity* of 100%. On the contrary, an ideal detector would yield a *false positive rate* of 0%.

The number of *total annotated beats* equals the sum of *TP* and *FN*, whereas the number of *detected beats* results the sum of *TP* and *FP*. Therefore, the expressions for *sensitivity* and *positive predictivity* can be equivalently rewritten as:

$$Se = \frac{TP}{Annotated\ beats}$$
$$PP = \frac{TP}{Detected\ beats}$$

Alternative definitions of these parameters based on the concepts of *shifted positive error* (*SP*) and *shifted negative error* (*SN*) are presented in [62], actually providing a correction that prevents reported results from under-estimating performance goodness. The *shifted errors* introduce the consideration of replacing a formal sequence of a *FP* immediately followed by a *FN* by one error only, labeled as *SP*. *SN* follows the same principle but in the opposite sense. Thus, Equations (4.4.1) and (4.4.2) would be rewritten as[4]:

$$Se = \frac{TP}{TP + FN + SN}$$

$$PP = \frac{TP}{TP + FP + SP}$$

[4]In [62] and other papers, *positive predictivity* is called *specifity*. The nomenclature is simply an agreement issue; what is important is the meaning of the parameters being calculated. In the present thesis we will follow the definitions given by Equations 4.4.2 and 4.4.3

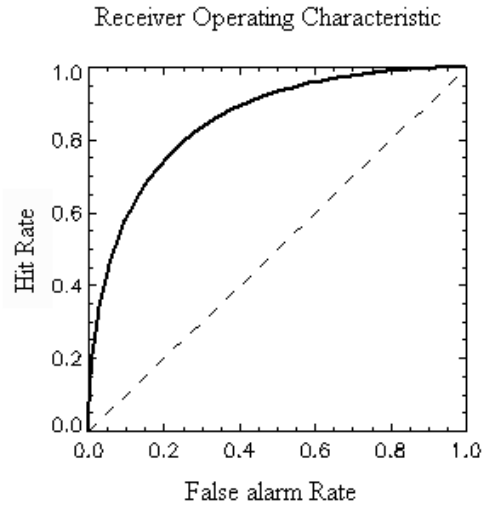


Figure 4.2. Typical ROC curve. The diagonal depicted represents a totally random prediction. The bigger the area between the ROC curve and the diagonal, the better the performance results.

Note that any single parameter constructed using only two numbers out of the four numbers *TP*, *TN*, *FP* and *FN*, is bound to be highly biased in some trivial way ([65]). Furthermore, the information given is not complete, since a detection system could be for instance highly tuned and not miss any detection, providing a 100% result in *sensitivity*, but at the same time introduce many *FP*, thus decreasing the *positive predictivity* measure. That is the reason why a bigger amount of information is prefered when reporting detection performance results.

Other merit factors or parameters commonly encountered in literature for these purposes are:

- *False discovery rate.* $FDR = FP/(FP + TP)$
- *Accuracy.* $ACC = (TP + TN)/(TP + FN + FP + TN)$
- *Receiver operating characteristic* (*ROC*), or *relative operating characteristic*. Graphical plots displaying *hit rate* versus *false alarm rate* for different values of a certain parameter adjusted in the algorithm, usually decision thresholds. Alternatively, sensitivity can be plotted versus specificity. The concept of ROC is presented in [66], comprehensively studied in [67] and extensively used in [56] and many other research papers. Figure 4.2 shows the aspect of a typical ROC curve.
- *d'*. In association with *ROC* plots, *d'* is defined as the distance along the negative diagonal between the *ROC* curve and the 50%-50% point. Larger *d'* values indicate superior performance of the overall detection.

- Other related features to *ROC* curves are also studied and provide information about the detection performance, such as the area under the curve (commonly called "*integrated receiver operating characteristic*", *IROC*), the area between the *ROC* curve and the no-discrimination line (positive diagonal), etc. See Figure 4.2. [68] gives an insight on these calculations.

- If accurate time annotation records are available, the *average time error* can be computed as:

$$\text{Average time error (ms)} = \frac{\sum |\textit{Detected QRS time} - \textit{Actual QRS time}|}{TP}$$

- *Matthews correlation coefficient* (*MCC*), parameter that takes into account true and false positives and negatives and is generally regarded as a well balanced measure with applicability in many prediction algorithms fields, avoiding the bias of simpler parameters as discussed above. Its value spans between -1 and 1. A coefficient of +1 means a perfect prediction, 0 an average random prediction and -1 a completely inverse prediction.

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \tag{4.4.5}$$

  If any of the four sums in the denominator equals zero, the denominator can be arbitrarily set to 1, coherently with the results of the corresponding limits, resulting in a *MCC* of zero. This parameter is described in detail in [65].

- Q_α, defined as the average of the *hit rate* (sensitivity) and the *specifity* (sensitivity of the negative category), also takes into account the four significant numbers: *TP*, *TN*, *FP* and *FN*:

$$\begin{aligned} Q_\alpha = \frac{Se + Sp}{2} &= \frac{\frac{TP}{TP+FN} + \frac{TN}{TN+FP}}{2} = \\ &= \frac{TP \cdot FP + 2 \cdot TP \cdot TN + TN \cdot FN}{2 \cdot (TP + FN) \cdot (FP + TN)} \end{aligned} \tag{4.4.6}$$

For the evaluation of the detectors employed in this Thesis, carried out in Section 6.3 we will mainly use the *sensitivity* and *positive predictivity* performance measures defined as in the standard ANSI/AAMI EC57 by Equations (4.4.1) and (4.4.2), since that is the simplest way of providing comparable results in relation to other studies. Nevertheless, correlation measures will also be provided, since despite of being uncommon in the QRS complexes detection field, the author understands that completeness of performance reports is important and hence a parameter collecting all information given by the *TP*, *TN*, *FP* and *FN* quantities is an interesting indicator.

4.5 Beats Classification

Another important part of the ECG processing is beat classification, or arrhythmia classification. The idea underlying is the same to both concepts: once

the QRS detection has been done and reference points have been located, for instance the onsets of R-waves, one has valuable information about the heart beats present in the record. By segmenting the whole ECG record into windows centered on, or referred to, these onset instants, we can isolate the different beats and perform varied kind of analysis over the samples.

Many different approaches orbit arrhythmia and beat classification. In literature it is common to find *time dominion analysis*, *frequency dominion analysis*, and even mixed *time-frequency techniques*.

Locating specific faulty areas of the heart is not easy by means of just comparison to known features and grouping, but really simple procedures can help keeping track of isolated abnormal beats or persistent episodes.

A simple possibility to classify arrhythmyas is to calculate the FFT (Fast Fourier Transform) of the ECG records and obtain the PSD (power spectral density). With the basis of the magnitude and frequency of the components that appear, it is also possible to sort out the different arrhythmias in some coarse groups.

Below we present a sorting scheme that can be used for classifying arrhythmias based on the PSD distribution of the signals. It is easy to develop automated algorithms that perform this classification or more complex ones based on related criteria. For further details about these arrhythmias, turn to Sections 2.1 and 2.2.

Non-defibrillable rhythms
- Disperse PSD.
- Main peak between 0.8–4 Hz.
- High-order, random, non-decreasing harmonics.
- Types:
  - Sinus rythm (SR). Many peaks, most of them located in frequencies multiples of 1 Hz. Very disperse PSD.
  - Atrial fibrillation (AF). Many peaks, most of them located in frequencies multiples of 1 Hz. More compact PSD.
  - Supraventricular tachycardia (SVT). High-powered harmonics, minimum power in the frequency range between harmonics.

Defibrillable rhythms
- Average-high concentration of PSD.
- Lesser pressence of harmonics.
- Types:
  - Ventricular fibrillation (VF). PSD highly concentrated. Main peak between 4–8 Hz. In most cases, no harmonics are present.
  - Ventricular tachycardia (VT). Medium concentration of PSD. Main peak between 2–4 Hz. Small value, decreasing harmonics.

One of the main trends in present-day is the use of wavelets decomposition in this kind of analysis. Readers are referred to [69, 70, 71, 72, 73] for a sample of diverse approaches to beat classification.

Another simple morphological analysis in the time-domain, can consist of scanning the signal beat by beat, correlating each beat with previously acquired templates, or reference beats. If the result of the correlation exceeds a certain threshold for one of the templates, the beat in question can be grouped in a

common 'type' with that template and other similar beats. Otherwise, a new group ought to be created in which to collect the current beat and other similar ones that might appear later in the scanning process.

Hopefully, in a healthy ECG the beats should not differ much one from each other. If the beat classifier finds many ectopic beats in a short time, it could be a sign of a severe arrhythmic episode, which could be detected triggering apropriate alarms in the algorithm.

Two time-domain beat classifiers based on these principles have been programmed as a part of this work, using template comparison by correlation threshold as explained above, intending to compare their results to those of the annotations in the MIT files. We present here a brief description of their operation. Notice that implementation is designed for a sample frequency of 250 Hz, and window length for the beat set has been chosen to be 140 ms, which corresponds to 35 samples at that sampling frequency. A matrix of templates keeps track of the different reference beats. An optional input to the algorithms is the threshold for which the correlations are accepted (*threshold*).

The algorithmic description of the first classifier is as follows:

Step I) *Preprocessing: filtering to remove noise.*
- *i) High-pass filter, cuttoff frequency = 1 Hz.*
- *ii) Low-pass filter, cutoff frequency = 16 Hz.*

Step II) *Window of 140 ms around the position of R-wave. 15 samples before R-wave, 20 samples after R-wave.*

Step III) *First beat is stored as the first template.*

Step IV) *Place on next window. Correlate current beat with all the previous templates.*
- *i) If the maximum correlation coefficient is $\geq$ threshold, current beat is added to the group of the template for which correlation was maximum. Template is updated by averaging old template and current beat.*
- *ii) Otherwise, current beat is added to template matrix as a new template.*

Step V) *Go back to Step IV if the current is not the last window. If it is, END.*

Both filters described in Step III have been designed as Butterworth type filters because of their maximally flat response in the passband, since the aim is to attenuate other noise and interference but not distorting the signal of interest. The MATLAB code of this algorithm can be found in App. D under the name of `beatclassifier.m`.

In order to improve the results in terms of robustness, a modified extension of this method, called `beatclassifiermod.m`, is presented as well. The difference is that this modified version, after finishing the sorting of all the beats, validates the templates by checking if they have been obtained by averaging a specified minimum amount of heartbeats (five by default). If this is not the case, the algorithm scans back and tries to sort the beat into any of the other groups by slightly reducing the correlation threshold (in subsequent steps of 0.01), until reaching a specified critical value, below which the beat is considered unsortable into one of the other templates, and constitutes a valid template by itself.

By doing this, the method gains robustness against noise and performs better in "difficult" records producing more compact classifications, with the

tradeoff of accepting a bigger dissimilarity between the beats sorted into a common group for which the templates are progressively updated. For comparison with the simple method, the procedure is shown below. Optional inputs are the threshold for which correlations are accepted initially ($threshold$), the critical minimum value of threshold until which the resorting will be tried ($criticalthreshold$), and the number of significant beats needed to validate a template ($signbeats$).

Step I) *Preprocessing: filtering to remove noise.*
- *i) High-pass filter, cuttoff frequency = 1 Hz.*
- *ii) Low-pass filter, cutoff frequency = 16 Hz.*

Step II) *Window of 140 ms around the position of R-wave. 15 samples before R-wave, 20 samples after R-wave.*

Step III) *First beat is stored as the first template. A weight of 1 is assigned.*

Step IV) *Place on next window. Correlate current beat with all the previous templates.*
- *i) If the maximum correlation coefficient is $\geq threshold$, current beat is added to the group of the template for which correlation was maximum. Template is updated by averaging old template and current beat, and the weight of the template is increased by 1.*
- *ii) Otherwise, current beat is added to template matrix as a new template and a weight of 1 is assigned.*

Step V) *Go back to Step IV if the current is not the last window. If it is, proceed to Step VI.*

Step VI) *Validate templates by checking if their weight is $\geq signbeats$.*
- *i) If not all templates meet the requirement and threshold $\geq criticalthreshold$, do $threshold = threshold - 0.01$ and go to Step III.*
- *ii) Otherwise, END.*

Results of these two methods are presented in Section 6.4.

4.6 Analysis of HRV

As said in Section 2.4, studying the heart rate variability can be of great interest due to the fact that it shows the entrainment between the cardiac performance and neural activity by means of regulation exerted by the ANS. It is also a proofed indicator of sudden death risk. Thus, a variety of HRV measures have been developed over the years.

A common and very simple way of measuring the HRV in the time domain is by computing the standard deviation, square root of the variance, of the beat-to-beat periods, providing a statistical measure of the variation of the rate. This is often referred to as *SDNN*, standard deviation of so-called *normal-to-normal intervals* (N-N: all intervals between adjacent QRS complexes resulting from sinus node depolarizations).

Also the average of R-R intervals duration can be computed over a certain number of cardiac cycles, noting it *MeanRR*, providing the maximum and minimum values of the R-R intervals analyzed: *MaxRR* and *MinRR*. Note that the *mean HR* in bpm can be calculated as the inverse of the *MeanRR* measure.

Other commonly used statistical parameters calculated as segmented measures include *SDANN*, the standard deviation of the average N-N intervals calculated over short periods, usually 5 minutes, which is an estimate of the changes in heart rate due to cycles longer than 5 minutes, and the *SDNN index*, the mean of the 5-minute standard deviations of N-N intervals calculated over 24 hours, which measures the variability due to cycles shorter than 5 minutes ([74]).

Statistics calculated from interval differences are also common: *RMSSD* is the square root of the mean squared differences of successive N-N intervals, *NN50* is the number of interval differences of successive N-N intervals greater than 50 ms, and *pNN50* is ratio of NN50 to the total number of N-N intervals. All these measurements of short-term variation aim to estimate high-frequency variations in HR and are thus highly correlated. However, each of the different measures has been shown to be better predictor of relative risk of death due to concrete pathologies ([75], [76]).

The *Poincaré plot*[5] is also used in literature to analyze HRV ([77]). A Poincaré plot is a 2-D scattergram representation used to quantify self-similarity in processes. The values of R-R intervals are plotted as a function of previous respective values of the intervals, this is to say, $R - R(n)$ are plotted on the x-axis vs. $R - R(n+1)$ on the y-axis.

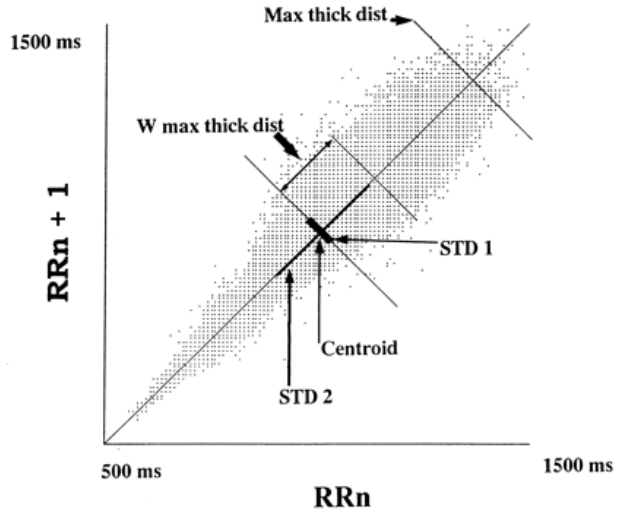


Figure 4.3. Quantitative analysis of Poincaré plots. Centroid indicates point of the average R-R interval. STD1 and STD2 indicate SD of instantaneous and continuos R-R interval variability, respectively. Max thick dist and W max thick dist indicate distance between centroid and maximum instantaneous and averaged maximum of instantaneous R-R interval variability, respectively. Reproduced from [77].

Quantitative analysis of Poincaré plots is shown in Figure 4.3. It involves fitting an ellipse to the plot with its centre coinciding with the centroid of the

[5]Named after French mathematician, physicist and philosopher Jules Henri Poincaré.

data (point of average R-R interval). Its two axis, STD1 and STD2, indicate the standard deviation of instantaneous (beat-to-beat) and long-term variability, respectively. The ratio STD1/STD2 is computed for characterization. Also, the distance between the centroid and the averaged maximum of instantaneous R-R interval variability and the one between the centroid and the maximum instantaneous R-R interval variability can be calculated. According to the studies and results reported in [77], healthy individuals or patients without propensity for ventricular tachycardia typically exhibit fan-shaped scattergrams, whereas patients who suffer from ventricular tachyarrhythmias show either complex ball-shaped or torpedo-shaped plots, as can be seen in Figure 4.4, which justifies the fact that computation of STD1, STD2, their ratio, and the SD of heart rate can help providing useful information for clinical diagnose.

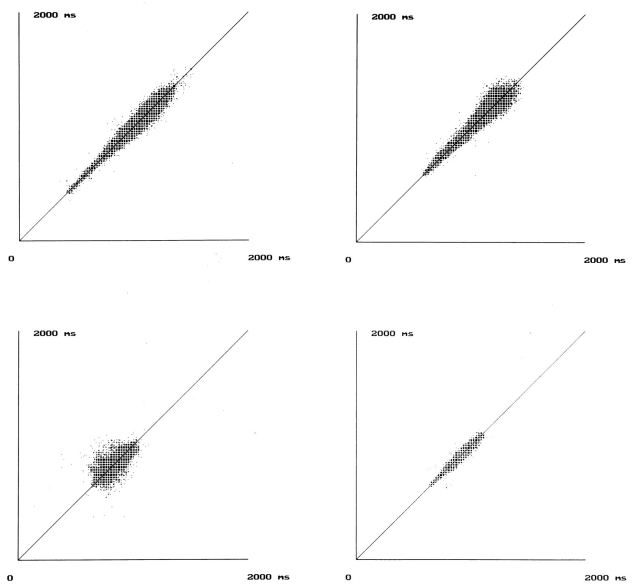


Figure 4.4. Examples of Poincaré plots analyzed from at least 80.000 consecutive sinus beats for a healthy subject (top left), a postinfarction patient without any propensity for ventricular tachycardia (top right), and two patients with a propensity for ventricular tachyarrhythmias (lower plots). Reproduced from [77].

The research exposed in [78] shows however that the ellipse-fitting technique, and others such as those based on histogram analysis or the correlation coefficient to characterize the shape of the scattergram, rely ultimately on linear indices, thus not exploiting the nonlinear characteristics of the R-R intervals variability showed in the Poincaré plot. A slightly new interpretation and generalizations are proposed.

Another geometrical measure of HRV commonly found in articles and scientific literature is the *triangular index*. For computing it, the sample density distribution of N-N intervals duration D is constructed, assigning the number of equally long N-N intervals to each value of their lengths, similarly to an

histogram. The most frequent N-N interval length, labelled X in Figure 4.5, is found, with a number of $Y = D(X)$ occurances. The HRV triangular index indicates how the area of D resembles a triangle around X, being calculated as the result of the division between the integral of D and the maximum value Y. If a measurement of N-N intervals on a discrete scale is used, the measure is approximated by dividing the total number of N-N intervals and the number of N-N intervals in the modal bin, result which is dependent on the length of the bin, that is, on the precision of the discrete scale of measurement.

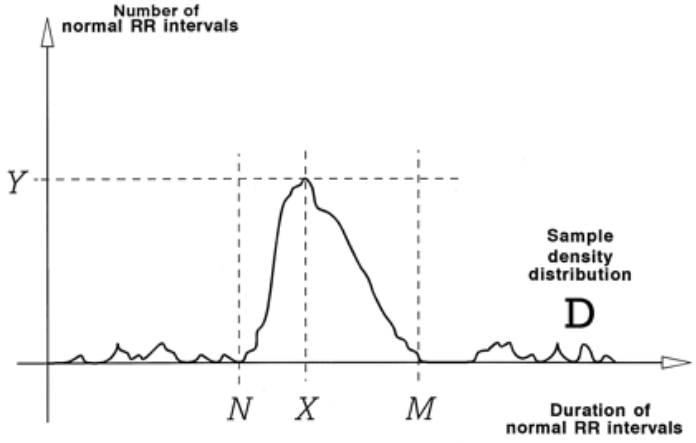


Figure 4.5. Sample density distribution of N-N interval durations for the calculation of the HRV triangular index. Reproduced from [74].

The major disadvantage of geometrical measures such as the triangular index is the need for a relatively big number of N-N intervals to be able to construct the pattern, in this case the distribution D. This usually requires long ECG recordings to ensure the correct performance of the method, being thus inappropriate to assess short-term changes in HRV.

Frequency domain measures have also been widely used in literature. Commonly, the power spectral density (PSD) of the heart rate time series, in which ectopic beats have been previously eliminated to consider only N-N intervals, is calculated via an FFT or a different method[6]. Then, several measures are computed by integrating the PSD over different frequency ranges:

- ULF power, <0.0033 Hz,
- VLF power, 0.0033 to <0.04 Hz,
- LF power, 0.04 to <0.15 Hz,
- HF power, 0.15 to 0.40 Hz, and
- total power in the band ≤0.40 Hz.

[6]According to [59], the Blackman-Tukey method of spectral estimation, or in general, harmonic PSD techniques, are less sensitive to signal morphology than traditional PSD estimates because of the accounting for variations in the power distribution among harmonic frequencies; also, they achieve better frequency resolution on the lower harmonics by leveraging the relatively better resolution at the harmonic frequencies.

According to [79], these frequency bands are useful for clinical prognosis. The VLF band power shows an increase in patients with congestive heart failure, the LW power reflects modulation of sympathetic or parasympathetic tone by baroreflex activity, and the HF range power reflects modulation of vagal tone, primarily by breathing. See Figure 4.6. The distribution of the PSD and the relation between the power measures in each range allows to determine relative risk of death due to arrhytmic episodes, cardiac failure or other reasons.

The ratio of LF to HF power is calculated too, as an indicator of sympathovagal balance, with high values for the ratio suggesting predominance of sympathetic nervous activity and viceversa. Also, normalization is sometimes applied in short-duration recordings to the LF and HF ranges, dividing the power in those bands by the total power minus the ULF and VLF components and multiplying by 100. This is what is referred to as *nu*, normalized units, in Table 4.1, which shows the normal values of some of the most used measures of heart rate variability, both in the time and the spectral domain.

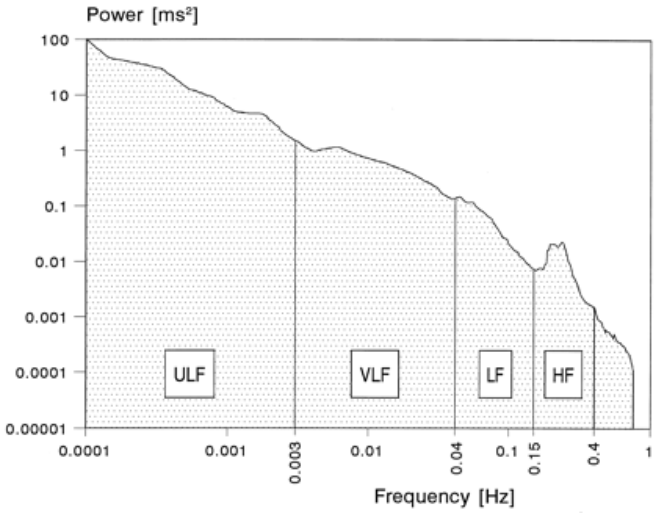
Time domain analysis of nominal 24 hours		
SDNN	141±39	ms
SDANN	127±35	ms
RMSSD	27±12	ms
HRV triangular index	37±15	
Spectral analysis of stationary supine 5 min recording		
Total power	3466±1018	ms^2
LF	1170±416	ms^2
HF	975±203	ms^2
LF	54±4	nu
HF	29±3	nu
LF/HF ratio	1.5–2.0	

Table 4.1. Normal values of some of the standard measures of HRV. Reproduced from [74]. Data said to be obtained from a small number of subjects; no other complementary information is provided.

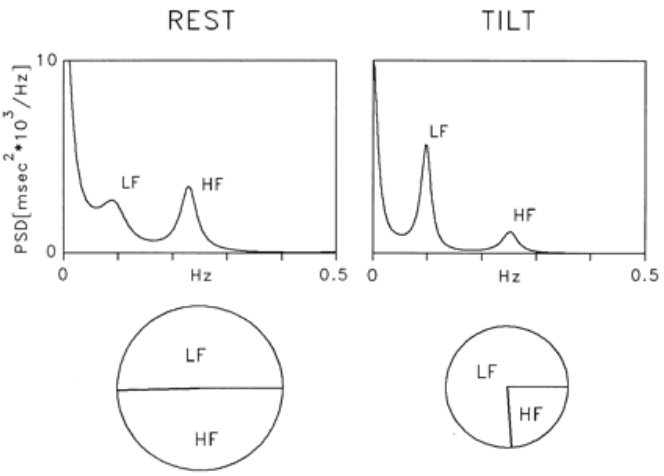
The slope of 24-hours recordings' spectrum can be assessed on a log-log scale by linear fitting the spectral values, and is usually called the α measure of HRV (see Figure 4.6).

Regarding the interrelation of the different measures, it is known that SDNN, SDANN, and the total power and the ULF power in spectral analysis are highly correlated among each other. The same occurs for the SDNN index and VLF and LF power, as well as for the RMSSD, pNN50 and HF band power. However, the LF/HF ratio does not strongly correlate with any other HRV measures.

Other methods of measuring heart rate variability include power-law scaling analysis (computed on log-log power-frequency plots as explained above



(a) PSD of HR obtained from a 24-hours interval of a Holter recording



(b) Spectral analysis (autoregressive model, order 12) of RR interval variability in a healthy subject at rest and during 90° head-up tilt.

Figure 4.6. In the example PSD obtained from 24-hours recording in (a), only the LF and HF bands correspond to peaks in the spectrum, while the VLF and ULF can be approximated by a line in this log-log plot. The slope of such a line is the α measure. In (b), at rest, two major components of similar power are detectable at low and high frequencies. The LF/HF ratio is 1.02. During tilt, the LF component becomes dominant, but as total variance is reduced, the absolute power of LF appears unchanged compared with rest. Normalization procedure leads to predominant LF and smaller HF components, which express the alteration of spectral components due to tilt. The LF/HF ratio is 3.34. Pie charts show the relative distribution together with the absolute power of the two components represented by the area. Reproduced from [74].

with the α measures, [80], [81], or others), detrended fluctuation analysis ([82]), statistical analysis based on Student's t-tests, and more recently, diverse methods based on nonlinear system theory, namely chaos theory and fractal measures ([83, 84, 85, 86]).

The interested reader can find in [74] a comprehensive description of time domain and spectral measurements of HRV, as well as a clinical insight on the correlation between HRV and physiological and physiopathological processes. The study in [87] also reports the relation between HRV and several common diseases and parameters such as gender, age, health, drugs and alcohol consumption, etc, as well as listing a compilation of measures from which entropy and nonlinear analysis stand out. In [88] we find an exhaustive compilation of traditional and novel HRV measures of several domains and approaches, as well as several mathematical models for HRV. Also, [84] contains an appealing comparative study between new fractal analysis and traditional spectral and non-spectral measures for HRV, reporting that fractal analysis provides better all-cause mortality prediction results than traditional methods.

For the analysis of the HRV, we have programmed in this work an M-function[7] called `HRV.m`, which code is listed in Appendix D, implementing some time-domain measures calculation and plotting of the HR. We also use the Kubios HRV analysis software, developed by the Biosignal Analysis and Medical Imaging Group (BSAMIG) at the Department of Physics, University of Kuopio, Finland, which implements some more time-domain measures and spectral analysis. Results are provided in Section 6.5.

[7] M-functions is the name commonly given to functions programmed in the MATLAB environment.

> The fundamental problem of communication is that of reproducing at one point either exactly or approximately a message selected at another point. Frequently the messages have meaning; that is they refer to or are correlated according to some system with certain physical or conceptual entities. These semantic aspects of communication are irrelevant to the engineering problem. The significant aspect is that the actual message is one selected from a set of possible messages. The system must be designed to operate for each possible selection, not just the one which will actually be chosen since this is unknown at the time of design.
>
> *A Mathematical Theory of Communication*
> Claude Elwood Shannon, 1948

Chapter 5

Communication Model

In this Chapter we present the communication model proposed for the implanted control unit. This model has been specifically chosen for the application of ULTRASPONDER project based on the results obtained in the mathematical-statistical analysis performed on the signals of interest.

We first focus on signal compression introducing the fundamentals of general data compression and focusing later on direct ECG data compression methods in the time-domain, and concretely, on a closed-loop DPCM system. A brief discussion of the linear predictor design process is provided, along with references for further detail. Possibilities for quantization systems are also explained.

5.1 Fundamentals of Data Compression

Necessity of real-time analysis algorithms, proliferation of ECG data transmission over the public phone network or dedicated lines, improvement in functionality of ambulatory ECG monitors and recorders and massive storage of signals for usage in standarized databases, are just some of the reasons and applications that started calling for the development of specific ECG data compression techniques.

The main goal of any *compression technique* is to achieve maximum data volume reduction while preserving the information content upon reconstruction. Conceptually, *data compression* is the process of detecting and eliminating redundancies in a given data set ([89]) in order to reduce the necessary storage capacity. The definition for *redundancy* given by Shannon in [37] states that it

is "that fraction of a message or datum which is unnecessary and hence repetitive in the sense that if it were missing the message would still be essentially complete, or at least could be completed".

The process is also commonly called *source coding*, in the sense that it consists of encoding information so that it can be represented with fewer bits or any other information-bearing units. But apart from coding, data volume reduction can also be approached by pattern recognition, linear prediction, and other methods.

There are two general approaches to data compression: a) *lossless compression* algorithms, which exploit statistical redundancy in the original data and permit reconstruction without any error, and b) *lossy compression* algorithms, which are designed to yield greater compression rates at the tradeoff of accepting some loss of fidelity.

Some of the latter are also referred to as *perceptual coding* schemes, for they are usually based on the way an eventual receiver perceives the specific data type in question. Examples of these are the JPEG image compression and MP3 audio compression formats, in which lossy schemes are applied and as high compression ratios as 10:1 or even more can be achieved with little perceptible loss in image and audio quality to the average human eye and ear, respectively[1].

Examples of lossless compression schemes are Slepian-Wolf coding, entropy coding or Lempel-Ziv dictionary-based coding. These compression algorithms always have a fixed limit in their capacity to remove redundancies and hence compress the data, whereas lossy methods could theoretically attain arbitrarily big compression by also increasing the error upon reconstruction. In most applications, obviously, the admissible difference between the original data and the reconstructed data after encoding, transmission (or storage), and decoding, imposes a limit on the feasible compression.

Applying these concepts to the very specific field of ECG data compression, results in fixing the goal as to achieve the maximum *compression ratio* (CR) by eliminating redundancies in the original data while preserving the significant signal morphology features relevant for clinical diagnosis. Virtually all ECG data compression methods are based on lossy algorithms, which allow for some reconstruction error that must be controlled by some mechanisms to avoid excessive distorsion.

With regard to the dominion in which they operate, compression methods can be classified into three major groups [89]:

a) Direct data compression,
b) Transformation methods, and

[1]Subjective perception of quality is a vital feature in the design process of all modern digital cameras, which apply lossy algorithms to compress and store the pictures in their constrained memory devices with minimal degradation of picture quality. This subjective perceived quality is also an interesting factor in all the audio compression standard formats. While MP3 greatly reduces the amount of data required to represent audio recordings and the played-back files still sound like a faithful reproduction of the original uncompressed audio for most listeners, people with highly-trained hearing, such as musicians, sound specialists, etc., claim to notice a significant reduction of quality, depending on the compression rate.

c) Parameter extraction techniques.

Techniques based on *parameter extraction* approach the compression by analyzing the signal and extracting diverse particular characteristics or parameters that are used to represent the signal. The original signal is irreversibly substituted by its representation in accordance to these parameters in the transmission channel or storage system, and it will be reconstructed using them based on an *a priori* knowledge of the signal features. Therefore, these kind of approaches are strongly application-dependant since the concrete context determines the relevant parameters to be extracted, and the possibility of further compression. These methods are very popular in the speech compression, recognition, and voice reconstruction and generation fields, specially in mobile communications, since the great simplification of the amount of information to be processed yields good results in terms of the tradeoff between compression and perceived quality. However these techniques are less popular for application in ECG data compression for ambulatory monitors, at least in pure implementation forms. Direct data compression techniques and diverse transformed dominion analysis were historically developed before, and although ECG compression methods based on analysis by synthesis have been described and applied in literature, they are much less common.

Thus, popular existing ECG data compression techniques lie mainly in one of the two first categories. Nevertheless, methods called *peak-picking compression* techniques based on the descriptions proposed in [90] have been described in literature ([91] and others). These methods, although operating directly on actual sample values, could be understood as based in the extraction of some concrete parameters of the waveforms; reconstruction of the signal is done by means of spline functions.

Direct data compressors base their detection of redundancies on direct analysis of the actual signal samples, whereas *transformation compression methods* mainly utilize spectral and energy distribution analysis for it. Since the compressed data obtained by either family of methods contains actual or transformed data from the original signal, the reconstruction is done by an inverse process, unlike the case of *parameter extraction* in which a reconstructed version of the original data are somehow generated from the available parameters and knowledge of how the signal should look like and which should its spectral content be.

Some of the most popular *transformation compression* techniques applied to ECG signals are reviewed in [89]. In general, they involve preprocessing the input signal by means of linear orthogonal transformations and properly encoding the transformed output (a set of coefficients dependant on the transformation characteristics), reducing the amount of data needed to adequately represent the original signal. They have been broadly applied in VCG or multilead ECG compression.

In this Thesis we focus on *direct data compression* techniques, which have classically and so far shown better results in ECG compression regarding particularly to processing speed and in general to compression ratio, according

to the review in [92]. They are also easier to implement than transformation based algorithms. These techniques are described in Section 5.3.

5.2 Compression Ratio and Distortion Measures

Before proceeding, it is convenient to give a formal definition of the concept of *compression ratio* referred to above, since it is the main parameter and goal that measures the goodness of compression techniques. Intuitively, it can be defined as:

$$CR = \frac{\text{Uncompressed Size}}{\text{Compressed Size}} \tag{5.2.1}$$

Thus a representation that compresses a 10 MB file to 2.5 MB has a compression ratio of $10/2.5 = 4$, usually notated as a ratio of 4:1 (read "four to one"). However, this absolute and simple quotient can be misleading, since when directly computed on the amount of samples that the original signal had and the compressed signal has, it can obviate the overhead information that may be needed to actually reconstruct the signal.

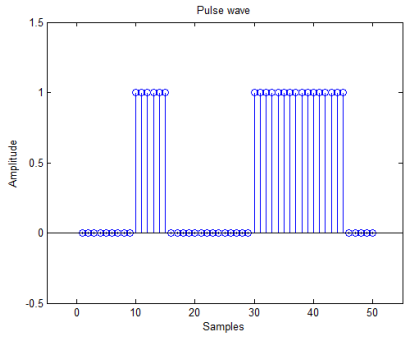
Consider for instance a very simple compression algorithm which saves one value of a signal sample, and then eliminates all subsequent identical samples until it finds a sample whose value is different, storing it again and repeating the process. It would certainly yield a very poor CR for most signals, but consider its application to compressing an also very simple signal such as the one depicted in Figure 5.1a. The algorithm would start by storing the value 0 which is the first sample's. It would then delete all subsequent samples until the tenth, in which the signal changes its value to 1, saving that sample and deleting the next samples until the sixteenth, in which the signal changes again to 0, being stored. The process would be repeated for the second pulse wave.

After this algorithm has compressed the signal, instead of a binary sequence of fifty samples, we would have the sequence: [0 1 0 1 0], still binary, so one could think that the compression ratio is CR = 50/5 = 10, since the length of the obtained sequence is 5. But of course this representation of the signal ('One'), although being a compressed version of the original, is not complete, since no information has been included to stablish the duration of pulse waves, that the decoder would need to know when to stop the repetition of holding values. During the reconstruction task, the decoder could generate a big (actually, infinite) number of different final sequences that match the compressed version of the original signal. Obviously, the coding-decoding process must be univocal and thus some additional info is necessary so that the decoder knows how long should the pieces of the signal be.

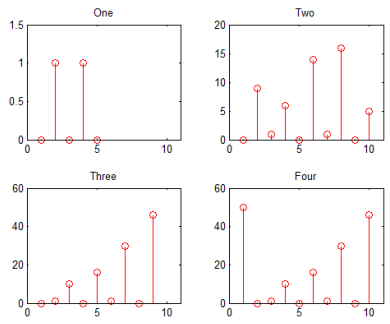
Hence a rather simple possibility would be to attach after the value of each new sample, the number of samples with identical value as the one just saved. This would give us the sequence ('Two'): [0 9 1 6 0 14 1 16 0 5], which is not binary and whose length is 10, so without further coding, the CR could thought to be CR = 50/10 = 5. A new disadvantage would be that the compressed sequence contains both data from the original signal, namely values

1 and 0, and newly introduced data, which could in turn increase the dynamic range of the quantizer used to code the sequences and waste big amounts of representation space decreasing the efficiency of the scheme.

Other possibilites could be thought of, such as storing the index of each sample with different value, and assuming that the first value corresponds to the first sample and is always stored, saving in this case ('Three') one sample in the final sequence in comparison with the previous option and providing CR = $50/9 = 5.\widehat{5}$, also with the disadvantage of including not-binary information, i.e. newly generated samples that were absent in the original signal. Notice that this approach is useless if the decoder does not know the total length of the sequence, since it would not know when to stop adding 0s to the reconstructed samples. Adding the length of the sequence as the first value, for example, would prevent this problem, lowering the CR to 5:1 again, but in this case ('Four') the statistical parameters of the compressed sequence would be quite different from those of case 'Two'. Figure 5.1b shows these alternative representations of the original signal.



(a) Original pulse wave



(b) Alternative compressed versions of the pulse wave with four different compression schemes

Figure 5.1. Compression of the pulse wave showed in (a) by four different very simple algorithms. The sequences obtained by 'One', 'Two', 'Three' and 'Four' achieve a CR of 10, 5, $5.\widehat{5}$ and 5, respectively. Notice the fact that these CR are referred simply to the number of samples, and do not take into account quantized amplitudes

These simple examples illustrate that when computing the CR, not only the samples that survived the redundancy deletion from the original signal must be taken into account, but also any other extra information generated by the algorithm, which the decoder will need to know in order to reconstruct the signal without uncertainty. Depending on the signal to compress and the algorithm employed, this overhead data can become a significant percentage of the final compressed data.

It is important to consider that the attainable CR will depend on the sampling frequency employed to digitize the signal. The higher the sampling rate, the greater the CR can be, and conversely. This is due to the fact that oversampled signals, and particularly, oversampled ECG data, are intrin-

sically redundant, thus a higher degree of compression can be achieved without degradating the quality of the reconstructed signal. Linked to this, besides visual comparison between the original and the reconstructed data, which is impractical in real-time applications and carries all evident disadvantages of non-objectivity, a fidelity measure commonly employed is the *percent root-mean-square difference* (PRD), calculated as follows:

$$PRD = \sqrt{\frac{\sum_{i=1}^{N}[x_{org}(i) - x_{rec}(i)]^2}{\sum_{i=1}^{N} x_{org}^2(i)}} * 100 \tag{5.2.2}$$

where $x_{org}(i)$ and $x_{rec}(i)$ are samples of the original and reconstructed data sequences, and N is the length of the sequences.

For signals of indefinite or unknown size, such as for instance those used in audio or video streaming applications, or in real-time ECG processing in ambulatory devices, the compression ratio is defined in terms of uncompressed and compressed data rates instead of data sizes, and Equation (5.2.1), that can only be calculated *a posteriori*, is rewritten as:

$$CR = \frac{\text{Uncompressed Data Rate}}{\text{Compressed Data Rate}} \tag{5.2.3}$$

and PRD must be calculated on windowed segments of the signal:

$$PRD_j = \sqrt{\frac{\sum_{i=1}^{m}[x_{org}(i) - x_{rec}(i)]^2}{\sum_{i=1}^{m} x_{org}^2(i)}} * 100 \tag{5.2.4}$$

where m has been taken as the size of the j-th window, and $j = 1..N/m$.

An alternative definition of PRD using other normalization is also found in literature:

$$PRD_m = \sqrt{\frac{\sum_{i=1}^{N}[x_{org}(i) - x_{rec}(i)]^2}{\sum_{i=1}^{N}[x_{org}(i) - \bar{x}_{org}]^2}} * 100 \tag{5.2.5}$$

where $\bar{x}_{org}$ is the mean of signal x_{org}. This definition is thus independent of the DC level of the original signal. If the signal treated has zero mean, 5.2.2 and 5.2.5 are of course the same. This measure PRD_m is usually more convenient in ECG distorsion evaluation, since the important part of an ECG signal is not the signal level itself, but the time variation of the signal.

Some other common distorsion measures ([93]) used in ECG compression evaluation are the *root mean square error* (rms):

$$rms = \sqrt{\frac{\sum_{i=1}^{N}[x_{org}(i) - x_{rec}(i)]^2}{N}} \tag{5.2.6}$$

the *signal-to-noise ratio* (SNR):

$$SNR = 10 \log \left(\frac{\sum_{i=1}^{N}[x_{org}(i) - \bar{x}_{org}]^2}{\sum_{i=1}^{N}[x_{org}(i) - x_{rec}(i)]^2} \right) \tag{5.2.7}$$

or the *maximum amplitude error* or *peak error* (MAX or PE) expressed as:

$$MAX = \max_i \{|x_{org}(i) - x_{rec}(i)|\} \tag{5.2.8}$$

Whereas most authors measure their algorithms goodness by means of the sum of squared deviations, bounds on the error at individual samples should also be taken into account, since big errors on localized samples can be of little significance in distorsion measures such as PRD or rms, but should not be tolerated as they can be seriously significant in medical diagnosis in terms of the actual distortion introduced. Methods for computation of the *infinity norm error*, ϵ^{∞}, and control of it under a given upper bound $c^{-\infty}$, based on graphs theory, are covered in [94].

However, PRD and the other measures described above are not the best possibility for evaluating specific ECG compression schemes, since although the rms error between original and reconstructed waveforms is important, its measure does not reveal whether or not an algorithm can preserve diagnostically significant features of the ECG waveform, which is ultimately the goal that must be taken into consideration. These parameters assess only a measure of dissimilarity, in an arithmetic sense, between the original and the reconstructed signal. A problem arises which is that relevant diagnostic information to be preserved is mostly subjective and is determined by cardiologists perception.

Every segment in the PQRST complex has a different diagnostic meaning and significance, and some pathologies actually influence a part of the ECG and leave other untouched, so their automated detection focuses on the study of a concrete portion of the heartbeat. Thus, a given distortion in one segment does not necessarily have the same importance as the same distortion would have in a different segment. Mathematical measures which can be correlated with this expert diagnostic know-how are complex and uncommon in general literature, although application-dependant approaches sometimes consider them.

Such a measure is the *weighted diagnostic distortion* measure (WDD) first introduced in [95] accompanied by a compression algorithm based on analysis by synthesis coding, and furtherly described and developed in [96]. WDD has been successfully applied in ECG compression systems such as those reported in [97] and [98].

Its rationale is the extraction of diagnostic features, chosen with the help of an experienced cardiologist's criteria, such as location, duration, amplitudes and shapes of the waves and complexes that exist in heartbeats, from both the original signal and the signal recovered after compression and reconstruction. The comparison of these two sets of features allows to measure the relative preservation of diagnostic information along the compression process, as claimed by responsible researchers of the studies mentioned above, in a more useful way than measures such as PRD. This is because WDD gives information not only about the mathematical deviation from one signal to another, but also about the clinically relevant content preserved in the reconstructed signal in relation to that of the original signal.

However, the WDD measure was designed for the surface ECG, and its reliabilty and significance have not yet been proved, as far as the knowledge of the author of this work goes, for intracardiac ECGs which is the focus of the ULTRASPONDER project. Hence, it is not considered in this work, although it is likely that in the short-term future WDD proves to be a significant and useful distortion measure also for intracardiac ECG records.

5.3 Direct ECG Data Compression Methods

Having settled the basics of data compression in the previous Sections, we now focus on direct data compression methods applied to ECG signals. Most of these techniques rely on utilizing *prediction* or *interpolation algorithms*, reducing redundancy by examining a successive number of neighboring samples in a data sequence [89]. Predictors use *a priori* knowledge of some previous samples, whereas interpolators employ *a priori* knowledge of both previous and future samples.

For a better development of the discussion in this Chapter, details about the operation of these polynomial predictors and interpolators are given in Appendix B. Readers interested on them are suggested to consult the mentioned Appendix.

According to the algorithmic structure of the data reduction scheme, we can classify the direct data compression methods into three categories:

Tolerance-Comparison Compression These techniques attempt to reduce signal redundancy by taking advantage of the intersample correlation. A preset error threshold is employed in direct application of polynomial predictors or interpolators to eliminate redundant data samples. Predicted or interpolated data are compared with actual sample values over which a tolerance band of the established threshold is superimposed. If samples lie within the tolerance band they are considered redundant, and hence eliminated. If they lie outside this band they are considered nonredundant and they are stored. Therefore, higher values of the preset error threshold will result in higher resultant CR along with lower reconstructed signal fidelity, whereas lower values of the threshold produce better fidelity but attain lower CR's.

Differential Pulse Code Modulation (DPCM) Waveform redundancy reduction by DPCM coders is achieved by representing the original correlated signal in terms of an uncorrelated signal, namely, the estimation error signal: the data samples are estimated by means of polynomial interpolators or predictors, and the error or residual between the actual sample and the estimated value $e_n = y_n - \hat{y}_n$ is quantized and then stored or transmitted. The major source of reconstruction error in systems based on these methods is due to the amplitude quantization of the residual. The variance of the residual signal is generally smaller than that of the original signal, provided that the correlation of the latter is high and the estimation is relatively accurate. As the SNR achieved is dependent upon

the quantization process, quantizer design is a crucial issue in a DPCM system.

Entropy Coding Methods These schemes reduce signal redundancy that arises whenever the quantized signal amplitudes have a nonuniform probability distribution, by assigning variable-length codewords to the data according to their frequency of occurrence. Values ocurred more often (with higher probability in a statistical analysis) are assigned shorter code lengths, whereas less likely values are assigned longer codewords. This process results in the minimization of the mean code length. *Huffman coding*, first presented in [99], is the pioneer in variable-length codes construction, although many later techniques based on it have been developed. A systematic review of them is presented in [100]. Variable-length coding is often implemented as part of ECG DPCM systems for further increase of compression. An eventual disadvantage of this kind of coding is that simple transmission errors can lead to serious decoding errors if no special control techniques are implemented. These approaches necessarily imply some overhead data (a simple possibility is codeword delimitation by means of known patterns or alike) which must be minimized in order not to substantially reduce the data compression ratio attained by the coding process.

Popular schemes like AZTEC, TP, CORTES, and Fan/SAPA are ECG compression methods based on the tolerance-comparison principle, following different rationale.

From this point on, we center our discussion on the DPCM method, since this is the scheme chosen for the system proposed in this work due to the results of the theoretical analysis performed.

The estimator element of a DPCM system can be any algorithm such as the polynomial interpolators or predictors as said above (see Appendix B for details), but a more complex estimator called *linear predictor* is usually employed. Its operation is similar to that of the other predictors, estimating the next data sample by a linear combination of a weighted number of M previous samples, as given by:

$$\hat{y}_n = \sum_{j=1}^{M} \beta_j y_{n-j} \tag{5.3.1}$$

M is also the order of the predictor, determined by the number of preceding samples taken into account for the estimation. If $M = 1$, the first-order linear predictor is equivalent to the ZOP described in Appendix B, being the simplest possible DPCM system. The weighting coefficients β_j are chosen so that the mean square error between the predicted and the actual sample value is minimum. Note that the complexity of a DPCM system is directly related to that of the predictor algorithm employed ([101]).

In the case of M being equal to 1, and trivially, $\beta_1 = 1$, the method is called *delta coding*, in which the actual signal is replaced by its first-difference:

$$\hat{y}_n = y_{n-1}$$

$$e_n = y_n - \hat{y}_n = y_n - y_{n-1} \tag{5.3.2}$$

Taking advantage of waveform continuity and the presence of several isoelectric segments, ECG compression schemes based on delta coding have been described in literature reporting good results of CR and SNR, though a more advanced approach called *delta coding with threshold* proposed in [102] and based in *delta coding* plus tolerance-comparison principles, claims to obtain still better performance.

Usually linear predictors of order higher than two are not used since they don't result in substantial increase of compression ratio wherein desired reconstructed signal quality.

In the following Section we present the DPCM system employed in this work.

5.4 DPCM System proposed

Figure 5.2 shows the block diagram of the classic DPCM encoder, where:

$x(n)$ = actual sample value of the signal
$\hat{x}(n)$ = predicted sample value
$e(n) = x(n) - \hat{x}(n)$ = error (residual) signal
$e_q(n)$ = quantized error signal
$\tilde{x}(n) = e_q(n) + \hat{x}(n)$ = input signal for the predictor $P(\omega)$

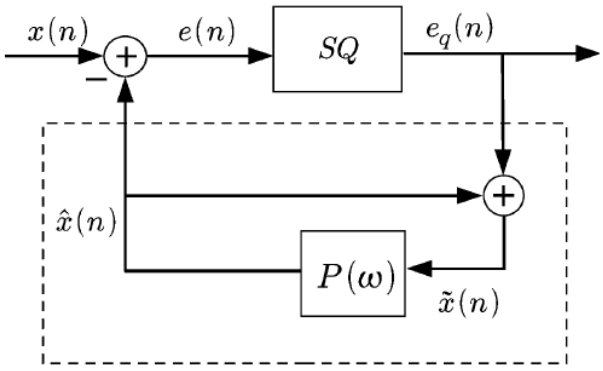


Figure 5.2. Block Diagram of the closed-loop DPCM Encoder with prediction filter $P(\omega)$ and scalar quantizer SQ.

This generic *closed-loop DPCM* encoder ([103]) consists of a *scalar quantizer SQ* and a feedback prediction loop with a *predictive filter* $P(\omega)$, on which operation we will focus in the next Section. The quantizer operates on the error signal result of the comparison between the discrete time input sequence $x(n)$ and its predicted samples $\hat{x}(n)$. This error, quantized and coded, is then stored or transmitted becoming the representation of signal $x(n)$ in this scheme. The

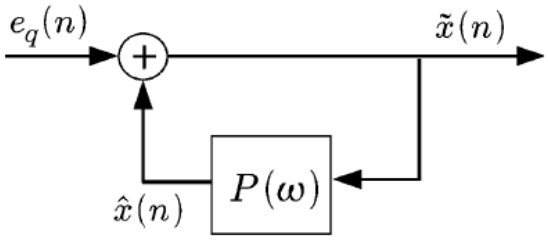


Figure 5.3. Block Diagram of the classic DPCM Decoder with prediction filter $P(\omega)$.

quantized prediction error $e_q(n)$ is combined with previously predicted samples to form the input $\tilde{x}(n)$ for the predictor, which thus receives updated sample values to make the prediction more accurate.

The decoder is embedded in the encoder and has the structure within the dashed box in Figure 5.2. It is shown again isolated in Figure 5.3. Note that the operation of the encoder and the decoder is independent from each other. The feedback loop around the quantizer in the encoder structure allows for each system having its own prediction filter and being able to work with its own independent time reference.

The structure depicted in Figure 5.2 is shown to be equivalent to the system in Figure 5.4 ([104]), where $q(n)$ is the quantization noise added to the prediction error $e(n)$ inside the closed DPCM loop, and $H(\omega) = 1 - P(\omega)$.

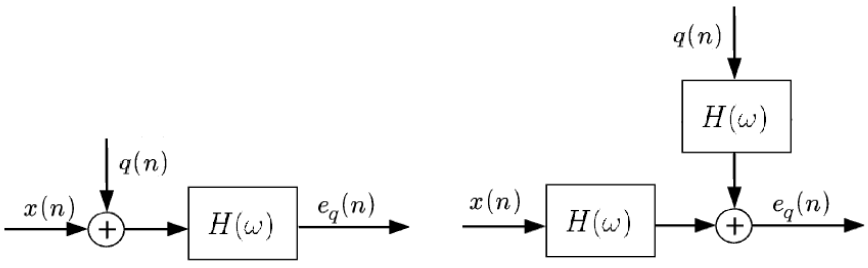


Figure 5.4. DPCM Encoder equivalent models with additive quantization noise. Reproduced from: [104].

The system described above is an improved modification of the *open-loop DPCM* scheme ([101]), shown in Figure 5.5. The encoder is the structure to the left of the dashed line, whereas the decoder is the one on its right. Note that in this case the subsystems are not independent, since the encoder uses the value of predicted samples computed in the decoder. Only the decoder includes the prediction filter, and the quantizer is located in the encoder.

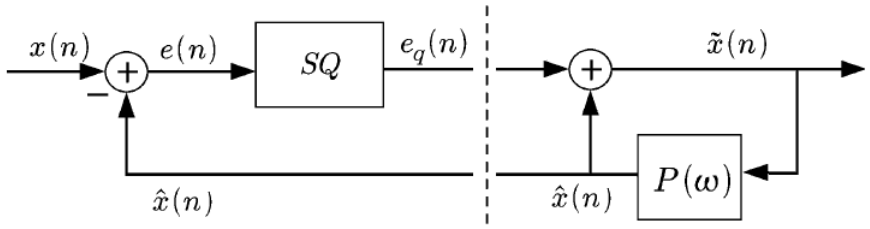


Figure 5.5. Block Diagram of the open-loop DPCM Encoder and Decoder with prediction filter $P(\omega)$ and scalar quantizer SQ.

With this *open-loop* scheme, each subsystem's time references are related, and from the point of view of the encoder, the following relations are valid:

$$\begin{aligned} x(n) &= \text{current sample value of the signal} \\ \hat{x}(n-1) &= \text{predicted sample value, received through the feedback loop} \\ e(n) &= x(n) - \hat{x}(n-1) = \text{current error (residual) signal} \\ e_q(n) &= \text{current quantized error signal} \\ \tilde{x}(n) &= e_q(n-1) + \hat{x}(n-1) = \text{reconstructed sample} \end{aligned}$$

The delay introduced by the channel affects the communication between encoder and decoder, which does not occur in the case of the *closed-loop* scheme (there, it only affects $e_q(n)$, which the coder sends through the channel and stops worrying about). Thus, in the *open-loop* scheme, the coder must wait for the prediction computed by the decoder to be locally available in order to calculate $e(n)$ and quantize it. Consequently, there is an asynchrony between the subsistems. Note that the decoder must wait a further delay to receive $e_q(n)$ and then obtain $\tilde{x}(n)$ and $\hat{x}(n)$.

In this work we use the *closed-loop* system for its reported better results.

5.5 Linear Predictor Design

The predictor filter $P(\omega)$, chosen to be a *linear predictor*, is designed to provide minimum prediction error variance σ_e^2 so that the scalar quantizer can calculate $e_q(n)$ from $e(n)$ using fewer bits for an equal SNR. Rewriting Equation 5.3.1 as:

$$\hat{x}(n) = \sum_{j=1}^{M} \beta_j \cdot \tilde{x}(n-j) \tag{5.5.1}$$

in accordance to the schemes showed above, the problem of choosing values of the coefficients β_j of the *linear predictor* arises. This analysis is often referred to as *linear predictive coding* (LPC) or *auto-regressive* (AR) modeling in fields such as speech analysis ([105]).

For being properly fitted to the input signal statistics, the weighting coefficients must be extracted in accordance to the following optimization process, in which we assume that the objective is minimizing the power of signal $e(n)$:

$$e(n) = x(n) - \hat{x}(n)$$

$$P_{e(n)} = E\left[e^2(n)\right] = \sigma_e^2 = \text{power of } e(n)$$

Establishing the optimization condition:

$$\frac{\partial E\left[e^2(n)\right]}{\partial \beta_j} = E\left[\frac{\partial e^2(n)}{\partial \beta_j}\right] = 0 \tag{5.5.2}$$

and developing the inner expression, we obtain:

$$\begin{aligned}\frac{\partial\left[e^2(n)\right]}{\partial \beta_j} &= \frac{\partial\left[(x(n) - \hat{x}(n))^2\right]}{\partial \beta_j} = \\ &= 2\,(x(n) - \hat{x}(n))\,\frac{\partial \hat{x}(n)}{\partial \beta_j} = \\ &= 2\,(x(n) - \hat{x}(n))\,x(n-j)\end{aligned}$$

Substituting this in Eq. 5.5.2:

$$E\left[2\,(x(n) - \hat{x}_{opt}(n))\,x(n-j)\right] = 0$$

$$E\left[x(n)x(n-j)\right] = E\left[\hat{x}_{opt}(n)x(n-j)\right] \tag{5.5.3}$$

where $\hat{x}_{opt}(n)$ refers to the $\hat{x}(n)$ that minimizes the power of $e(n)$. Notice that the left side of Eq. 5.5.3 denotes the correlation coefficients of signal $x(n)$, and hence it can be expressed:

$$E\left[x(n) \cdot x(n-j)\right] = R_x(j) \tag{5.5.4}$$

On the other hand, the right side of Eq. 5.5.3 can be expressed as follows, introducing Eq. 5.5.1 and in turn Equation 5.5.4:

$$\begin{aligned}E\left[\hat{x}_{opt}(n) \cdot x(n-j)\right] &= E\left[\left(\sum_{k=1}^{M} \beta_k x(n-k)\right) \cdot x(n-j)\right] = \\ &= \sum \beta_j \cdot E\left[x(n-k) \cdot x(n-j)\right] = \\ &= \sum \beta_j \cdot R_x(j-k)\end{aligned} \tag{5.5.5}$$

Finally, combining Equations 5.5.5 and 5.5.4 into 5.5.3 we obtain:

$$R_x(j) = \sum \beta_j \cdot R_x(j-k) \tag{5.5.6}$$

This set of M equations is a form of the so called *Yule-Walker equations*, and can be rewritten in matricial form as:

$$\begin{pmatrix} R_x(1) \\ R_x(2) \\ \vdots \\ R_x(M) \end{pmatrix} = \begin{pmatrix} R_x(0) & R_x(1) & \cdots & R_x(M-1) \\ R_x(1) & R_x(0) & \cdots & R_x(M-2) \\ \vdots & \vdots & \ddots & \vdots \\ R_x(M-1) & R_x(M-2) & \cdots & R_x(0) \end{pmatrix} \cdot \begin{pmatrix} \beta_{1,opt} \\ \beta_{2,opt} \\ \vdots \\ \beta_{M,opt} \end{pmatrix} \tag{5.5.7}$$

The *Yule-Walker equations* in their general form can be solved by any standard matrix inversion method. However, because of the special form of the matricial system, some efficient solutions are possible and common in literature ([106]), namely: a) the covariance method, b) the autocorrelation method, and c) the lattice method. Here we use the *autocorrelation method* with *Levinson-Durbin recursion*, an algorithm specifically designed for efficient solving of Hermitian[2] Toeplitz[3] matricial systems. Details on this algorithm can be consulted in [106].

Suffice to mention that linear predictors of order higher than two are not usually employed in ECG compression applications, since the compromise between CR and reconstructed signal fidelity does not perform better for greater orders.

5.6 Quantization

Quantization is the final step of our coding and compression system, in which the samples of the residual signal are assigned binary symbols used in storage or transmission over a channel.

There are two general approaches of quantization in lossy data compression schemes, namely, *uniform quantization* and *non-uniform quantization*. The former employs a constant bitrate for every sample, assigning to each value of the original signal the closest corresponding binary code, without analyzing the adjacent samples. The latter uses variable bitrate and is applied when the signals to be compressed are known to be more sensitive in a defined frequency range. Quantization levels are then applied non-uniformly, assigning a bigger number of levels to the margins for which the signal changes faster, since the information density is bigger there.

Logarithmic quantization is a hybrid method in which a constant bitrate is used, but as a previous step to quantization, the signal is passed through a logarithmic compressor which reduces the quantization noise. Small values of amplitude are assigned to small quantization steps, whereas big values are processed into big quantization steps, thus providing a bigger resolution for the case of weak signals in comparison to a uniform quantization, but smaller resolutions for signals of great level. A-law and μ-law are example algorithms of logarithmic compressors.

Even though *non-uniform quantization* can provide better results by analyzing the signal's entropy and dedicating finer quantization intervals for the margins in which the signal would need more data to be represented in a fix quality, *uniform quantiziers* are much more common due to their simplicity and low cost in computational terms. The tradeoff is less efficiency in the com-

[2] A Hermitian matrix (named after mathematician Charles Hermite), also called self-adjoint matrix, is a matrix with complex entries which is equal to its own conjugate transpose. Formally, an $n \times n$ matrix A is a Hermitian matrix if $A_{i,j} = \bar{A}_{j,i}$.

[3] A Toeplitz matrix (named after mathematician Otto Toeplitz), also called diagonal-constant matrix, is a matrix in which each descending diagonal from left to right is constant. Formally, an $n \times n$ matrix A is a Toeplitz matrix if $A_{i,j} = A_{i-1,j-1}$.

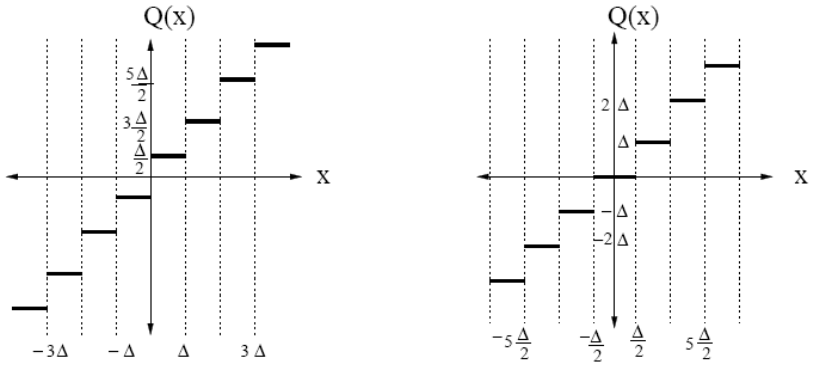


Figure 5.6. Uniform quantizers variants: left, mid-rise quantizer; right, mid-tread quantizer

pression due to the fact that the quantization noise probability is proportional to the increase in the signal's amplitude.

We will use *uniform quantization* in this work, so we now focus on that approach. In *uniform quantizers* the decision boundaries are uniformly spaced:

$$B_i - B_{i-1} = \Delta, \quad \forall i = 2, 3, ..., N-1$$

where N is the number of quantization intervals and $\Delta = 2^{-n}$ is the *quantization step*, being n the number of bits used for the coding. The representative values are given by:

$$Q(x) = \begin{cases} B_1 - \Delta/2, \text{if} x \leq B_1 \\ B_i + \Delta/2, B_i < x \leq B_{i+1}, \quad \text{where } i \in 1, .., N-1 \end{cases}$$

There are two common variants of uniform quantization, called *mid-rise* and *mid-tread* uniform quantizers, depending on how the quantization is performed about the value of zero. The quantization values $Q(x)$ are plotted in Figure 5.6 for these two variants as a function of the original samples values.

In the case of mid-rise quantizers N is even and $B_{\frac{N}{2}} = 0$. The values of B_i are multiples of Δ, and the quantization vales $Q(x)$ are multiples of $\Delta/2$.

For mid-tread quantizers N is odd and $B_{\frac{N-1}{2}} = -\frac{\Delta}{2}$. The values of B_i are multiples of $\Delta/2$ and those of $Q(x)$ are multiples of Δ, being $Q(x) = 0$ for $x \in \left[-\frac{\Delta}{2}, \frac{\Delta}{2}\right]$. In this case, the quantization levels are symetric around zero but there is no zero quantization level (thus, the quantized signal must be either positive or negative).

We have implemented a mid-rise uniform quantizer with selectable number of bits in the function `quantizer_encoder.m` and the opposite system in `quantizer_decoder` to perform the complementary tasks in coding an decoding steps carried out by functions `dpcm_encoder.m` and `dpcm_decoder.m` respectively. All these functions are listed in Appendix D.

A comment on the possibility of further channel coding by means of lossless techniques (Huffman variable-length codes, arithmetic encoding, etc.) is included in Chapter 7. These have not been included in this work since their implementation simply means one additional encoding step, of usually well-known indisputable benefit, and for space and content reasons. Readers interested in these improvements, often employed as the last part of practical compression and coding systems, are referred to the suggested literature.

5.7 Compressibility Analysis

As it is known and explained in Section 3.1, the smaller the entropy of a data source is, the more compressible the sequences generated by it will be, since the entropy measures the expected value of self-information, lower information meaning high redundancy. Moving this rationale to practical compression schemes, such as the DPCM system proposed in previous sections, the smaller the sample entropy of a sequence is, the bigger CR we will be able to attain when compressing that sequence for some specific distorsion measure bounds, or equivalently, the smaller distorsion we will introduce for a fixed compression ratio.

Thus, it would be convenient to know if we can get better compression-distorsion results with the system we have designed for other signals different than the ECG; not necessarily with the intention of deciding to transmit such signals if the results are better, but to complementarily include them in the transmission. As one of the aims of ULTRASPONDER project is to efficiently manage these several types of signals, we will conduct an analysis to find out which sequences are better compressed.

This study will be done in terms of distorsion analysis for a fixed compression ratio (number of bits for the residual quantization in the DPCM encoder) evaluating the entropy and the kurtosis of each signal for comparison and reference purposes, and employing PRD and SNR as fundamental measures.

For merging the information about the theoretical compression limit that could be attained on a signal, and the actual results obtained after DPCM encoding, we introduce a novel measure that collects the sample entropy of a sequence and the PRD resultant of the compression. As discussed, smaller entropy should provide bigger compression for a fix PRD, and conversely, smaller PRD for a fix compression ratio. Hence, both parameters are inversely related to this compressibility measure we are proposing.

Although it has no immediate or evident physical meaning, it measures the "compressibility" of a sequence, not theoretically, like entropy could do, neither as a result, as the compression ratio assesses, but collecting information from both in a balanced way, and so we will call it "*compressibility quotient*", CQ, defining it as:

$$CQ = \frac{10}{H \cdot PRD} \tag{5.7.1}$$

We will present results and derivated comments about this measure in Section 6.6 and Chapter 7.

Part III

Results, Discussion and Conclusion

Insanity is doing the same thing over and over again and expecting different results.

Albert Einstein, 1879-1955

Chapter 6

Results

In this Chapter we condense the results of most of the research, processing and simulations that this Thesis has comprised. Numerical data are provided, along with related figures when suitable. Brief commentaries accompany the results; a more general discussion will be carried out in Chapter 7.

Section 6.1 covers the mathematical and statistical analysis performed on some signals of diverse nature that cohabit the Ultrasponder system. The different sensors provide signals with distinct typical morphologies, artifact problematic, etc., each carrying an important diagnostic information by itself that none of the others may provide. Nevertheless, these signals are most likely correlated; they are, in their physiological origin. With our study we try to find out if there are underlying mathematically provable correlations and dependences.

Section 6.2 is an insight on the linear filters employed in Zeelenberg and Tompkins' QRS complexes detection systems. The different filters are characterized in terms of their difference equations, coefficients, impulsive and frequency response, etc.

The results obtained by the detection systems are presented in Section 6.3, and those of the beats classifiers are covered by Section 6.4.

Lastly, 6.5 reports the results of thr heart rate variability analysis performed, and 6.6 collects everything related to the DPCM, compression and quantization simulations.

6.1 Multisignal Mathematical Analysis

In this Section we present the results of a rudimentary approach to studying the mathematical relations underlying some different types of signals that must cohabit the Ultrasponder control unit, as discussed in Chapter 1.

For this purpose, coherently correlated signals of this diverse nature were needed. We have carried out some experiments using the records of the MGH/MF Waveform Database, which contain ECG leads, arterial pressure (ART), pulmonary arterial pressure (PAP) and central venous pressure (CVP) waveforms, among others.

The study consisted in analyzing the relations among these four different types of signals by trying to extract conclusions from the calculations of parameters such as marginal entropy, conditional and joint entropy, mutual information and correlation coefficient, calculating also the p-values for testing the hypothesis of no correlation and the bounds for 95% confidence intervals.

Results are omitted in this Section and presented in Appendix C in order not to break the continuity of the text. Test signals were chosen to be excerpts of 8000 to 20001 samples of records included as said above in the MGH/MF Waveform Database.

Computation of excess kurtosis is also included for completeness of statistical characterization reasons, but not much information about the correlation between signals can be extracted from it. The cases in which it is high show a bigger peakedness in the signal, but a waveform morphology analysis does not suffice to assess the information content of the sequence.

Figures 6.1 to 6.6 depict the signals under test for 10 seconds excerpts of six arbitrary records of the mentioned database, for which the numerical results are included in Appendix C (note in the figures artifacts like baseline drifting, noisy beats, interferences in the isoelectric intervals, etc). The order of the signals employed in those results reports is: ECG, ART, PAP, CVP. When the parameters involve two signals, the order of the reported data is: relation ECG-ART, relation ECG-PAP, relation ECG-CVP, relation ART-PAP, relation ART-CVP and finally relation PAP-CVP.

Results are given in the following manner:

- `entropies` is an array containing the entropy of each signal.
- `kurtosix` is an array containing the excess kurtosis of each signal.
- `rho` is an Hermitian matrix containing the crosscorrelation coefficients of each pair of signals. The elements of the diagonal obviously equal 1.
- `P` is an Hermitian matrix containing the p-values for the statistical test of no correlation hypothesis. The smaller the value of an element of P, the more likely it is that there exists a true correlation between the corresponding signals, hence being more consistent the associated value of `rho`.
- `rlo` and `rup` are matrices containing lower and upper bounds for a 95% confidence interval for each coefficient.
- `mutualI` is again an Hermitian matrix containing the mutual information of the corresponding pair of signals. The elements of the diagonal are equal to the `entropies` array, since of course $I(X;X) = H(X)$.
- `condentropies`, likewise, contains the conditional entropies. The elements of the diagonal equal 0, since the conditional entropy of a variable over itself is evidently zero.

- `jointentropies`, likewise, contains the values of joint entropies between each pair. Again, the elements of the diagonal coincide with the `entropies` array, since $H(X, X) = H(X)$.

From the results obtained (see App. C), the observation of the fact that there must be a somehow strong correlation between the magnitude of the ECG signal and the pulmonar arterial pressure, as well as between the latter and the arterial pressure arises easily. There also seems to be a certain linear dependence between the ECG time series and the central venous pressure, but it seems to be not so strong. The same occurs between arterial pressure and central venous pressure, according to the values of the correlation coefficients and the p parameter of the p-tests.

Also, one can see how the ART and PAP signals generally show a bigger marginal entropy than the ECG and CVP. This fact leads to reasoning that the ECG and CVP signals ought to be easier to compress, since they appear to be more redundant sources.

At the same time, the ECG proves to attain a highest index of mutual information with the ART pressure signal, being generally the maximum element of the mutual information matrices. Then, the PAP and lastly the CVP. The contrary occurs for the case of conditional entropy, though exceptions appear. This is a sign that the information shared between the ECG and ART pressure is significant and thus the observation of them alltogether yields clarification of uncertainty and carries a big amount of information. Therefore, if by transmitting these two signals we could grant the preservation of diagnostically useful information, that seems to be the most reasonable choice. If however the content of information carried by ART is not more useful for clinical diagnosis than that contained in CVP, being the latter easier to compress, the election ought to be changed.

The results obtained suggest that efficient compression could be obtained on these signals by lowering the number of bits representing each sample to an amount close to the limits bound by entropy values. In light of this, and the morphology of the ECG waveform, which is the fundamental signal to be processed, we choose a DPCM system for this task.

6.2 Linear Filter Analysis

In this section we present a detailed analysis of the linear filters employed in our QRS detection systems. The detector based on the Zeelenberg algorithm is covered in 6.2.1, and 6.2.2 discusses the one based on Tompkins' method.

6.2.1 Zeelenberg Algorithm

As described in 4.3, Zeelenberg algorithm uses two cascaded filters before the decision stage: a notch filter designed to cancel the power-line interference, and

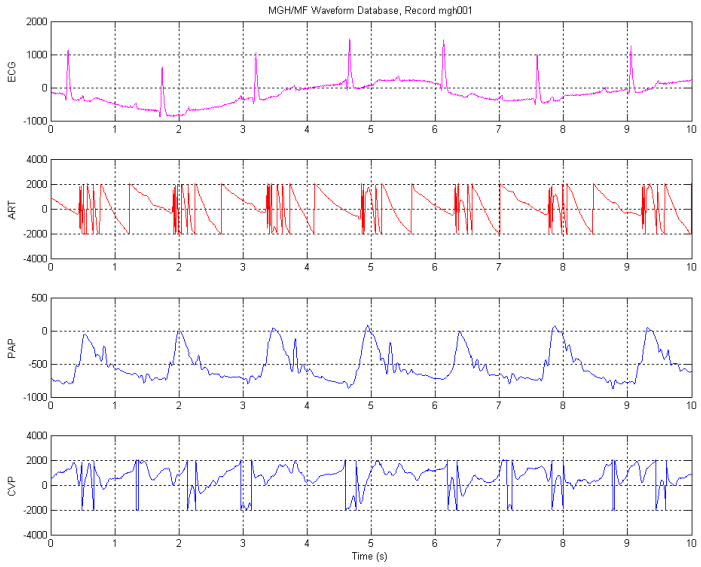


Figure 6.1. ECG, arterial pressure (ART), pulmonary arterial pressure (PAP) and central venous pressure (CVP) for a 10 s excerpt of record mgh001 of MGH/MF Waveform Database.

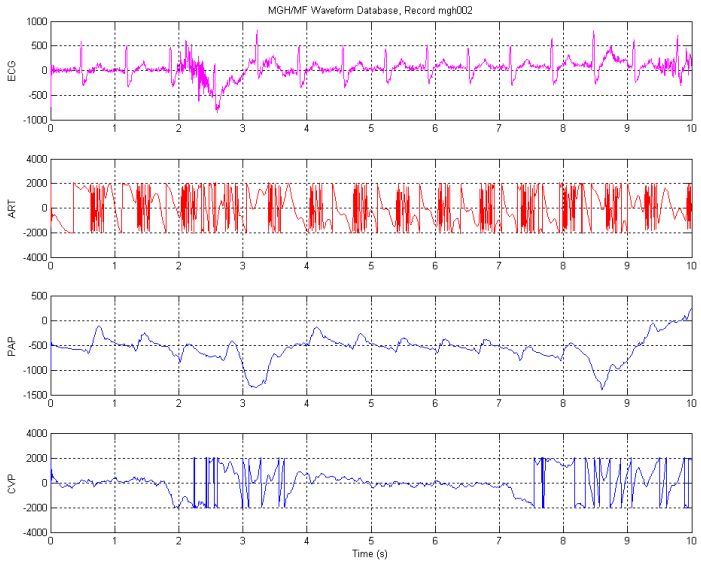


Figure 6.2. ECG, arterial pressure (ART), pulmonary arterial pressure (PAP) and central venous pressure (CVP) for a 10 s excerpt of record mgh002 of MGH/MF Waveform Database.

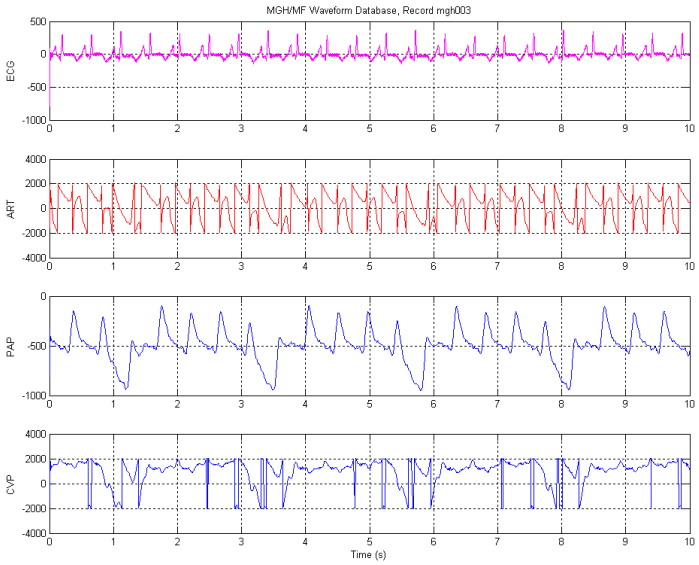


Figure 6.3. ECG, arterial pressure (ART), pulmonary arterial pressure (PAP) and central venous pressure (CVP) for a 10 s excerpt of record mgh003 of MGH/MF Waveform Database.

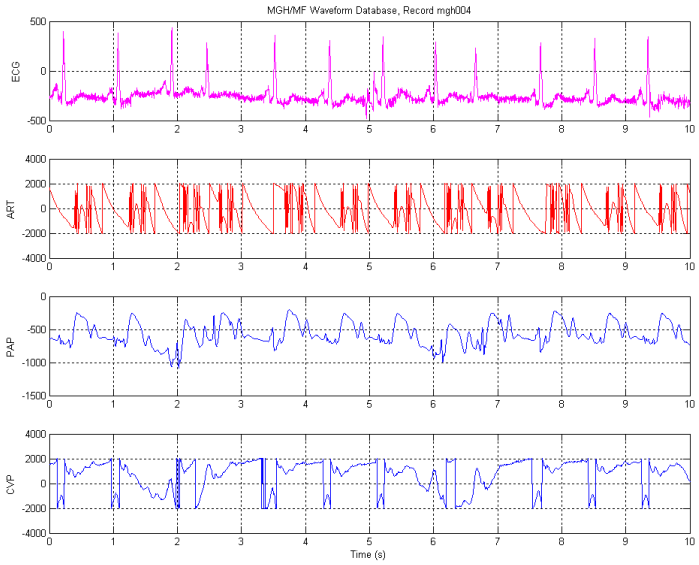


Figure 6.4. ECG, arterial pressure (ART), pulmonary arterial pressure (PAP) and central venous pressure (CVP) for a 10 s excerpt of record mgh004 of MGH/MF Waveform Database.

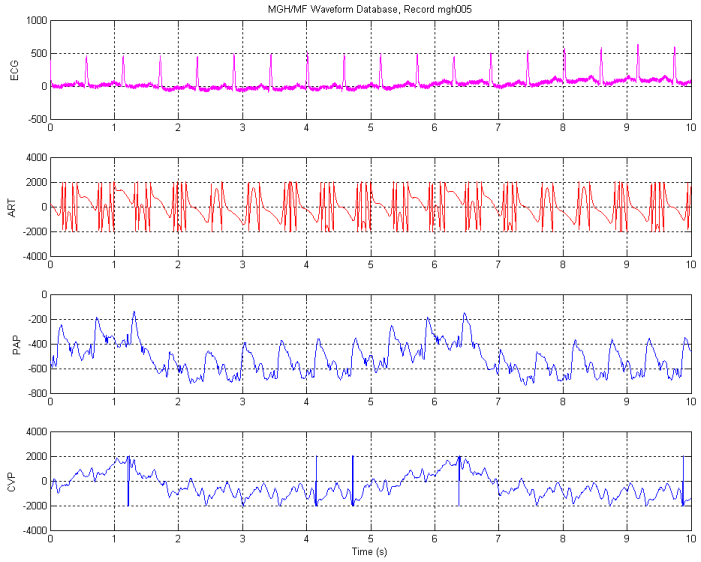


Figure 6.5. ECG, arterial pressure (ART), pulmonary arterial pressure (PAP) and central venous pressure (CVP) for a 10 s excerpt of record mgh005 of MGH/MF Waveform Database.

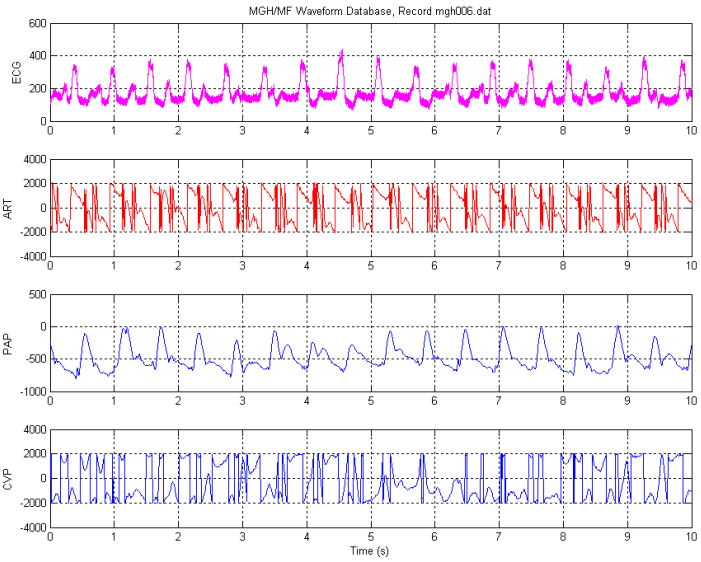


Figure 6.6. ECG, arterial pressure (ART), pulmonary arterial pressure (PAP) and central venous pressure (CVP) for a 10 s excerpt of record mgh006 of MGH/MF Waveform Database.

a low-pass filter for removing the DC component and noise due to EMG and P- and T-waves. Both of them are designed for a sampling frequency of 250 Hz.

The difference equation that describes the notch filter is:

$$y_0(n) = x(n) - x(n-4) \tag{6.2.1}$$

Thus, its transfer function can be expressed as:

$$H_1(z) = \frac{Y_0(z)}{X(z)} = \frac{1 - z^{-4}}{1} = 1 - z^{-4} \tag{6.2.2}$$

We can see that it is a 4th-order FIR filter, and its coefficients are:

$$B_1 = \begin{bmatrix} 1 & 0 & 0 & 0 & -1 \end{bmatrix}$$

$$A_1 = \begin{bmatrix} 1 \end{bmatrix}$$

The difference equation that describes the low-pass filter is:

$$y_1(n) = y_0(n) + 4y_0(n-1) + 6y_0(n-2) + 4y_0(n-3) + y_0(n-4) \tag{6.2.3}$$

The transfer function can be written as:

$$\begin{aligned} H_2(z) = \frac{Y_1(z)}{Y_0(z)} &= \frac{1 + 4z^{-1} + 6z^{-2} + 4z^{-3} + z^{-4}}{1} \\ &= 1 + 4z^{-1} + 6z^{-2} + 4z^{-3} + z^{-4} \end{aligned} \tag{6.2.4}$$

This is as well a 4th-order FIR filter, whose coefficients are:

$$B_2 = \begin{bmatrix} 1 & 4 & 6 & 4 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 1 \end{bmatrix}$$

In the following Figures 6.7 and 6.8 we can see the frequency response of the filters in magnitude and phase, as well as their group delay and impulse response. The plots have been obtained using the function `see_filter.m`, listed in Appendix D. These filters have been designed for a sample frequency of 250 Hz. Because of the fact of being FIR filters, they are inherently stable, since their impulse responses are absolutely summable (they last for 5 samples, and then extinguish). The fact that their impulse responses are symmetric allows for a linear phase response, and the group delay being constant and equal to the time simmetry point of the impulse responses. Other features of interest of the filters are showed in Table 6.1.

The notch filter has been designed for rejecting the 60 Hz component corresponding to the fundamental frequency of power-line interference in the

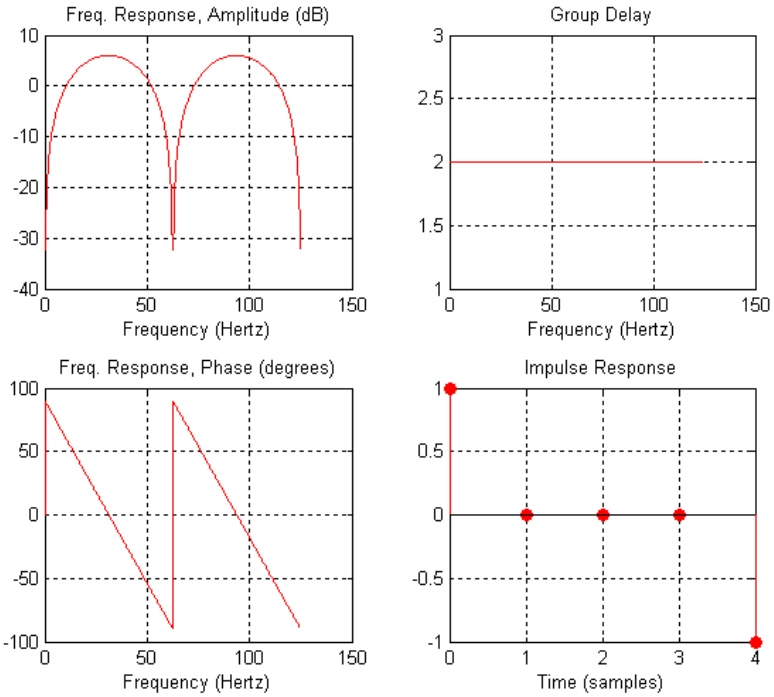


Figure 6.7. Notch filter of Zeelenberg detector

Filter	DC Gain (dB)	$f_{c,-3\ dB}$ (Hz)	Gr. delay (samples)
Notch Filter	$-\infty$ @ 0 Hz -32.2014 @ 0.2441 Hz	15.8691 46.6309 78.3691 109.1309	2
Low-pass Filter	24.0824	32.4707	2

Table 6.1. Characteristic features of Zeelenberg detector's filters

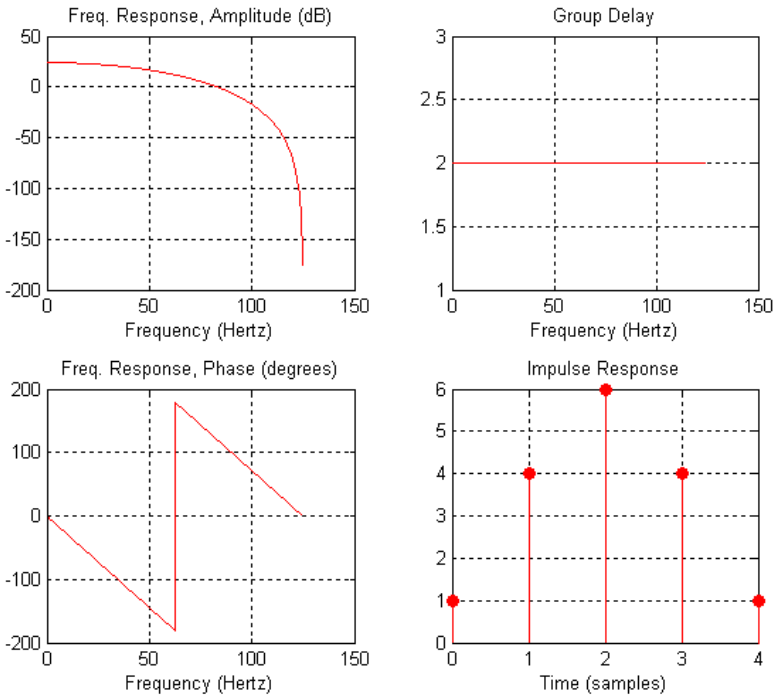


Figure 6.8. Low-pass filter of Zeelenberg detector

U.S.A., therefore being applicable to the MIT-BIH collections (see Sections 2.6 and 4.2). It reaches the minimum of its response for 62.74 Hz with -32.2 dB.

Figure 6.9 shows the application of these two filters on the ECG signal of record 210 of MIT-BIH Arrhythmia Database. A shortened interval of 12 seconds is considered, in which as it can deduced from the original ECG shown in the first row, there are several 'normal beats' and an abnormal one, the seventh. Besides, there is considerable baseline drift, and one can appreciate a certain R-R rate variability, particularly in the onset of the 4th beat in comparison to the 6th, for example. The second row shows the output signal of the notch filter, after which the power-line interference should be attenuated, and in the third row we see the final output of the filtering stage, which would be presented to the peak detection and decision stage to extract the exact instants of onset or offset of the QRS complexes. One can see how the processing is aimed to shape the waveform so that the peak detector can achieve a better performance. Although abnormal, the seventh beat does contain a QRS complex that the system should be able to detect. The updating mechanism of the adaptive threshold would determine if this QRS pressence is actually identified or on the contrary it would be missed. Results of detection will be provided in Section 6.3.

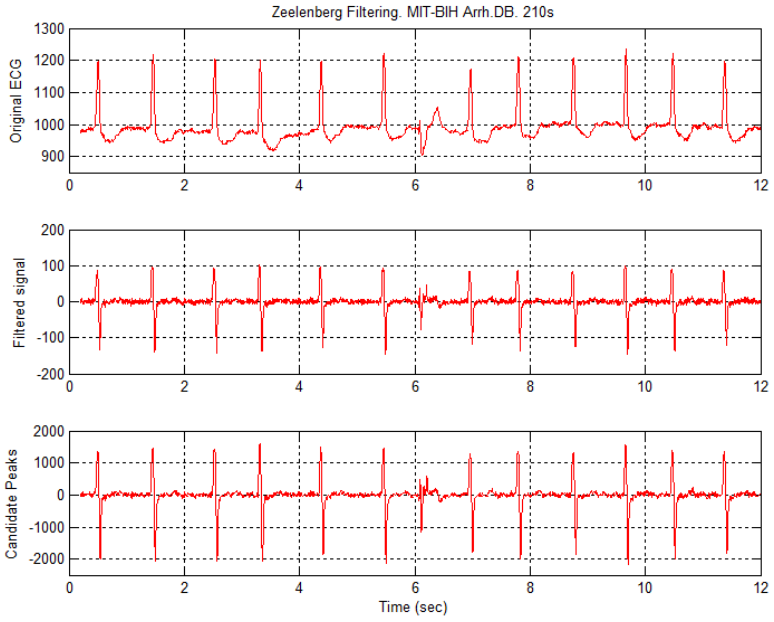


Figure 6.9. Zeelenberg filtering stage applied to an excerpt of record 210 of MIT-BIH Arrhythmia Database

6.2.2 Tompkins Algorithm

The linear filtering that takes part in Tompkins algorithm is done in three stages as described in 4.3. A low-pass, a high-pass and a derivative filter are cascaded

in order to produce the signal needed for the following decision stage. All these filters have been designed for a sample rate of 250 Hz as well.

The characteristic difference equation of the initial low-pass filter is:

$$y_0(n) = x(n) - 2x(n-7) + x(n-14) + 2y(n-1) - y(n-2) \tag{6.2.5}$$

After some simple operations, the transfer function can be compactly expressed as:

$$H_1(z) = \frac{Y_0(z)}{X(z)} = \frac{(1-z^{-7})^2}{(1-z^{-1})^2} \tag{6.2.6}$$

It is a 14th-order FIR filter, which coefficients are:

$$\begin{aligned} B_1 = [1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad -2 \quad ... \\ ... \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1] \end{aligned}$$

$$A_1 = \begin{bmatrix} 1 & -2 & 1 \end{bmatrix}$$

The high-pass filter in Tompkins detector is expressed in terms of an auxiliary low-pass transfer function given by:

$$H_{LP}(z) = \frac{1-z^{-32}}{1-z^{-1}} \tag{6.2.7}$$

such that:

$$H_2(z) = \frac{Y_1(z)}{Y_0(z)} = z^{-16} - \frac{H_{LP}(z)}{32} \tag{6.2.8}$$

Sustituting Equation 6.2.7 in 6.2.8 and developing, we can find the following expression for the difference equation of the high-pass filter, which is a 32nd-order FIR filter:

$$32y_1(n) = -y_0(n) + 32y_0(n-16) - 32y_0(n-17) + y_0(n-32) + 32y_1(n-1) \tag{6.2.9}$$

which provides the subsequent coefficients:

$$\begin{aligned} B_2 = [-1 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad ... \\ ... \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad ... \\ ... \quad 32 \quad -32 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad ... \\ ... \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 1] \end{aligned}$$

$$A_2 = \begin{bmatrix} 32 & -32 \end{bmatrix}$$

Finally, the derivative filter responds to the next transfer function:

$$H_3(z) = \frac{Y_2(z)}{Y_1 z} = 0.1(2 + z^{-1} - z^{-3} - 2z^{-4}) \tag{6.2.10}$$

and difference equation:

$$y_2(n) = 0.1(2y_1(n) + y_1(n-1) - y_1(n-3) - 2y_1(n-4)) \tag{6.2.11}$$

It is therefore a 4th-order FIR filter with coefficients given by:

$$B_3 = \begin{bmatrix} 0.2 & 0.1 & 0 & -0.1 & -0.2 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 \end{bmatrix}$$

Figures 6.10, 6.11 and 6.12 show the frequency response, group delay and impulse response of the low-pass filter, high-pass filter and derivative filter employed in Tompkins detector, respectively. Again, all the filters are FIR and therefore stable. The phase response is found to be linear, which is in agreement to the fact that they have constant group delay. The glitches found in the plot of the group delay of the low-pass filter are of a very small magnitude, and occur exactly in the rejected frequencies of the comb-response[1], points in which the linear phase suffers stepwise discontinuities. As we see from Table 6.2, the group delay of this filter can be considered constant without significant loss of accuracy.

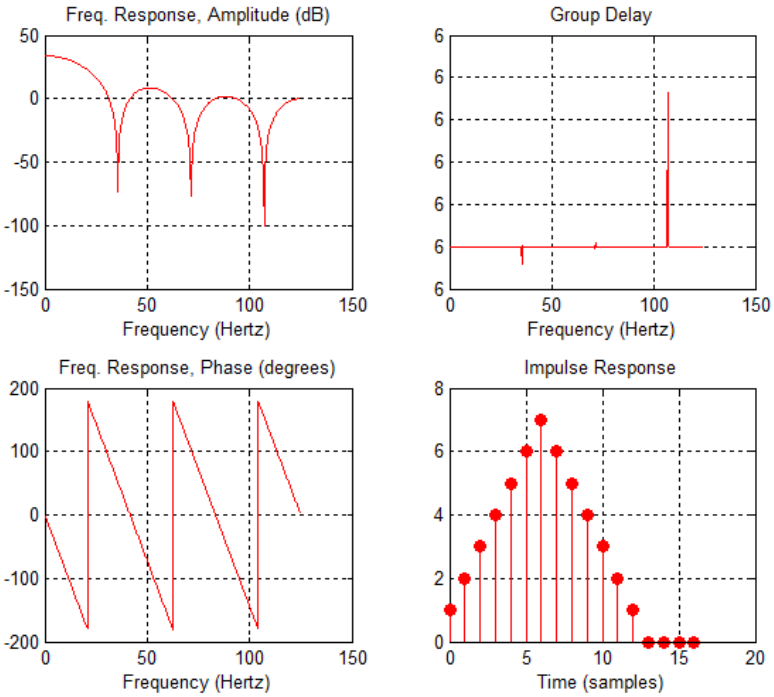


Figure 6.10. Low-pass filter of Tompkins detector

In the case of the high-pass filter a brief transient is observed in the plots; this is due to the abrupt slope of its response, being a filter of a considerably high order. Once the phase response is linearized, after just the first samples, the group delay reaches a margin centered around a value of 16 in which it keeps

[1] Notch filters whose nulls repeat over the spectrum at multiples of a fundamental rejection frequency are called *comb-filters*.

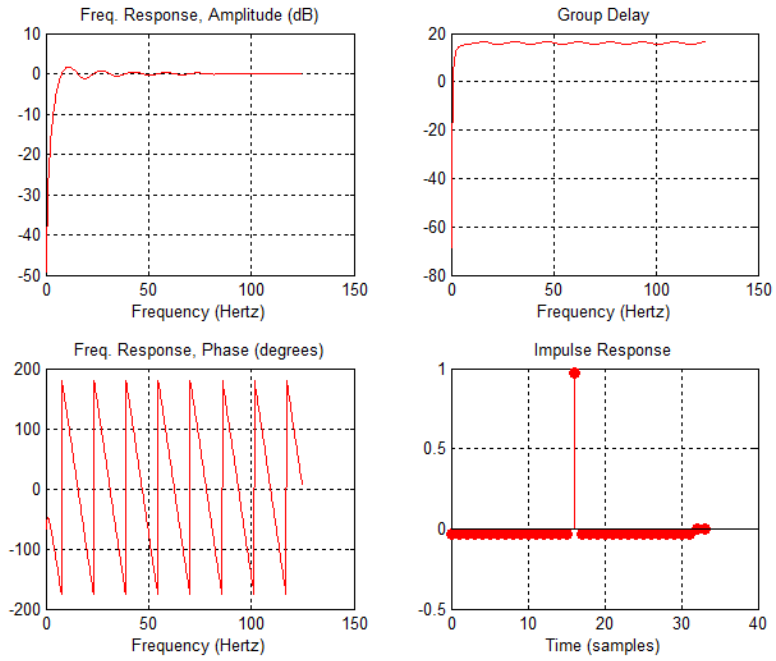


Figure 6.11. High-pass filter of Tompkins detector

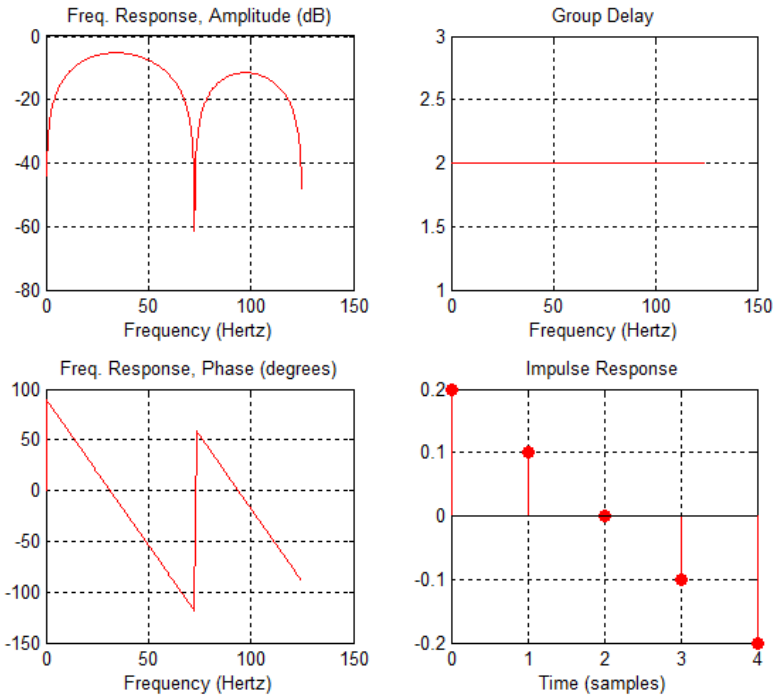


Figure 6.12. Derivative filter of Tompkins detector

Group delay statistics	
Mean (samples)	6.000000000320828
Median (samples)	6.000000000000013
Standard deviation (samples)	8.073994438300113 e-009
Variance (samples^2)	6.518938618970117 e-017

Table 6.2. Statistical parameters of the low-pass filter group delay in Tompkins scheme

constrained. However, this fluctuation and the initial rise reflect in the mean and the variance, as can be seen from Table 6.3; not so in the median, a much more robust statistic. See also how the standard deviation and variance have been enlarged in comparison to the case of the low-pass filter.

Group delay statistics	
Mean (samples)	15.417962762282638
Median (samples)	15.949741458818528
Standard deviation (samples)	5.316670970711032
Variance (samples^2)	28.266990210801382

Table 6.3. Statistical parameters of the high-pass filter group delay in Tompkins scheme

The derivative filter does not arouse any similar comments, since its response is perfectly regular and as expected. Table 6.4 describes the characteristics of the three filters employed in Tompkins algorithm.

Filter	DC Gain (dB)	$f_{c,-3\ dB}$ (Hz)	Gr. delay (samples)
Low-pass Filter	33.8026	11.4746	6
High-pass Filter	$-\infty$ @ 0 Hz -29.7069 @ 0.2441 Hz	5.8594	16
Derivative Filter	$-\infty$ @ 0 Hz -44.2426 @ 0.2441 Hz	17.0800 52.4900	2

Table 6.4. Characteristic features of Tompkins detector's filters

Figure 6.13 shows the successive steps in the filtering stage of Tompkins detection algorithm. A reduced interval of 10 s retrieved from signal 100 of the MIT-BIH Arrhythmia Database is presented in the first row. The result of the low-pass filtering in showed in the second row, where we can see that the high frequency components that were present in the original ECG have disappeared. In the next step, the high-pass filtering emphasizes the QRS

complexes attenuating the amplitude of other ECG waves in order to make the detection easier. Finally, in the fourth row we can observe the waveform of the output of the derivative filter, which is prepared to be provided to the peak detection stage, studied in the next Section.

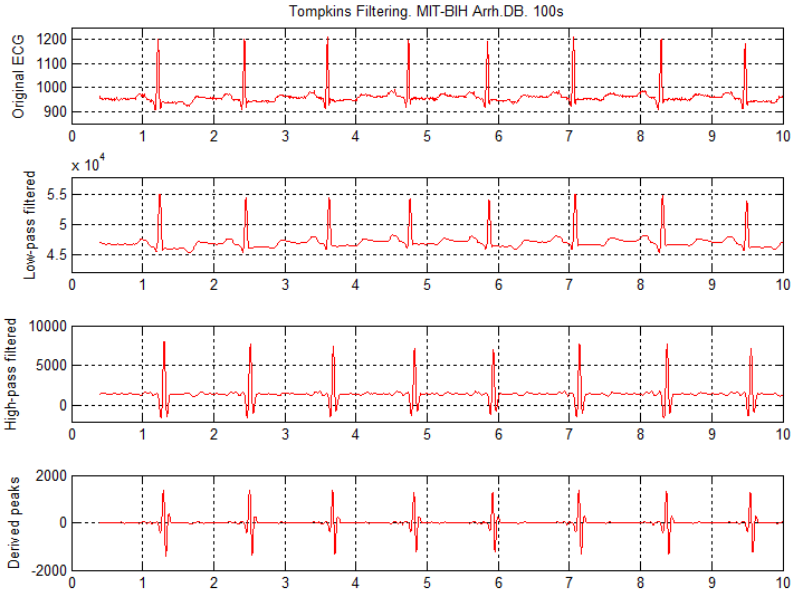


Figure 6.13. Tompkins filtering stage applied to an excerpt of record 100 of MIT-BIH Arrhythmia Database

6.3 QRS Detection

After having studied the operation of the filters employed in Zeelenberg and Tompkins algorithms, we present in this Section the results achieved by the two QRS detection systems implemented.

From their TP, TN, FP and FN results, the following parameters are calculated according to the definitions given in 4.4: sensitivity (Se), positive predictivity (PP), specifity (Sp) and false alarm rate (FPR, false positive rate). Two other parameters are computed for completeness of the report and avoiding the bias of the previous rates, namely Q_α and Matthews correlation coefficient (MCC).

All these calculations are performed by functions `QRSdeteval.m` and `QRSdetevalaux.m`, listed in Section D.2. `QRSdeteval.m` receives as input arguments the sequence of locations where QRS complexes have been identified by the algorithm to be evaluated, and the name of the record file and folder in which this file is located, needed for retrieving the correspondent annotations.

It compares the sequence of QRS locations provided with the sequence given by the annotations, accepting a slight disalignment controlled by a window length defined with a *tolerance* parameter, and classifies each position as a *TP*, *FP*, *TN* or *FN* according to the matching of the two sequences. It then calculates their total count and obtains the percentages and performance measures Se, PP, Sp, FPR, Q_α and MCC.

`QRSdetevalaux.m` has been programmed in order to automate this evaluation process, launching it sequentially over a defined number of signals and providing a useful output for performance comparison. It internally calls the specified detection algorithm and uses `QRSdeteval.m` for calculating the performance parameters, for each of the record files selected by the user. By default, if Zeelenberg detector is used the analysis is done over the full length of the whole record of lead I, whereas in the case of Tompkins detector we have chosen to employ a 10000-samples excerpt (27.78 s for a sampling frequency of 360 Hz) of the signal contained as well in lead I of the records, in order to reduce processing time, but all this can be varied. Experiments carried out on the two leads provided in the MIT-BIH Arryhthmia Database records showed that in most cases, better results are obtained performing the detection on lead I, with very few exceptions.

Tables 6.5, 6.6 and 6.7 show the tabulated evaluation results of the two detectors. Figures 6.14, 6.15 and 6.16 display boxplots[2] of the values of each parameter for each algorithm. Figure 6.17 shows a comparison. Details and discussion will be covered in Chapter 7.

6.4 Beat and Arrhythmias Classifier

We present here the results of the beat classifiers described in Section 4.5. As we now focus only in the correlation threshold comparison, and not in the QRS complexes detection itself, which has been discussed before, the QRS positions sequence input for these classifiers can be any coherent sequence retreived by any means. We use in these experiments the MIT-BIH annotations of the corresponding records. This is done by the program `Atrreader` which provides an interface between the ATR files (where the annotations are stored) and the MATLAB environment.

In the following tables, Classifier I is the algorithm implemented in function `beatsclassifier.m` and Classifier II is the modified version implemented in function `beatsclassifiermod.m`, both listed in Appendix D. The test database chosen for these experiments is the MIT-BIH Arrhythmia Database, from which 650000-samples long excerpts of the records have been taken, corresponding to approximately 30 min for a sampling frequency of 360 Hz.

[2]Boxplots include visual representations of the median (red line), interquartile range (blue box), apparent minimum and maximum (black whiskers) disregarding outliers, and outliers themselves (red crosses). They constitute a useful visual tool collecting central tendency and dispersion measures for statistical analysis of data samples. Turn to Appendix A for reference on these statistics.

Record	Det. B.	Ann. B.	Se	PP	Sp	FPR	Q_α	MCC
100	2274	2274	99.96	99.96	94.74	5.26	97.35	0.9469
101	1870	1874	99.57	99.79	84.62	15.39	92.09	0.7846
102	2188	2192	96.76	96.94	53.79	46.21	75.28	0.4992
103	2085	2091	99.71	100.00	100.00	0.00	99.86	0.8807
104	2252	2311	91.52	93.92	63.27	36.73	77.40	0.5157
105	2641	2691	94.05	95.84	63.46	36.55	78.76	0.5376
106	1787	2098	84.80	99.55	97.40	2.60	91.10	0.6283
107	2133	2140	24.67	24.75	85.92	14.08	55.30	0.1060
108	1872	1823	1.04	1.02	66.68	33.32	33.86	−0.3200
109	2531	2535	98.07	98.22	68.75	31.25	83.41	0.6596
111	2126	2133	90.76	91.06	52.85	47.15	71.81	0.4332
112	2568	2550	55.22	54.83	96.34	3.66	75.78	0.5139
113	1796	1796	99.94	99.94	85.71	14.29	92.83	0.8566
114	1884	667	0.15	0.05	94.08	5.92	47.12	−0.0350
115	1954	1962	10.96	11.00	95.50	4.50	53.23	0.0647
116	2389	2421	96.57	97.87	59.20	40.80	77.89	0.5010
117	1538	1539	98.31	98.38	66.22	33.78	82.26	0.6412
118	2314	2301	0.39	0.39	85.83	14.17	43.11	−0.1374
119	1989	2094	94.89	99.90	98.18	1.82	96.54	0.6831
121	1866	1876	99.04	99.57	93.10	6.90	96.07	0.8865
122	2477	2479	99.92	100.00	100.00	0.00	99.96	0.7068
123	1519	1519	16.79	16.79	96.56	3.44	56.67	0.1334
124	1613	240	99.17	14.76	1.86	98.14	50.51	0.0279
200	2653	2792	62.75	66.04	93.90	6.10	78.33	0.5785
201	1954	1	0.00	0.00	25.11	74.90	12.55	−0.0338
202	2132	1062	97.83	48.73	31.43	68.57	64.63	0.3602
203	2927	1	100.00	0.03	35.14	64.86	67.57	0.0110
205	2653	2672	96.82	97.51	22.35	77.65	59.59	0.1739
207	2132	47	40.43	0.89	11.96	88.04	26.19	−0.1952
208	2640	3040	85.76	98.75	79.63	20.37	82.69	0.3767
209	3017	713	59.89	14.15	93.51	6.49	76.70	0.2674
210	2619	1	0.00	0.00	2.64	97.36	1.32	−0.1163
212	2752	2763	98.15	98.55	46.67	53.33	72.41	0.4194
213	3247	3294	97.45	98.86	68.64	31.36	83.05	0.5630
217	2192	229	9.61	1.00	9.17	90.83	9.39	−0.6217
219	2153	1	0.00	0.00	4.52	95.48	2.26	−0.0963
220	2048	484	79.75	18.85	95.86	4.14	87.81	0.3750
221	2414	1	0.00	0.00	14.91	85.09	7.46	−0.0448
222	2484	673	98.96	26.81	11.45	88.55	55.20	0.1578
223	2587	2643	95.95	98.03	69.28	30.72	82.61	0.5701
232	1788	1	0.00	0.00	26.18	73.82	13.09	−0.0341
233	3078	3152	73.83	75.60	54.35	45.65	64.09	0.2789
234	2753	1303	99.54	47.11	2.67	97.33	51.11	0.0868

Table 6.5. Performance Results for Zeelenberg QRS detection algorithm on full MIT-BIH Arrhythmia Database records, 30 minutes long each. Detection performed on modified limb lead II.

Record	Det. B.	Ann. B.	Se	PP	Sp	FPR	Q_α	MCC
100	36	36	100.00	100.00	100.00	0.00	100.00	1.0000
101	33	33	100.00	100.00	100.00	0.00	100.00	1.0000
102	35	35	31.43	31.43	86.13	13.87	58.78	0.1756
103	33	33	100.00	100.00	100.00	0.00	100.00	0.0000
104	34	38	21.05	23.53	78.86	21.14	49.96	−0.0009
105	39	39	97.44	97.44	80.00	20.00	88.72	0.7744
106	35	33	90.91	85.71	93.51	6.49	92.21	0.8305
107	33	34	2.94	3.03	64.84	35.17	33.89	−0.3253
108	52	31	0.00	0.00	89.47	10.53	44.74	−0.0831
109	46	45	8.89	8.70	80.65	19.36	44.77	−0.1038
111	36	33	3.03	2.78	91.05	8.95	47.04	−0.0569
112	78	41	34.15	17.95	88.24	11.77	61.19	0.1681
113	31	28	82.14	74.19	98.36	1.64	90.25	0.7675
114	26	26	3.85	3.85	94.12	5.88	48.98	−0.0204
115	30	30	0.00	0.00	94.21	5.79	47.10	−0.0579
116	37	37	16.22	16.22	94.64	5.36	55.43	0.1085
117	31	24	4.17	3.23	91.45	8.55	47.81	−0.0389
118	44	35	0.00	0.00	81.67	18.33	40.83	−0.1667
119	38	32	87.50	73.68	79.17	20.83	83.33	0.6540
121	29	29	96.55	96.55	75.00	25.00	85.78	0.7155
122	42	42	100.00	100.00	100.00	0.00	100.00	1.0000
123	27	23	0.00	0.00	94.83	5.17	47.41	−0.0479
124	24	24	95.83	95.83	75.00	25.00	85.42	0.7083
200	44	43	23.26	22.73	93.35	6.65	58.30	0.1643
201	41	1	0.00	0.00	0.00	100.00	0.00	−1.0000
202	29	25	96.00	82.76	54.55	45.46	75.27	0.5883
203	43	1	0.00	0.00	42.67	57.33	21.33	−0.1318
205	42	42	100.00	100.00	100.00	0.00	100.00	1.0000
207	41	28	42.86	29.27	91.50	8.50	67.18	0.2895
208	64	50	92.00	71.88	64.71	35.29	78.35	0.5885
209	44	44	0.00	0.00	91.32	8.68	45.66	−0.0868
210	43	1	0.00	0.00	6.52	93.48	3.26	−0.4834
212	43	43	2.33	2.33	80.91	19.09	41.62	−0.1677
213	52	52	13.46	13.46	90.41	9.59	51.93	0.0387
214	41	36	91.67	80.49	61.91	38.10	76.79	0.5751
215	53	53	1.89	1.89	87.68	12.32	44.78	−0.1044
217	34	34	0.00	0.00	77.33	22.67	38.67	−0.2267
219	45	1	0.00	0.00	49.44	50.56	24.72	−0.1060
220	34	35	2.86	2.94	94.00	6.00	48.43	−0.0319
221	39	1	0.00	0.00	2.50	97.50	1.25	−0.6982
222	35	36	97.22	100.00	100.00	0.00	98.61	0.6972
223	38	38	92.11	92.11	57.14	42.86	74.62	0.4925
228	44	39	84.62	75.00	59.26	40.74	71.94	0.4576
230	39	8	0.00	0.00	91.18	8.82	45.59	−0.0414
232	49	1	0.00	0.00	46.74	53.26	23.37	−0.1100
233	50	50	76.00	76.00	94.89	5.11	85.45	0.7089
234	43	43	97.67	97.67	66.67	33.33	82.17	0.6434

Table 6.6. Performance Results for Zeelenberg QRS detection algorithm on shortened MIT-BIH Arrhythmia Database records, approximately 28 seconds long each. Detection performed on modified limb lead II.

Record	Det. B.	Ann. B.	Se	PP	Sp	FPR	Q_α	MCC
100	35	36	94.44	97.14	66.67	33.33	80.56	0.5368
101	32	33	96.97	100.00	100.00	0.00	98.49	0.6963
102	22	35	34.29	54.55	97.72	2.28	66.00	0.3979
103	32	33	96.97	100.00	100.00	0.00	98.49	0.0000
104	20	38	18.42	35.00	96.23	3.77	57.33	0.1969
105	37	39	94.87	100.00	100.00	0.00	97.44	0.5624
106	31	33	93.94	100.00	100.00	0.00	96.97	0.8548
107	33	34	20.59	21.21	91.13	8.87	55.86	0.1187
108	26	31	74.19	88.46	75.00	25.00	74.60	0.4513
109	29	45	64.44	100.00	100.00	0.00	82.22	0.1947
111	25	33	72.73	96.00	66.67	33.33	69.70	0.2364
112	39	41	95.12	100.00	100.00	0.00	97.56	0.5631
113	27	28	96.43	100.00	100.00	0.00	98.21	0.8018
114	22	26	76.92	90.91	50.00	50.00	63.46	0.2070
115	29	30	96.67	100.00	100.00	0.00	98.33	0.6952
116	37	37	97.30	97.30	66.67	33.33	81.98	0.6396
117	23	24	91.67	95.65	66.67	33.33	79.17	0.5161
118	30	35	65.71	76.67	98.39	1.61	82.05	0.6885
119	30	32	90.63	96.67	80.00	20.00	85.31	0.6165
121	28	29	96.55	100.00	100.00	0.00	98.28	0.6948
122	39	42	90.48	97.44	66.67	33.33	78.57	0.4193
123	22	23	95.65	100.00	100.00	0.00	97.83	0.6916
124	24	24	91.67	91.67	60.00	40.00	75.83	0.5167
200	35	43	79.07	97.14	97.14	2.86	88.11	0.7621
201	37	1	0.00	0.00	22.92	77.08	11.46	−0.2535
202	22	25	84.00	95.46	50.00	50.00	67.00	0.2292
203	37	1	0.00	0.00	50.67	49.33	25.33	−0.1125
205	41	42	97.62	100.00	100.00	0.00	98.81	0.6986
207	24	28	82.14	95.83	91.67	8.33	86.91	0.6904
208	41	50	82.00	100.00	100.00	0.00	91.00	0.0000
209	40	44	84.09	92.50	62.50	37.50	73.30	0.3990
210	40	1	100.00	2.50	7.14	92.86	53.57	0.0423
212	32	43	74.42	100.00	100.00	0.00	87.21	0.0000
213	47	52	90.39	100.00	100.00	0.00	95.19	0.3881
214	32	36	86.11	96.88	85.71	14.29	85.91	0.6077
215	48	53	88.68	97.92	50.00	50.00	69.34	0.2173
217	9	34	0.00	0.00	97.51	2.49	48.75	−0.0469
219	32	1	0.00	0.00	21.95	78.05	10.98	−0.2794
220	34	35	97.14	100.00	100.00	0.00	98.57	0.8048
221	33	1	0.00	0.00	5.71	94.29	2.86	−0.5606
222	35	36	97.22	100.00	100.00	0.00	98.61	0.6972
223	36	38	94.74	100.00	100.00	0.00	97.37	0.5620
228	23	39	58.97	100.00	100.00	0.00	79.49	0.4011
230	39	8	75.00	15.39	8.33	91.67	41.67	−0.2026
232	12	1	0.00	0.00	72.73	27.27	36.36	−0.0909
233	46	50	88.00	95.65	66.67	33.33	77.33	0.4415
234	41	43	95.35	100.00	100.00	0.00	97.67	0.5638

Table 6.7. Performance Results for Tompkins QRS detection algorithm on shortened MIT-BIH Arrhythmia Database records, approximately 28 seconds long each. Detection performed on modified limb lead II.

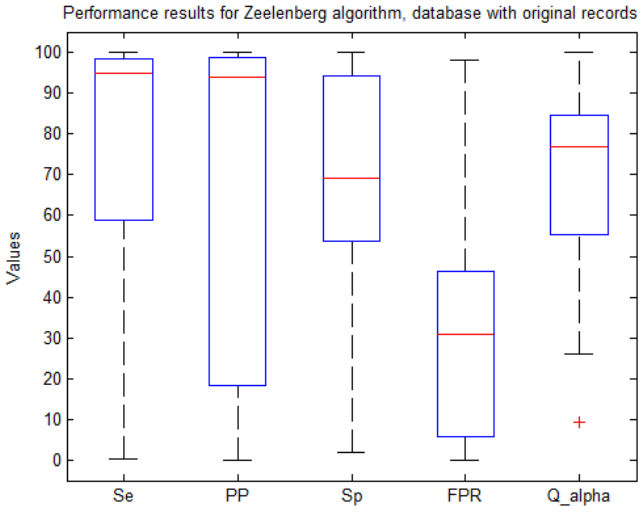


Figure 6.14. Boxplot of the performance measures for Zeelenberg algorithm on full MIT-BIH Arrhythmia Database records

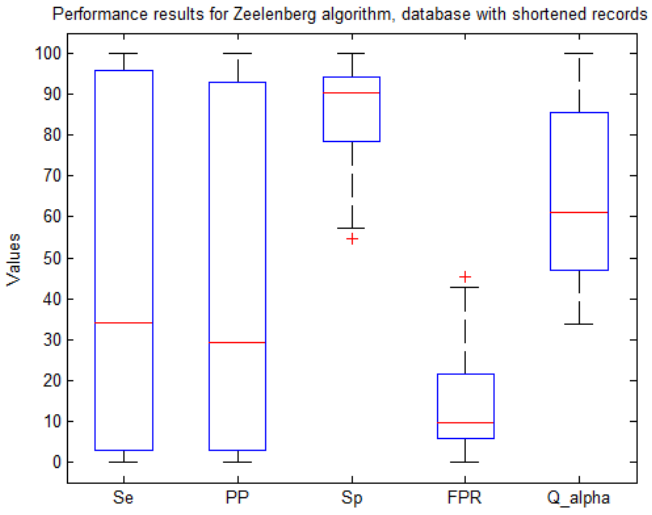


Figure 6.15. Boxplot of the performance measures for Zeelenberg algorithm on shortened MIT-BIH Arrhythmia Database records

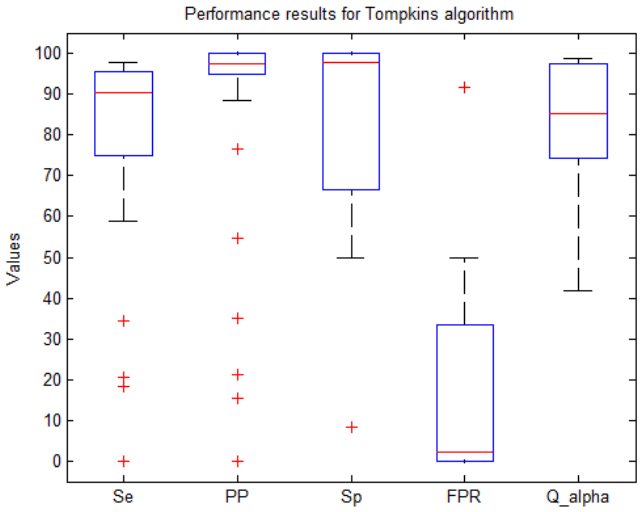


Figure 6.16. Boxplot of the performance measures for Tompkins algorithm on shortened MIT-BIH Arrhythmia Database records

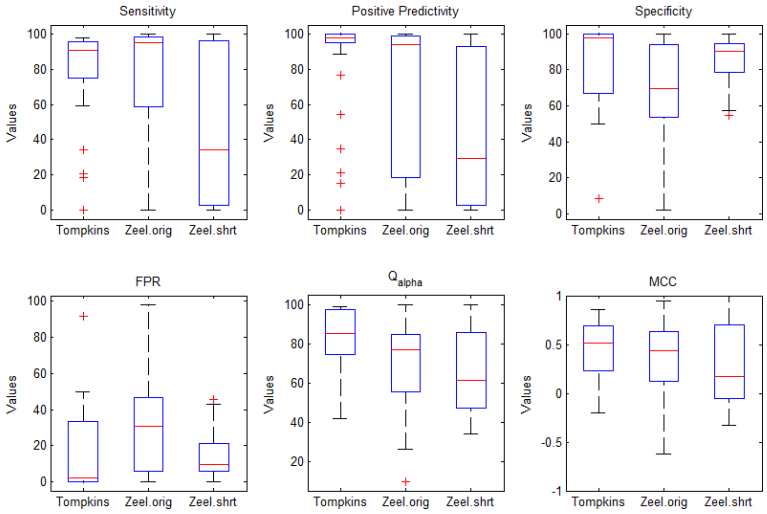


Figure 6.17. Boxplots comparison of the different parameters for each detection algorithm

Record	MIT Annotations		Classifier I		Classifier II	
	Beat Type	NB	Beat Type	NB	Beat Type	NB
101	APC	3	RC(N	1	RC(N	1
	ARFCT	4	NORMAL	1863	NORMAL	1864
	NOISE	4	UNKNOWN	2	NOISE	3
	NORMAL	1860	ARFCT	1	ARFCT	1
	RC(N	1	NOISE	1	ARFCT	3
	UNKNOWN	2	ARFCT	1	UNKNOWN	2
			NORMAL	1		
			ARFCT	1		
			ARFCT	1		
			NOISE	2		
Total num. of beats sorted		1874		1874		1874
109	FUSION	2	RC(N	1	RC(N	1
	LBBB	2492	LBBB	1443	LBBB	2393
	NOISE	2	PVC	3	LBBB	124
	PVC	38	PVC	25	NOISE	2
	RC(N	1	LBBB	63	PVC	15
			LBBB	973		
			PVC	1		
			NOISE	1		
			PVC	14		
			NOISE	1		
			LBBB	2		
			PVC	1		
			LBBB	7		
Total num. of beats sorted		2535		2535		2535
230	NORMAL	6	RC(N	1	RC(PRE	2
	RC(N	1	NORMAL	6	NORMAL	6
	RC(PRE	1	RC(PRE	1		
Total num. of beats sorted		8		8		8

Table 6.8. Performance results for the beats classification algorithms: number of beats. Column NB stands for the number of beats of the correspondant type. Beat types from annotations are displayed in alphabetical order. Beat types obtained by the two algorithms are displayed in the order in which correspondant templates are validated. The test signals are the first 30 min of each specified record of the MIT-BIH Arrhythmia Database.

Table 6.8 shows a small sample of the performance of the two methods in terms of the number of beats sorted into each group for three arbitrary records, chosen so that the comparison is more evident. We can see how the Classifier I is not too selective in the sorting, accepting several different templates of a common type. This can be an inconvenient as beats are scattered into different groups when they could be classified more compactly, or opposedly, seen as an advantage, in the sense that the algorithm allows for a finer sorting distinguishing mathematical (hence, irrefutable) dissimilarities by means of comparison to continuously updated templates. On the other hand, Classifier II is shown to provide more robust and compact classification, sorting the beats in less different groups, often assimilating several of the same-kind templates into a common group. Other remark to be noticed is that some concrete types of beats tend to be missclassified, such as APC and FUSION beats in the examples displayed in Table 6.8, due to ambiguity in waveform and incapability of the methods of distinguishing these isolated beats by threshold correlation comparison.

The reduction in the number of templates that Classifier II accepts as validated in comparison to Classifier I, can also be seen in Table 6.9, where we also see the number of redundant template types that the algorithms find. However, we can see that this reduction is in some cases excessive, for the modified method (Classifier II) merges groups of beats that the annotations of the records state to be distinct.

Figures 6.18 and 6.19 show a template morphological analysis for arbitrarily chosen records 102, 103, 109 and 222 of MIT-BIH Arrhythmia Database (more are not included for space reasons). These plots are performed by Classifier I, and the results can be contrasted with those of Tables 6.8 and 6.9, seeing the amount of similar beats clustered in the same group and averaged for each template and the dispersion in between them.

Table 6.9. Performance Results for the beats classification algorithms: number of templates. Columns labelled NT denote the number of templates incuded in the MIT annotations files, or encountered by the classification algorithms. Column Templates lists the different templates encountered, with an accompanying {x} denoting that x repetitions of the template are stablished by the algorithms. If no number is specified, only one instance of the template with that name appears. All templates are displayed in alphabetical order. The test signals are the first 30 min of each specified record of the MIT-BIH Arrhythmia Database.

Record	MIT Annotations		Classifier I		Classifier II	
	NT	Templates	NT	Templates	NT	Templates
100	4	APC NORMAL PVC RC(N	5	NORMAL {3} PVC RC(N	3	NORMAL PVC RC(N

Continued on next page...

Table 6.9. *... continued from previous page*

Record	MIT Annotations		Classifier I		Classifier II	
	NT	Templates	NT	Templates	NT	Templates
101	6	APC ARFCT NOISE NORMAL RC(N UNKNOWN	10	ARFCT {4} NOISE {2} NORMAL {2} RC(N UNKNOWN	6	ARFCT {2} NOISE NORMAL RC(N UNKNOWN
102	6	NORMAL PACE PFUS PVC RC(N RC(P	24	NORMAL {2} PACE {10} PFUS {4} PVC {3} RC(N {2} RC(P {3}	21	PACE {15} PFUS PVC RC(N {2} RC(P {2}
103	4	APC NOISE NORMAL RC(N	6	NOISE {4} NORMAL RC(N	3	NOISE {2} NORMAL
104	8	NOISE NORMAL PACE PFUS PVC RC(N RC(P UNKNOWN	67	NOISE {21} PACE {18} PFUS {12} RC(N {10} RC(P {6}	19	NOISE {6} PACE {5} PFUS {3} RC(N {2} RC(P {3}
200	8	APC FUSION NOISE NORMAL PVC RC(B RC(N RC(VT	124	NOISE {26} NORMAL {5} PVC {10} RC(B {37} RC(N {44} RC(VT {2}	27	NOISE {4} NORMAL {2} PVC RC(B {11} RC(N {8} RC(VT

Continued on next page...

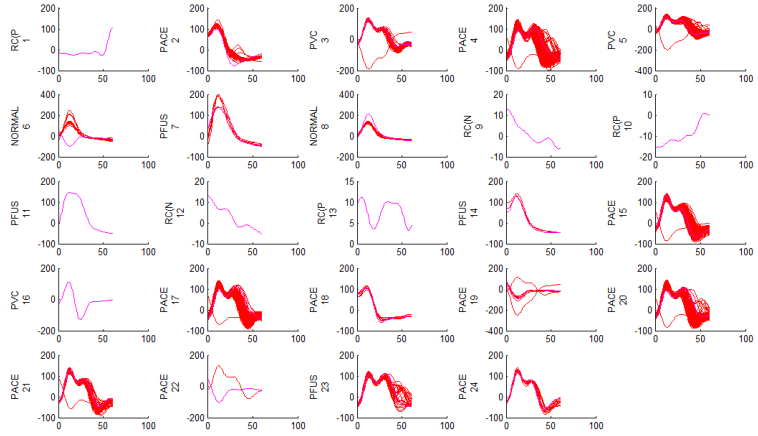
Table 6.9. *... continued from previous page*

Record	MIT Annotations		Classifier I		Classifier II	
	NT	Templates	NT	Templates	NT	Templates
205	8	APC ARTFCT FUSION NOISE NORMAL PVC RC(N RC(VT	24	ARFCT NOISE NORMAL {3} PVC {6} RC(N {7} RC(VT {6}	8	FUSION NORMAL PVC RC(N {4} RC(VT
212	5	ARFCT NOISE NORMAL RBBB RC(N	14	ARFCT NOISE {11} NORMAL RBBB	5	NOISE {4} NORMAL
222	5	APC NOISE NORMAL RC(AFI RC(N	13	APC {2} NOISE {6} NORMAL {4} RC(AFI	6	NOISE {4} NORMAL RC(AFI
230	3	NORMAL RC(N RC(PRE	3	NORMAL RC(N RC(PRE	2	NORMAL RC(PRE

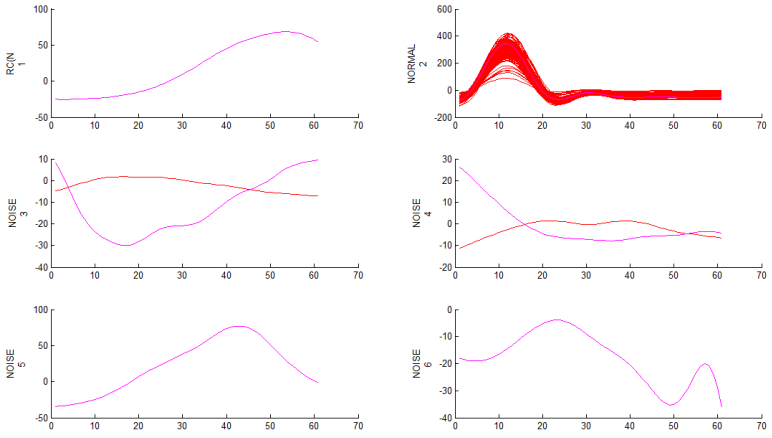
6.5 HRV measures

The function `HRV.m` introduced in the end of Section 4.6 that we have implemented for the HRV analysis receives as its basic input an array containing the collection of samples on which R peaks are found in an ECG signal; this can be calculated with the functions `zeelenberg.m` or `tompkins.m` reported in Section 6.3, or by any other method of QRS complexes detection.

Its operation is controlled by four threshold values, namely: *HRcritT* and *HRcritB*, which demarcate the limits of normal heart rates and discriminate tachycardias as bradycardias as HR above and below the respective values of the critical thresholds, and *threshup* and *threshdown*, which influence the method of discarding abnormal R-R periods. By default, *HRcritT* and *HRcritB* are set to 100 bpm and 50 bpm respectively, whereas *threshup* and *threshdown* have the values of 1.2 and 0.8, chosen for best performance.

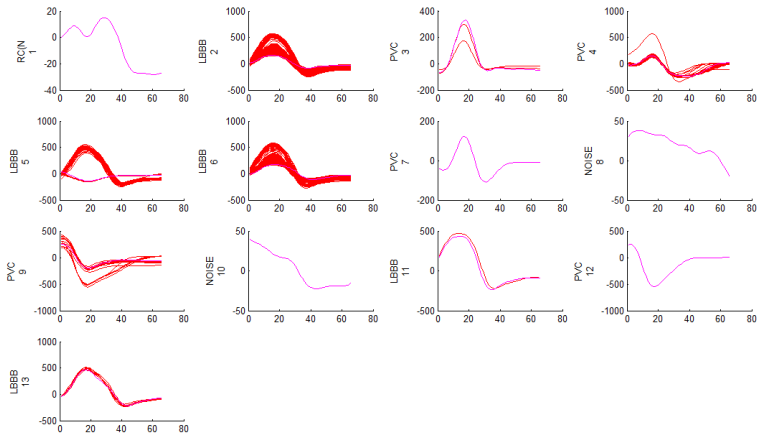


(a) Record 102

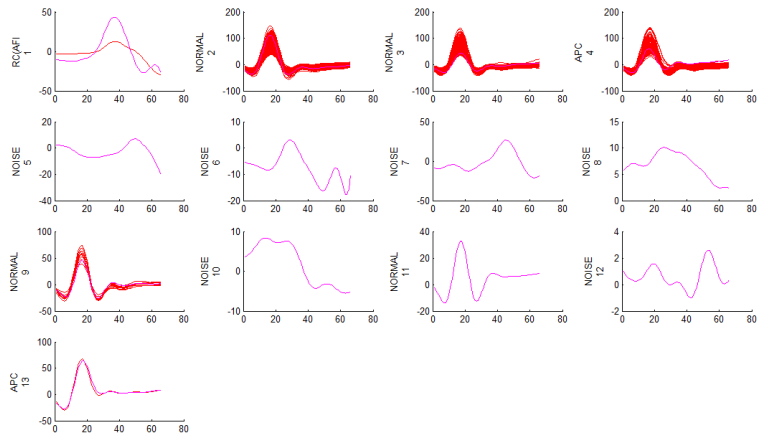


(b) Record 103

Figure 6.18. Beat templates morphological analysis for records 102 and 103 of MIT-BIH Arrhythmia Database. For each record, all templates found by Classifier I are plotted along with their respective mean.



(a) Record 109



(b) Record 222

Figure 6.19. Beat templates morphological analysis for records 109 and 222 of MIT-BIH Arrhythmia Database. For each record, all templates found by Classifier I are plotted along with their respective mean.

Record	HRm	SDNN	μSDANN	σSDANN	RMSSD
101	62.6409	75.6443	48.3558	14.2180	54.9346
109	84.4245	38.7593	45.4587	22.7278	45.5517
115	65.7824	93.1146	81.2505	36.4829	82.0522
200	93.9968	151.7396	143.1710	30.5453	248.3663
212	91.9645	43.8795	41.8293	4.5641	30.8130
217	73.6997	72.7091	99.3333	33.0676	89.8810

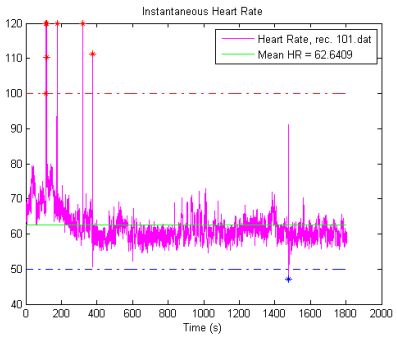
Table 6.10. Time-domain HRV measures of MIT-BIH Arrhythmia Database records. Column μSDANN displays the arithmetic mean of SDANN values over the intervals, and σSDANN its standard deviation. All records are 30 minutes long.

The function calculates the instantaneous heart rate of a beat-to-beat series removing abnormal intervals that would yield HRs outside the margin comprised between $threshdown \cdot HRcritB$ and $threshup \cdot HRcritT$, thus considering only what was referred as N-N (normal - normal) intervals in Section 4.6, and marking as bradycardias or tachycardias the HR samples outside the margin $[HRcritB, HRcritT]$. It plots, too, if required, the instantaneous HR, the mean HR and the limits mentioned above, and calculates the HRV time-domain measures SDNN, SDANN, SDNN index and RMSSD[3].

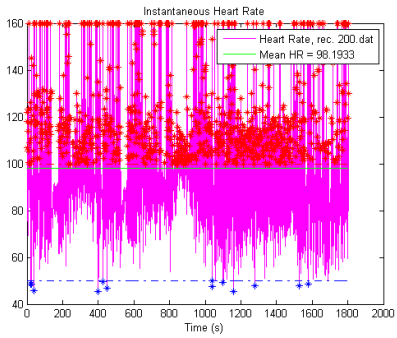
These measures are shown in Table 6.10 for some signals included in the MIT-BIH Arrhythmia Database. The SDNN index is not provided in the Table since it is calculated over a 24 hours period, and the records analyzed here are 30 minutes long. Thus, the mean SDANN over 30 minutes substitutes it; the standard deviation is also displayed. Figure 6.20 shows the plots performed by `HRV.m` of some records. Note the little dispersion in the HR of record 101, and opposedly, the great variability found on record 200. Remember from Tables 6.8 and 6.9 the amount of normal beats in record 101 and the several ectopic beats templates in record 200. Record 115 also shows a wide variation in HR, as can be seen from the tachycardic and bradycardic points in Figure 6.20 and its value of SDNN, SDANN and RMSSD in Table 6.10. Record 217 seems to be shortened because after approximately the 3 minutes, the beats are not classified as normal, and hence no position for R onsets is provided. Of course, the HR could be calculated nevertheless, and time-domain measures computed, skipping the restriction of N-N beats, and considering any QRS complex, or similar waveform, for the reference of the R-R intervals.

Figures 6.21 to 6.26 show the results of the HRV analysis performed by Kubios HRV software on some of the records of MIT-BIH Arrhythmia Database. First, the R-R intervals time series is plotted, and several time-domain measures are displayed. Probability distributions of both the R-R intervals and the HR are plotted as histograms, and spectral analysis is carried out by estimating the spectrum of the R-R intervals series by two methods: a FFT calculated by means of Welch's periodogram, and an autoregressive (AR) model of order 16.

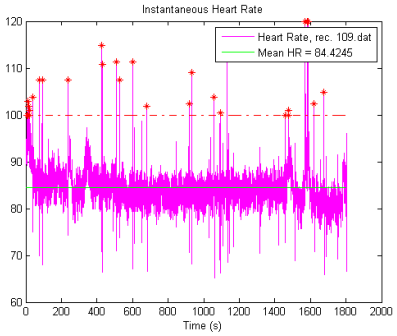
[3]Turn to Section 4.6 for reference.



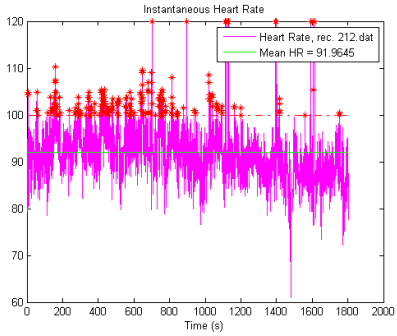
(a) 101.dat



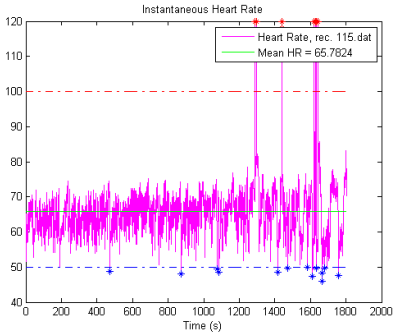
(b) 200.dat



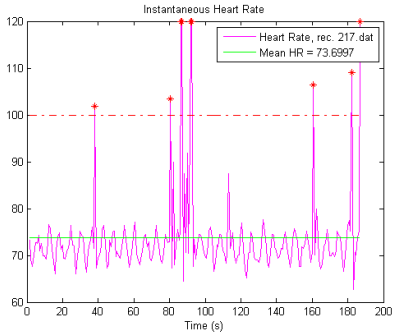
(c) 109.dat



(d) 212.dat



(e) 115.dat



(f) 217.dat

Figure 6.20. HRV plots of MIT-BIH Arrhythmia Database records. In every case, *HRcritT* and *HRcritB* are set to 100 bpm and 50 bpm respectively, and *threshup* and *threshdown* are set to 1.2 and 0.8 except in the case of Record 200, where *threshup* was set to 1.6. All records are 30 minutes long.

The spectral HRV measures are provided according to the terms exposed in 4.6, but assimilating the ULF inside the VLF band. This software also performs several nonlinear methods for evaluating the HRV, such as the Poincaré plot with elliptic fitting (STD1 and STD2 axis measures are provided), recurrence plot calculations, detrended fluctuation analysis (DFA) calculating the slopes of the best fit, and some entropy measures (ShanEn, ApEn and SampEn) for evaluating the complexity of the HRV.

We can see in Figures 6.21 and 6.22 the fan-like shape of the Poincaré plot that was commented in Section 4.6, though with much sample scatter, and opposedly, the round shape attained in Figure 6.25. Notice also in this figure the particular frequency distribution in comparison to the ones of the other records.

6.6 DPCM simulations

In this section we present the performance results of our DPCM compression system. We have implemented an M-function called `DPCMexp.m` which carries out the DPCM experiments, plots the results and calculates several distorsion measures, namely PRD, SNR(dB), and the rms error and maximum error throughout the record. The function receives as input arguments two parameters that control its operation: the number of bits used by the quantization system, and the order of the linear predictor used.

The tasks of coding-compressing and decoding-decompressing are done separately by two other functions, `dpcm_encoder.m` and `dpcm_decoder.m` which in turn call to the routines `quantizer_encoder.m` and `quantizer_decoder.m`. The code of all these functions is enclosed in Appendix D. This modular code design has been chosen to give the possibility of future expansions, like implementing different types of quantizers to be used by the same DPCM scheme, or a different prediction system based on filtering routines other than the linear predictor. The calculation of the linear predictor coefficients (usually a predictor of order 2 is employed) is carried out employing the MATLAB function `lpc.m` included in the software distribution, which uses the Levinson-Durbin recursion to solve the equations that arise from the least-squares formulation explained in Section 5.5.

The mid-rise uniform quantizer system employed uses a manual mask selection for coding with 3 bits and below, and for 4 bits and above, a decimal to binary and binary to decimal conversion routines included in App. D, Section D.6.

Table 6.11 shows the distorsion measures calculated for a coding with 10 bits and a linear predictor of order 2 on some records of the MIT-BIH Arrhythmia Database, which are shown in Figures 6.27 to 6.32, comparing the original and reconstructed signal and with an accompanying plot of the error between both.

We can see from the figures that the reconstructed signals resemble fairly enough the original signals from the records, but as we explained in Section 5.2

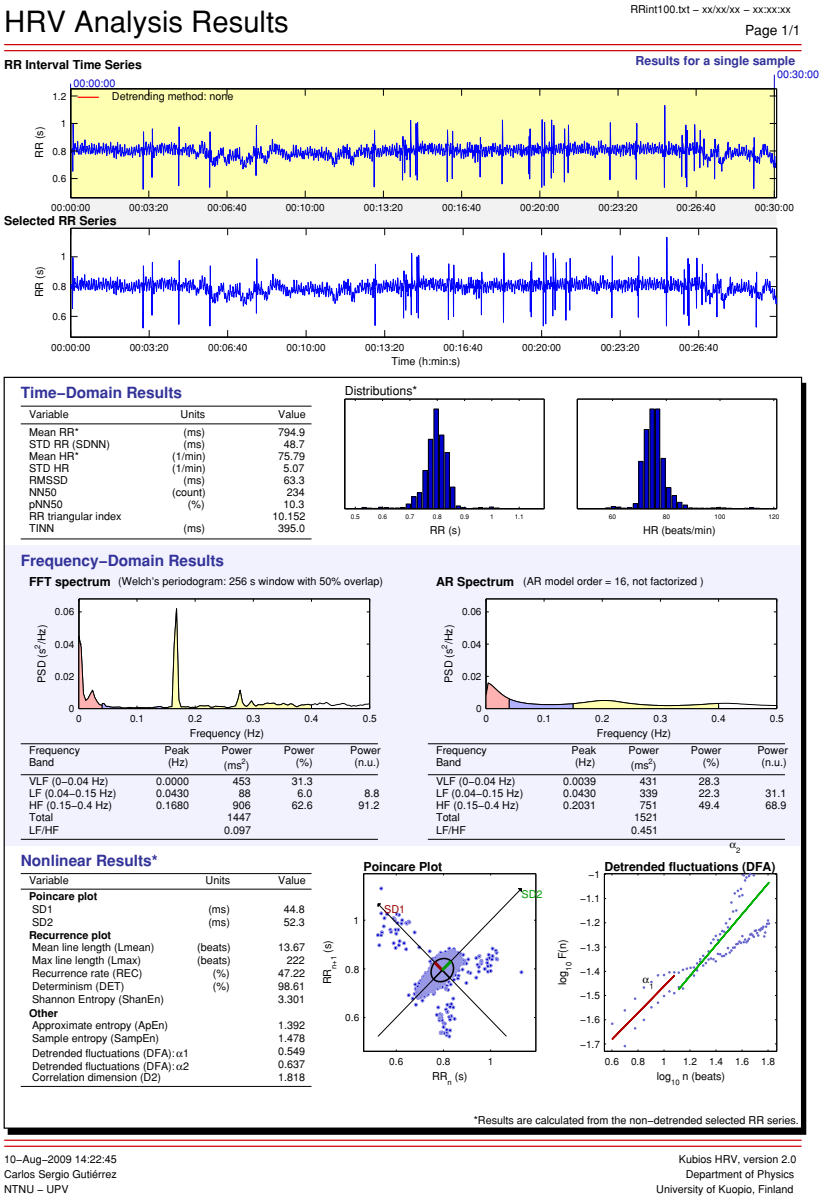


Figure 6.21. HRV Analysis for Record 100 of MIT-BIH Arrhythmia Database. Performed by Kubios HRV.

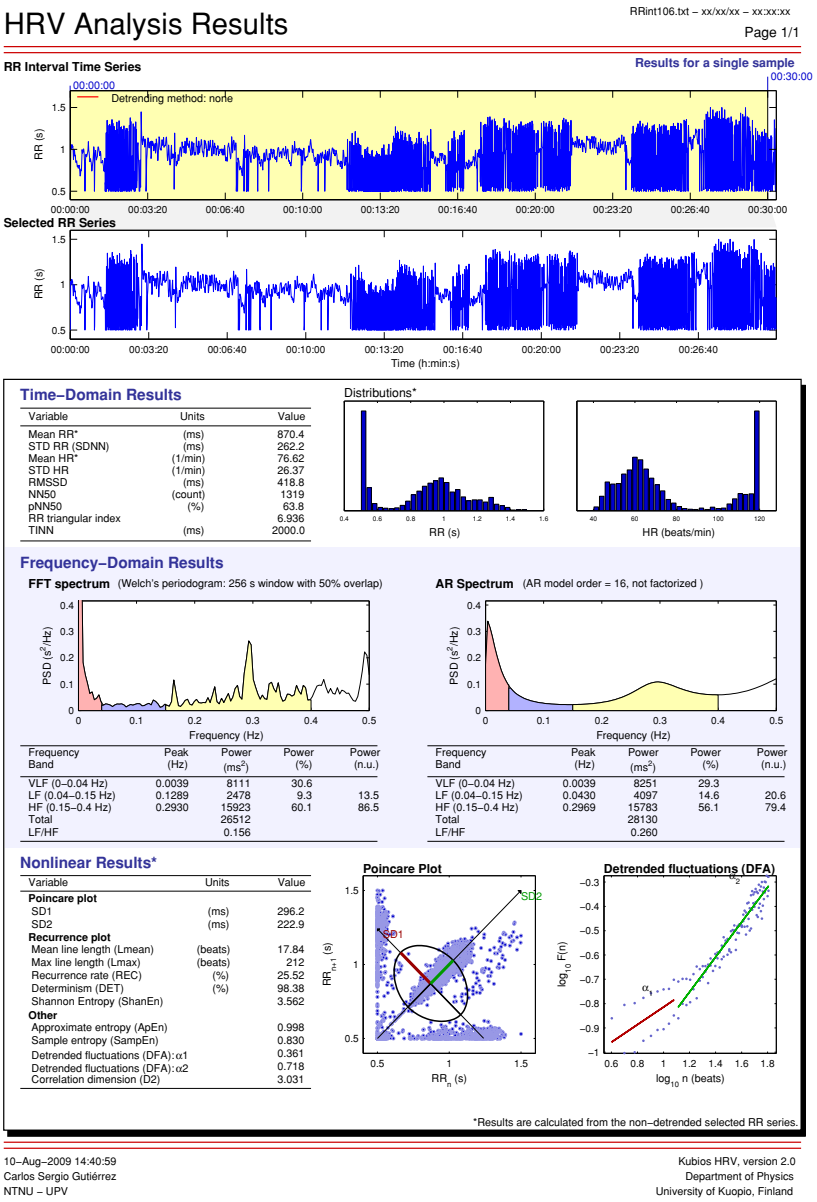


Figure 6.22. HRV Analysis for Record 106 of MIT-BIH Arrhythmia Database. Performed by Kubios HRV.

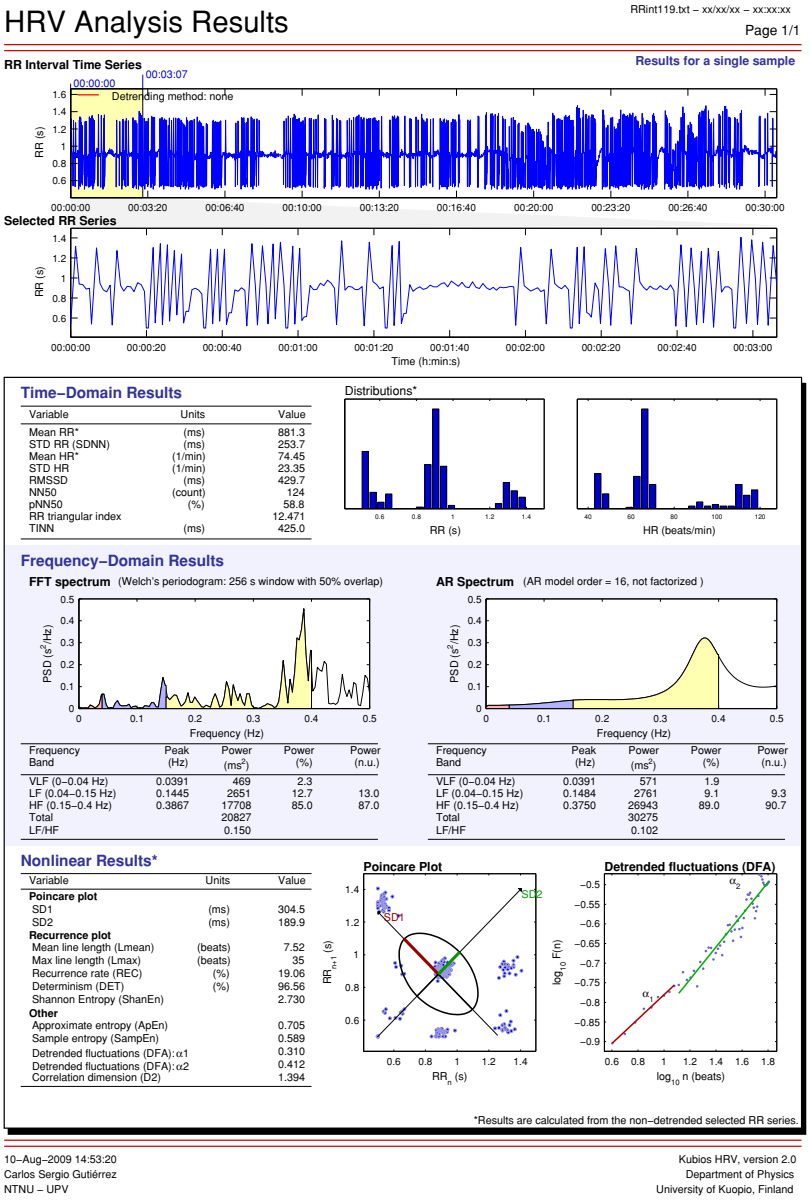


Figure 6.23. HRV Analysis for Record 119 of MIT-BIH Arrhythmia Database. Performed by Kubios HRV.

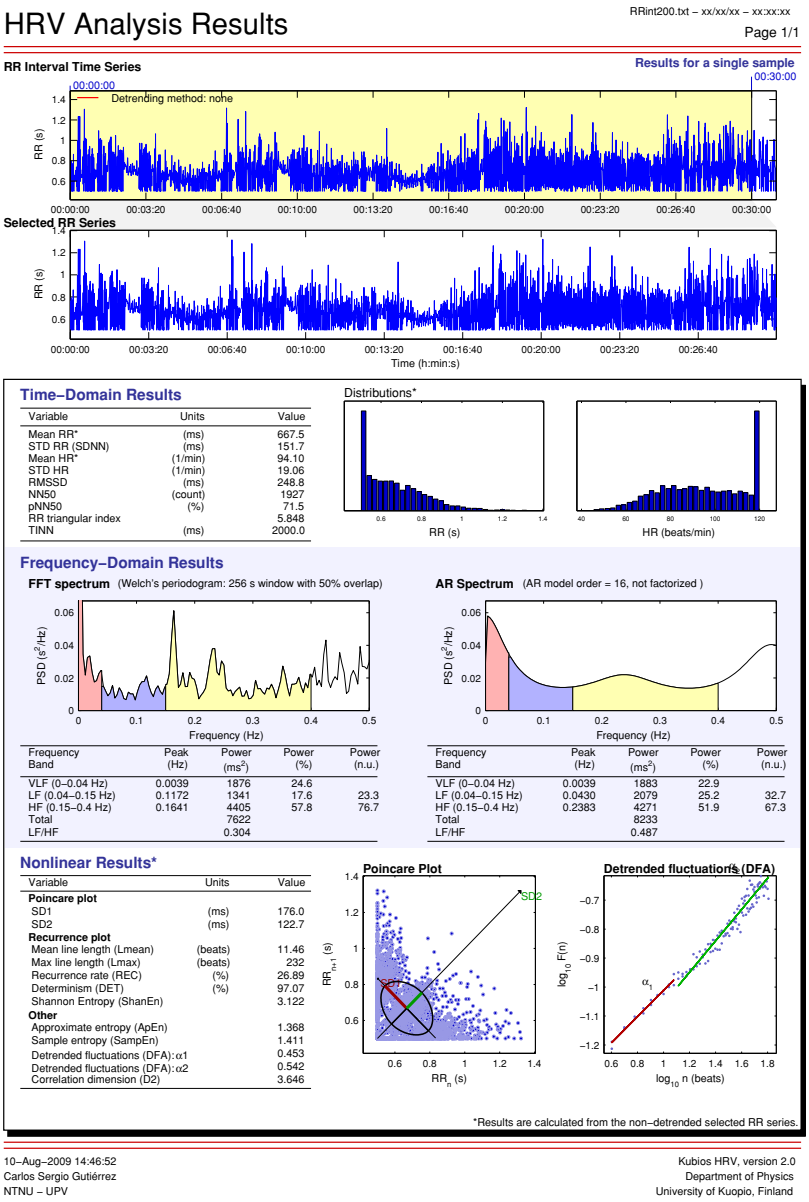


Figure 6.24. HRV Analysis for Record 200 of MIT-BIH Arrhythmia Database. Performed by Kubios HRV.

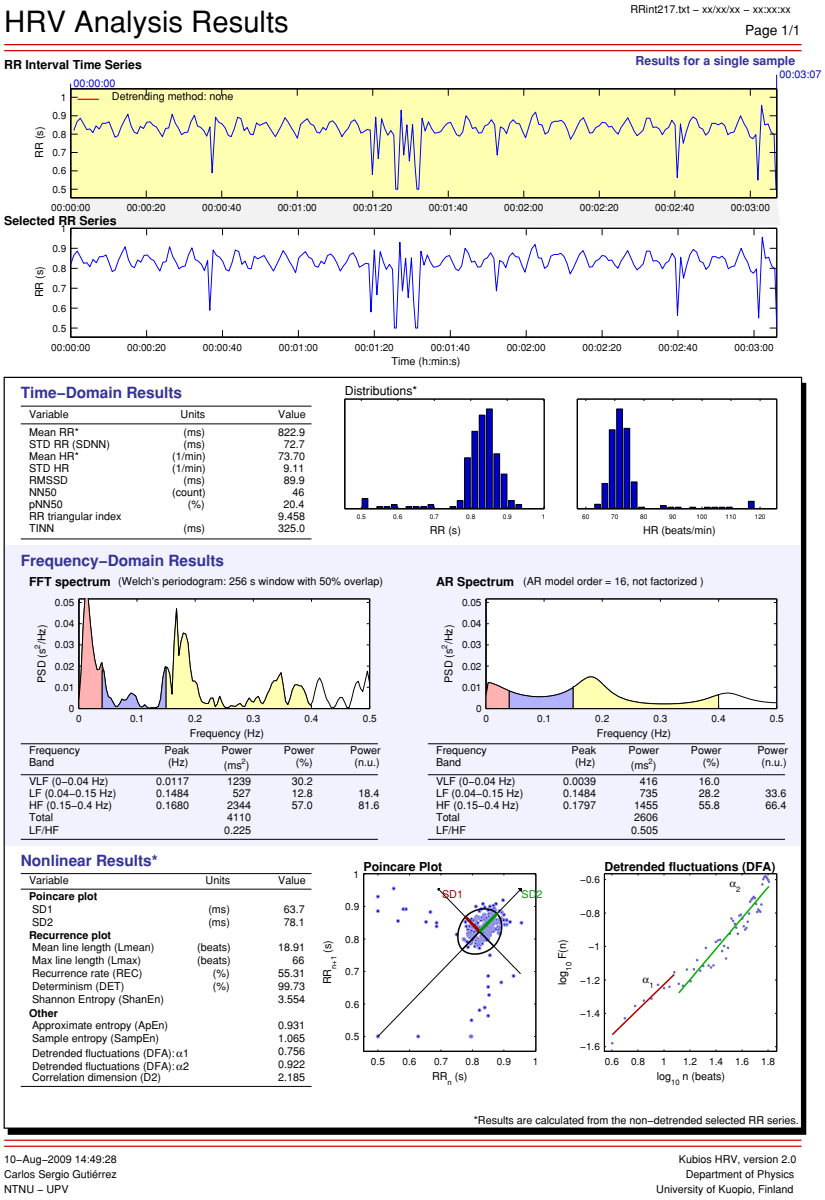


Figure 6.25. HRV Analysis for Record 217 of MIT-BIH Arrhythmia Database. Performed by Kubios HRV.

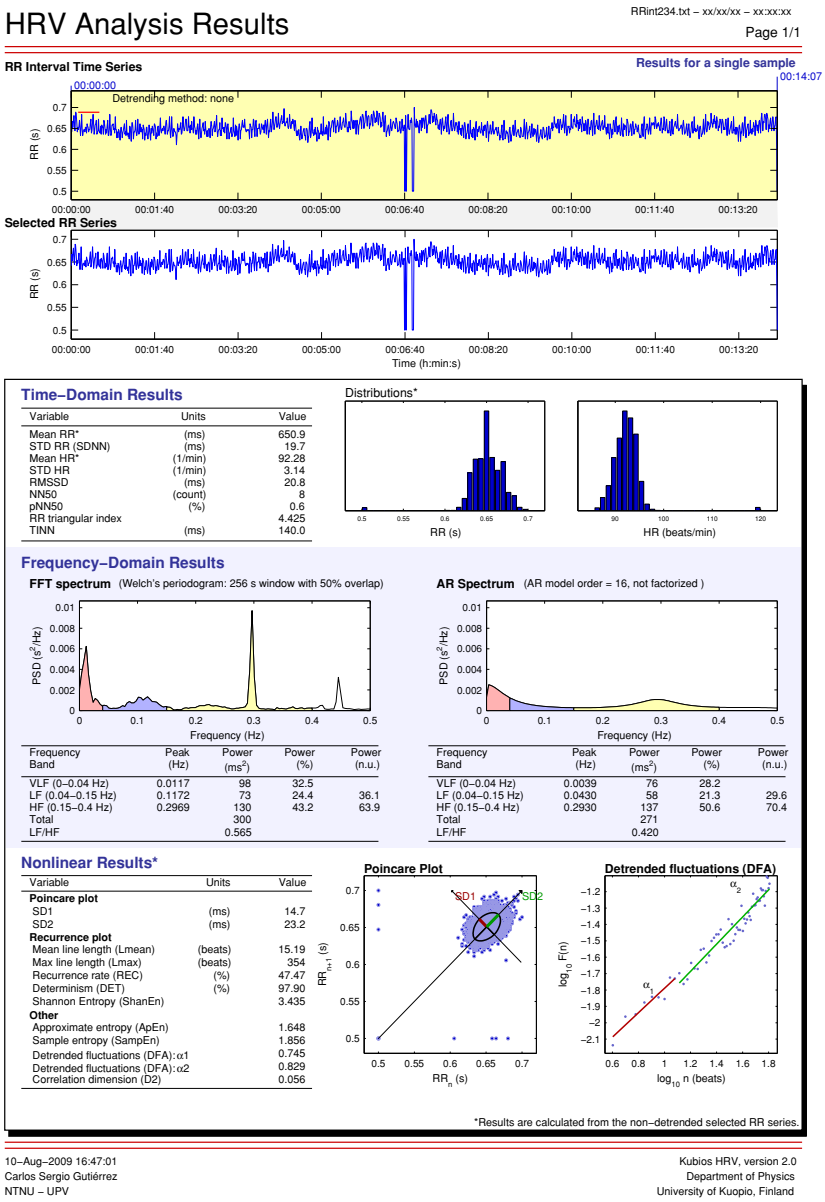


Figure 6.26. HRV Analysis for Record 234 of MIT-BIH Arrhythmia Database. Performed by Kubios HRV.

Record	PRD (%)	SNR (dB)	rms (x1000)	MAX (x1000)
100	7.9962	50.5241	1.1194	1.9530
105	4.8470	60.5363	1.1342	1.9531
111	7.0898	52.9302	1.1368	1.9529
115	5.3542	58.5457	1.1502	1.9531
203	3.3552	67.8932	1.1391	1.9527
228	7.3434	52.2275	1.1517	1.9531

Table 6.11. Distorsion measures of DPCM compression on different signals from MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization, 10 bits. CR = 1.1

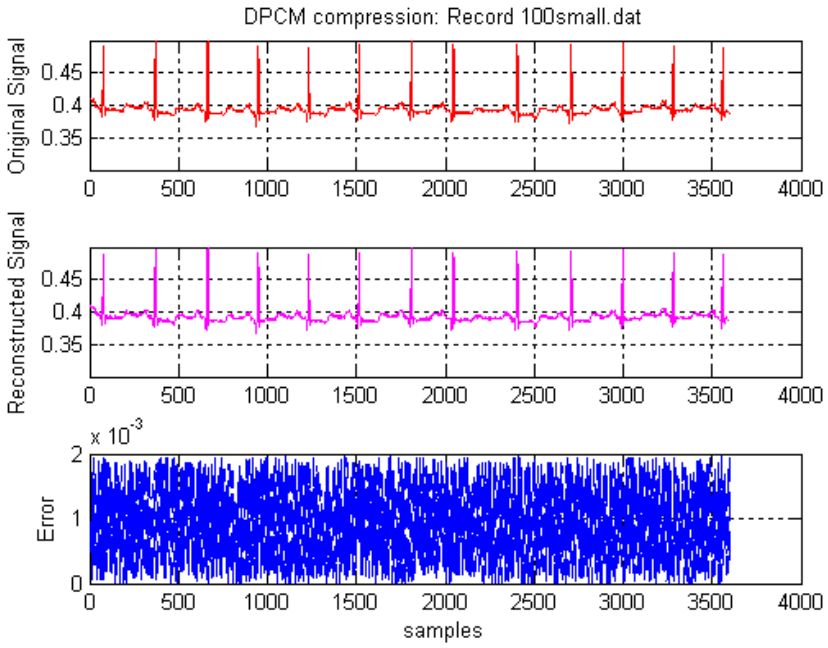


Figure 6.27. DPCM compression of Record 100, MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization, 10 bits.

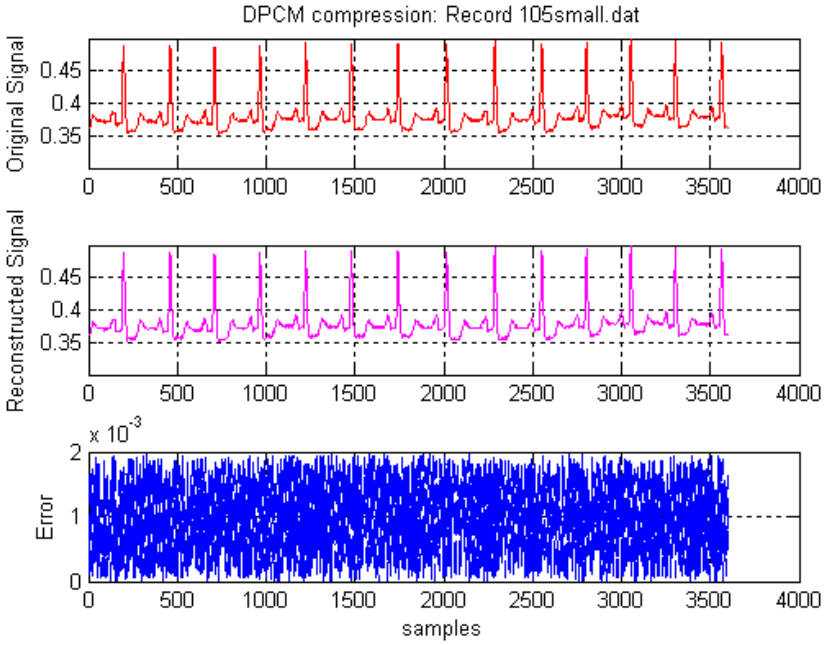


Figure 6.28. DPCM compression of Record 105, MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization, 10 bits.

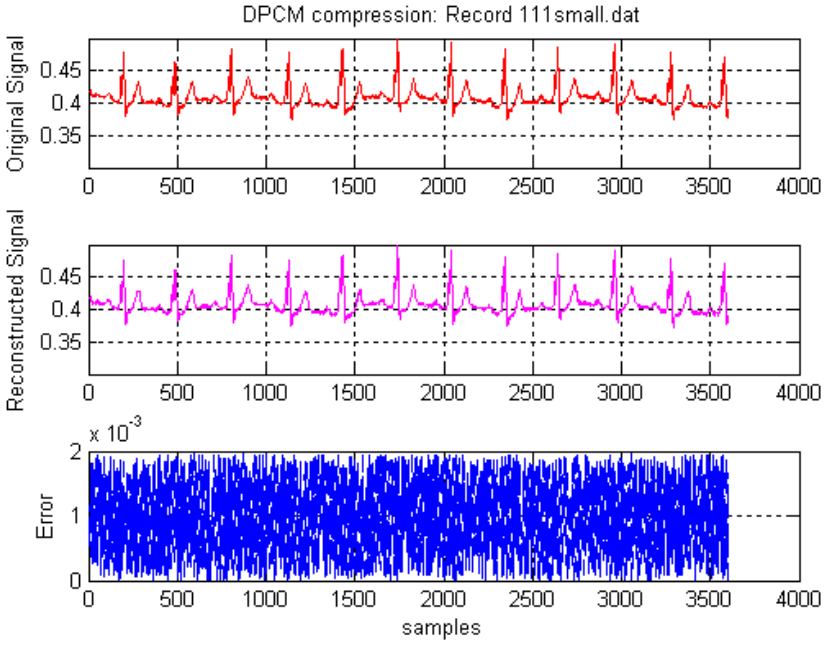


Figure 6.29. DPCM compression of Record 111, MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization, 10 bits.

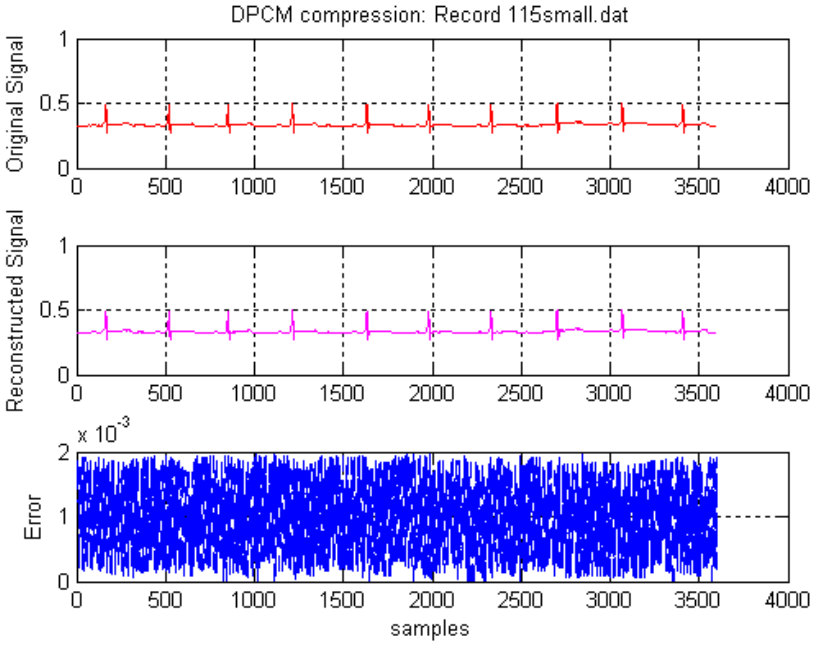


Figure 6.30. DPCM compression of Record 115, MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization, 10 bits.

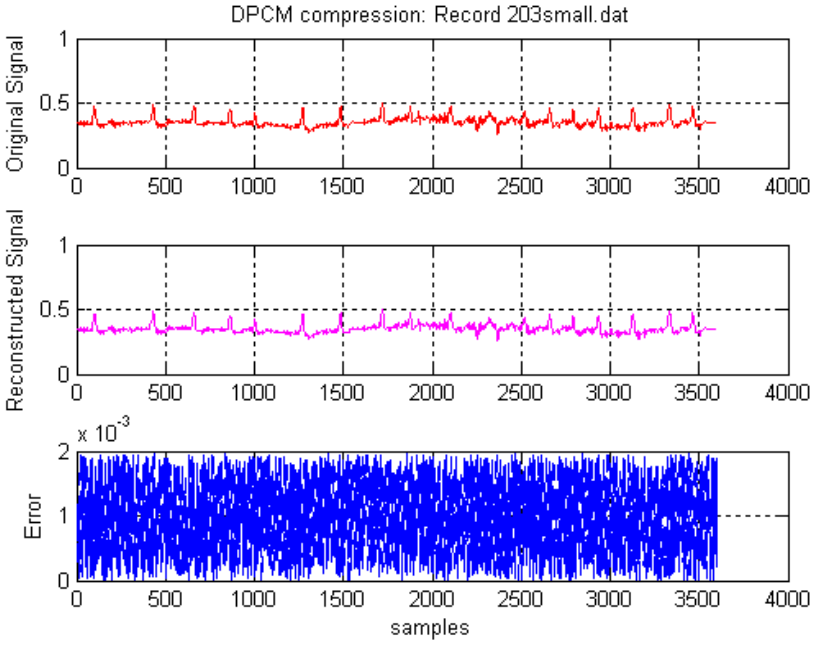


Figure 6.31. DPCM compression of Record 203, MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization, 10 bits.

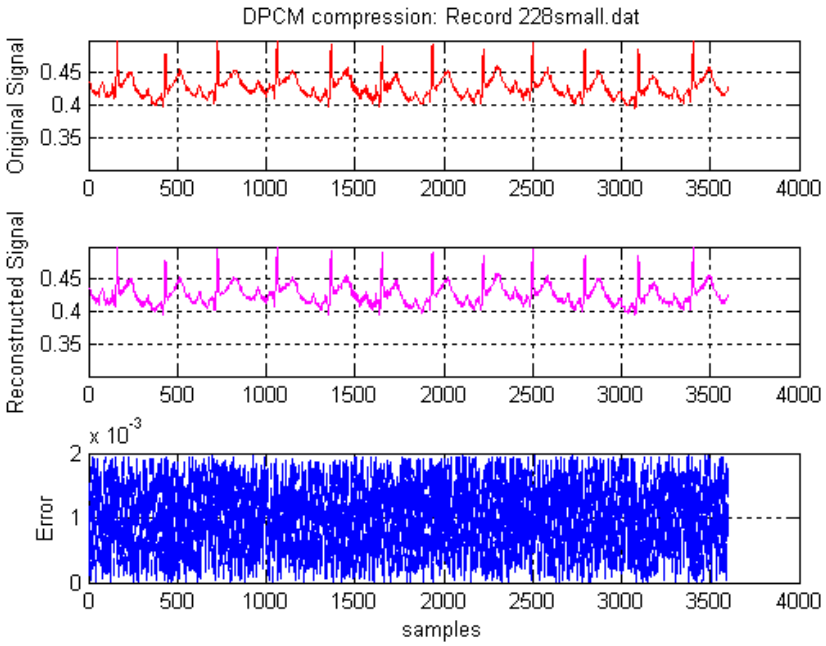


Figure 6.32. DPCM compression of Record 228, MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization, 10 bits.

a subjective visual inspection is not enough to assess the correct operation of a compression system to be used in clinical applications. That is why the values displayed in Table 6.11 are useful, since they provide mathematical data about how good or bad the reconstructed signals are in terms of their difference with the original samples. We can see that the PRD is kept below 8% in every case, which is an acceptable distorsion for most applications.

Nevertheless, the goodness or badness of a compression system must be analyzed not only in terms of the distorsion introduced, but also of course as a function of the reduction of information that it produces. In some applications, higher compression ratios even with the tradeoff of higher distorsion are needed, whereas in other contexts the most important factor is the diagnostical usefulness of the signals and lower compression ratios are tolerated with the condition that distorsion is kept under some maximum level. Hence, it is capital to study the quality attained in the process for a different number of codes. In this sense, we have analyzed how the number of bits utilized in quantization influences the final quality of the reconstructed signal.

Table 6.12 and Figure 6.33 show the quality degradation introduced by the system over a range of different number of bits employed in quantization. Test signal is arbitrarily chosen record 105. We can see by mere visual inspection how the signal loses its diagnostic information completely when few bits are employed for quantizing the residual signal.

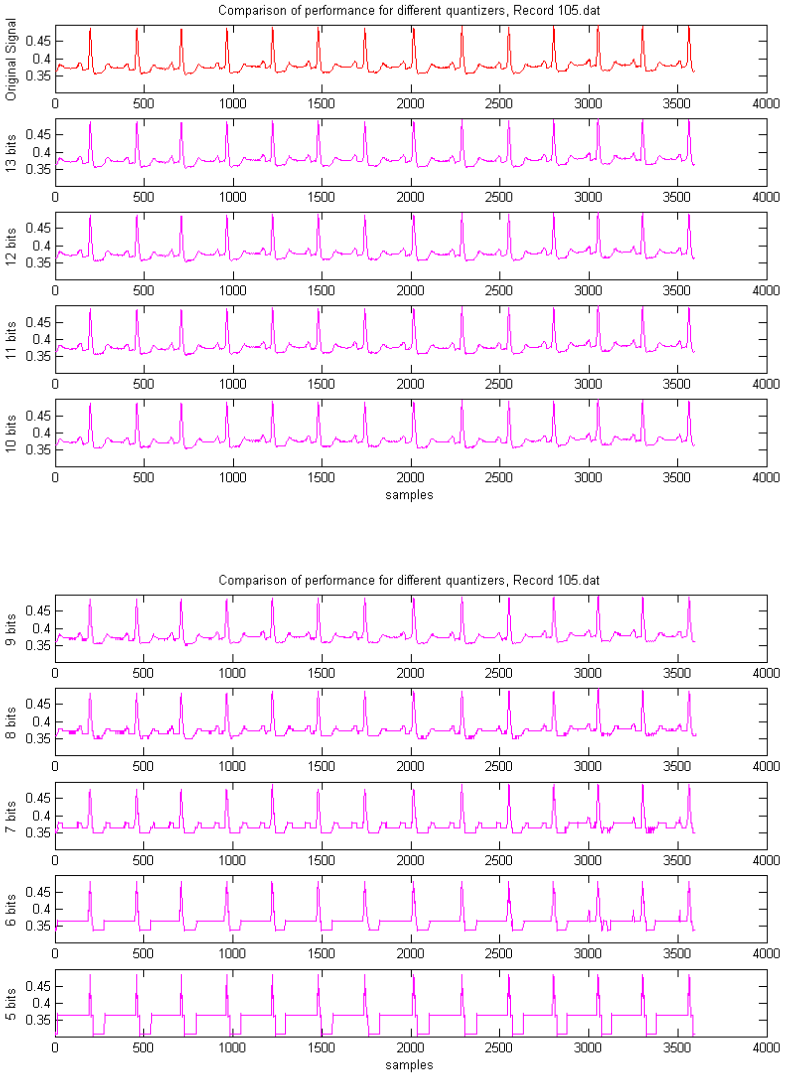


Figure 6.33. DPCM compression of Record 105, MIT-BIH Arrhythmia Database. Linear predictor, order 2. (a) shows the original signal in the top row, and the reconstructed signal for quantizations of 13 to 10 bits. (b) shows the reconstructed signal for quantizations of 9 to 5 bits.

N.bits	CR	PRD (%)	SNR (dB)	rms (x1000)	MAX (x1000)	
13	0	6040.102	1866.0	1413.0	2439.	
12	0.9167	1.1918	88.5949	0.2789	0.4882	
11	1.0000	2.4097	74.5135	0.5638	0.9764	
10	1.1000	4.8470	60.5363	1.1342	1.9531	
9	1.2222	9.5758	46.9186	2.2407	3.9062	
8	1.3750	18.6853	33.5487	4.3722	7.7974	
7	1.5714	39.1584	18.7511	9.1627	15.5758	
6	1.8333	74.0270	6.0148	17.3217	31.1208	
5	2.2000	131.0450	-5.4082	30.6646	62.3808	

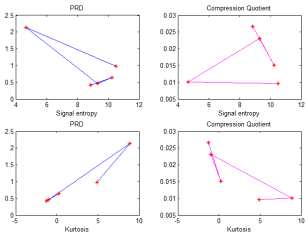
Table 6.12. Distorsion measures of DPCM compression on record 105 from MIT-BIH Arrhythmia Database. Linear predictor, order 2. Quantization: Different number of bits.

In order to confirm our theoretical study (see Section 6.1), it would be useful to effectively compress the different signals and see which ones can be reconstructed with smaller loss of information. Furthermore, by studying practical results we can assess if it is more convenient to fuse ones or others in terms of the distorsion introduced in the process.

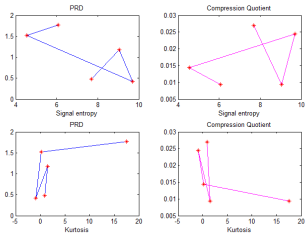
For this we have implemented an M-function called `compquotient.m`, listed as usual in App. D, which calls practically all other routines and functions used throughout this work in order to provide the precise results we are seeking.

We have employed for this purpose several signals from the MGH/MF Waveform Database for being able to study the compression performance on the ECG, arterial pressure, PAP and CVP signals, to which we add the instantaneous heart rate series, computed by our function `HRV.m` which in turn receives the input of QRS positions from the `zeelenberg.m` complexes detector. Entropy and kurtosis of each signal are calculated as reference parameters. After the DPCM encoding and decoding have been launched (`DPCMexp.m` does so), the distorsion measures PRD and signal-to-noise ratio are calculated. The compressibility quotient is computed according to Equation (5.7.1), and plots of resultant PRD and CQ versus entropy and kurtosis are displayed.

Figures 6.34 to 6.37 show these plots for arbitrarily chosen records mgh010, mgh020, mgh035, mgh080, mgh099, mgh156, mgh190 and mgh249, all from the MGH/MF Waveform Database. The compressibility quotient in these plots is normalized by a factor of 100. Table 6.13 shows the average compressibility parameters over the set of records considered.

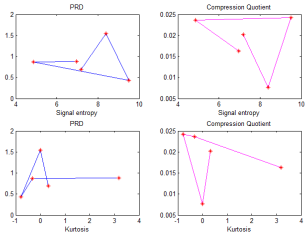


(a) mgh010.dat

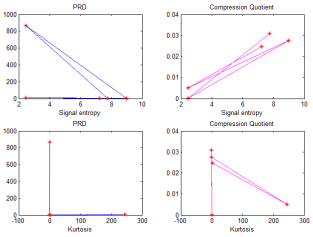


(b) mgh020.dat

Figure 6.34. PRD and CQ versus entropy and kurtosis, records mgh010 and mgh020 from the MGH/MF Waveform Database.

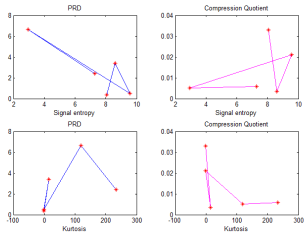


(a) mgh035.dat

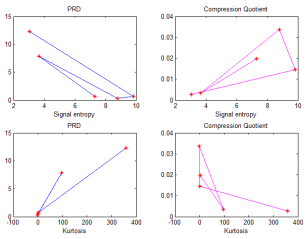


(b) mgh080.dat

Figure 6.35. PRD and CQ versus entropy and kurtosis, records mgh035 and mgh080 from the MGH/MF Waveform Database.

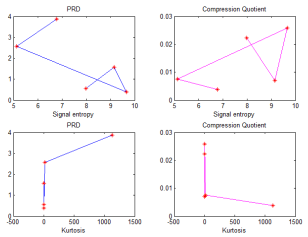


(a) mgh099.dat

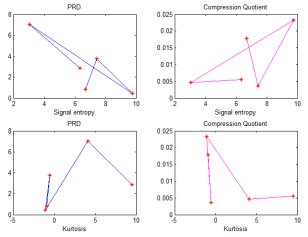


(b) mgh156.dat

Figure 6.36. PRD and CQ versus entropy and kurtosis, records mgh099 and mgh156 from the MGH/MF Waveform Database.



(a) mgh190.dat



(b) mgh249.dat

Figure 6.37. PRD and CQ versus entropy and kurtosis, records mgh190 and mgh249 from the MGH/MF Waveform Database.

	ECG	iHR	ART	PAP	CVP
Average entropy	7.3129	3.9168	9.4107	8.1497	7.1603
Average kurtosis	7.4002	51.3620	−0.7874	0.2962	−0.2185
Average PRD (%)	1.9164	4.0992	0.4279	1.8275	2.2359
Average SNR (dB)	85.2460	67.1380	109.3400	68.1990	97.2580
Average CQ	1.1860	0.9236	2.5408	0.7605	2.2571

Table 6.13. Compressibility parameters of the different signals: ECG, instantaneous HR, ART, PAP and CVP, calculated from the records under test. Quantization: 10 bits for all signals. DPCM compression ratio = 1.1.

You feel depressed and humiliated because you griefly admit that for producing little you need to make a big effort. But with slight differences, that is what happened to everyone in the beginning. Nevertheless, do not become disappointed and work eagerly. Give birth, although it brings harm, to the light of the first new truth, since it will work then, spontaneously, the course which other facts will swift and abundantly flow through.

Santiago Ramón y Cajal, 1852-1934

Chapter 7

Discussion

The first step done in the work that this Thesis comprised was to study the relations among several different types of signals that must cohabit the Ultrasponder control unit. In order to be able to obtain coherent conclusions, the analysis had to be done employing time-correlated signals, with the different records having been obtained simultaneously from a single patient.

For this purpose, we resorted to the MGH/MF Waveform Database provided by Physiobank, which collects 250 recordings of 3-lead ECGs, arterial, pulmonary arterial and central venous pressure, respiratory impedance and CO_2 signals from patients in critical care units, representing a broad spectrum of physiologic and pathophysiologic states. The test signals chosen were an ECG lead and the pressure records (ABP, PAP and CVP).

Our interest was to determine which combination of signals is more worthwile to transmit in terms of their information content. For estimating this we designed a test battery including computation of marginal entropies and kurtosis of each of the four signals considered, crosscorrelation coefficients of each pair of signals and p-values of the statistical p-test of no correlation hypothesis along with 95% confidence intervals for each coefficient, conditional and joint entropies for each pair of signals, and lastly, mutual information of each paired combination of the signals. Although not much information about the underlying relations between the signals can be extracted from kurtosis, its computation was included for completeness of statistical characterization.

In the light of the results, the hypothesis that these signals are all correlated among each other is accepted to be considerably proven, which could not be other way, since the physiological entrainment between them in the origin was known *a priori*.

The first fundamental findings were that the ECG and central venous pressure signals seem to be the ones with smaller marginal entropies, generally. Being the entropy a measure of the information content of a signal, the fact that it is small in average means that the uncertainty of the signal is lower, this is to say, the source producing the signal can be considered a more redundant

source, which means a bigger compressibility of the signal. Thus, the ECG and CVP were found to be good candidates to be stored and processed, since the compression systems can theoretically achieve greater compression ratios on them rather than operating on other signals which entropy is bigger. Note that the entropy rate of a data source measures the average number of bits per symbol needed to encode the sequences generated by the source.

Turning to the analysis of all the possible combinations of pairs of signals, the strongest correlation appeared to be between the ECG signal and the pulmonary arterial pressure, considering the values of the crosscorrelation coefficients as well as the results of the p-tests. Relations between PAP and ABP also seemed to be strong in many of the records, being the correlation between the ECG and CVP next, and in a smaller degree, that between ABP and CVP.

Since the ECG is considered to be the quintessential biomedical signal and virtually all signal processing related to the heart's electrical activity and organ's performance is analyzed on ECG records, it was interesting to study the mutual information between the ECG and the other three signals. The highest mutual information index was repeatedly attained between the ECG and the arterial pressure signal, meaning that these two signals yield a big clarification of uncertainty about the phenomenon they describe. Next in the combined results with ECG was PAP and last, CVP.

This suggests that although the CVP signal showed to be more easily compressible, at least in terms of the upper bound imposed in the compression ratio by its marginal entropy, if we had to choose a single companion to the ECG signal for transmission purposes, ABP or even PAP would be a more appropriate choice in strictly mathematical terms, since their pairing contains more information. Nevertheless, it might be that this mathematical criterion does not really match information usefulness for diagnostic purpose. Not being easily available a measure of diagnostic relevance for the signals in terms of their morphology, fiducial characteristics, even spectral content, etc., we must so far rely on mathematical relations, data redundancy and information measures such as the ones discussed here in order to pick, under the assumption of heavily capacity-constrained environments, the minimum amount of data to be processed.

Taking into account the morphologies of the signals considered, the choice of a DPCM compression system seems adequate. Minimizing the power of the residual signal between actual samples and values predicted by a linear predictor should not imply excessive difficulty.

As discussed above, the ECG is the fundamental and most widely employed biomedical signal. Although there exist methods described in literature for detecting QRS complexes, analyzing the heart rate variability, and performing other monitoring tasks with pressure signals, the biggest amount of signal processing and automated mechanisms in applications related to the heart's activity and performance are done on the basis of ECG records.

Let us now open the discussion on biomedical signal processing with the QRS complexes detection systems.

The two beat detection algorithms we implemented, one based on Engelse and Zeelenerg's method, the other based on Pan and Tompkins' proposal, were compared by exhaustive testing on the complete MIT-BIH Arrhythmia Database, chosen for collecting forty-eight 30 minutes excerpts of two-channel ambulatory ECG recordings obtained from subjects with different pathologies, arrhythmias and abnormal ECGs. This allows a critical testing of the methods implemented.

These methods showed to perform quite differently. In first place, and fundamentally, the Tompkins-based system had to be applied on shortened records of around 1 minute at most in order to function satisfactorily. On the other hand, no problem existed in applying the Zeelenberg detector to the 30 minutes records of the original database.

To represent the goodness of performance of an algorithm by means of its sensitivity and specifity (see Equations (4.4.1) and (4.4.3) in Section 4.4) can bear some problems ([107]). A concrete algorithm can have a high sensitivity but a low specifity, or conversely. In real applications of implantable cardioverters the specifity is more important than the sensitivity, since no patient should be defibrillated due to an analysis error which might cause cardiac arrest. Therefore, a low number of FP decisions should be achieved, even if this process makes the number of FN decisions higher.

Hence, to arrive at a common and single quality parameter, it is preferable to use the receiver operating characteristic (ROC), in which the sensitivity is plotted in dependence of (1 − specifity) for different values of a characteristic parameter in the algorithm, such as magnitude threshold, window length, etc. For comparison purposes between different algorithms and objective evaluation, calculations such as distance *d'* or the area under the ROC curve (IROC) are specifically useful. However, these measures are not as much common as sensitivity, specifity, positive predictivity, etc., on their own, reason for which we chose to include the calculation of Q_α and Matthews correlation coefficient as less biased estimators of the true performance of the algorithms.

The correlation coefficient uses all four numbers TP, TN, FP and FN and may often provide a much more balanced evaluation of the prediction than for instance the percentages or the biased rates (sensitivity, positive predictivity, etc). There are situations, however, in which even the correlation coefficient is unable to provide a completely fair assessment ([65]). The MCC will, for instance, be relatively high in cases where a prediction algorithm gives very few or no FP but at the same time very few TP. A useful observation is that MCC is symmetric with respect to FP and FN.

Tables 6.5 and 6.7 report the results of the detection performed on a shortened version of the MIT-BIH Arrhythmia Database, in which each signal is 10000-samples long ($27.\widehat{7}$ seconds at a sampling frequency of 360 Hz)[1]. Since Zeelenberg algorithm is applicable to longer records, we preferred to analyze as well the performance keeping the original database as source for the test signals,

[1]Record 231 is missing in both cases due to a corruption problem in its correspondant annotations file. Comparison of R waves positions was not possible, hence not being able to calculate sensitivity, positive predictivity, etc.

thus providing more complete and critical results: Table 6.5 reports the results on the complete 650000-samples long records (approximately 30 minutes)[2].

Those tabulated results were used to produce the boxplots in Figures 6.14 to 6.17. The three first figures plot respectively the sensitivity, positive predictivity, specificity, false alarm rate (or false positive rate) and Q_α attained for the cases of the Zeelenberg detector applied to the original records and to the shortened ones, and the Tompkins detector applied as well to the shortened records.

It is immediate to see the much bigger spread in the results of the Zeelenberg-based detector than that of the Tompkins scheme. Furthermore, the application of Zeelenberg algorithm to the shortened records gave considerably poor results, especially for sensitivity and positive predictivity, unlikewise for specifity. Being sensitivity and positive predictivity measures related to the TP count, and checking in Table 6.6 that the number of detected beats does not differ substantially from that of annotated beats in the problematic records which yield the poorest Se and PP results, one can conclude that the badness of the results must be due to a problem of misalignment of the positions sequence calculated by the algorithm in comparison to the annotation files. The algorithm could then be tuned to detect how this misalignment occurs, find out why, and prevent it. However, in the application of Zeelenberg detector to the original full-length records, even though the scattering of the performance parameters is high, the average values improved considerably.

For comparison purposes between the algorithms, the abnormal records in which only one beat is found to be reported in the Physiobank annotations files were removed from the data set, hence the boxplots and statistical calculations shown here do not take them into account.

As shown in Table 7.1, the Tompkins detector proved to perform well in the four fundamental parameters: Se, PP, Sp and FPR. On the other hand and as discussed above, the Zeelenberg detector on the shortened records performed very poorly. The algorithms ought to be compared to the maximum of their capabilities, and thus, the comparison between the Tompkins and the Zeelenberg detectors should be stablished between the best data set they provide, which means considering the application of Zeelenberg scheme to the full-length records. Even like this, the Tompkins detector outperforms it. This fact is supported by the results of t-tests performed under the assumption that the mean of the parameters attained by Tompkins detector are higher than those of the Zeelenberg, providing p-values of 0.1773 for sensitivity, 0.0011 for positive predictivity, 0.0020 for specifity, 0.0028 for Q_α and 0.1464 for Matthews correlation coefficient. The false positive rate was tested with the opposite hypothesis, and the result was as expected, stating its significance with a p-value of 0.0020.

Furthermore, Tompkins algorithm is more robust against noise. Some experiments were done employing records from MIT-BIH Noise Stress Test Database in order to study how the different noise sources and artifacts affect the detection. Records from MIT-BIH Arrhythmia Database were also

[2]Records 214, 215, 228 and 230 are not included due to the same problem referred to above.

		Tompkins	Zeel.orig	Zeel.shrt
Se	mean	80.53	74.30	48.54
	median	90.39	94.89	34.15
	std	23.35	35.08	43.63
PP	mean	88.43	64.47	45.92
	median	97.44	93.92	29.27
	std	25.10	40.52	42.30
Sp	mean	85.16	67.43	85.07
	median	97.72	69.28	90.41
	std	21.19	31.06	13.16
FPR	mean	14.85	32.57	14.93
	median	2.28	30.72	9.59
	std	21.19	31.06	13.16
Q_α	mean	82.84	70.86	66.81
	median	85.31	76.70	61.19
	std	15.27	21.61	21.60

Table 7.1. Statistical parameters of the performance measures for each algorithm. Column Tompkins notes the results for Tompkins detection algorithm whereas columns Zeel.orig and Zeel.shrt note respectively the results for Zeelenberg detection algorithm applied on the full-length records and on the shortened ones.

employed, subsequently decreasing the SNR by artificially introducing noise in the signals. Both algoritms performed similarly in comparative terms for the cases of baseline wander and electromiogram influence, but Tompkins detector showed again to be more powerful against movement artifacts.

For the several reasons discussed above, we must state that of the two algorithms implemented and analyzed, the Tompkins-based scheme gets significantly better results.

Regarding the application of beats classification, two methods have been proposed. Both are time-domain approaches based on beat-to-beat segmentation of the ECG signal by means of a fixed-length window centered on the QRS complex. The common rationale of the methods is the comparison of each new beat with the list of beat templates available by computation of the correlation coefficients, determination of the maximum correlation coefficient to find which template does the beat share the biggest similarity with, comparison with a threshold to determine if the similarity is fair enough, and decision if the beat should be sorted into the group of that given template, in which case the beat contributes to update the template by averaging with the stored waveform, or on the contrary it must be declared as a new template because it does not match well enough in any of the available groups. The first beat is stored as the initial template in order to start running the algorithms.

The two programmed methods showed a distinct behaviour: whereas the algorithm referred to as Classifier I produced a more scattered classification of beats, in which templates regarded as one same type by MIT annotations appeared dissociated in different groups, the Classifier II showed a more compact sorting, reducing the repetition of templates with the same beat type name. It does so by strictly validating only the templates that are product of the averaging of a specified minimum number of beats, given as an input datum to the program. If that condition is not met, the algorithm tries to re-classify the beats again with a lowered threshold to admit more dissimilarity between beats sorted in a group, retrying the process until every template meets the condition of the minimum number of significant beats, or the threshold reaches a lower bound also specified to the method, which was the case in most of the experiments carried out, as the normal situation is that the records of the database employed include some very few beats that are heavily different to the rest of the recording, such as isolated ectopic beats of diverse nature.

This behavior of Classifier II, though it can seem more robust and univocal in the terms of accuracy in the sorting, is not always preferred. For two reasons: first, the progressive lowering of the correlation comparison threshold could lead to dangerous situations in which one could demand the algorithm to sort out together beats that are profoundly dissimilar, just because of the fact that few copies of a beat type occur throughout the record. This can be the case of records with some few beats affected by noise or artifacts, uncorrupted in the rest of their length, or records that present isolated abnormal beats such as premature ventricular contractions, or on the other hand, some few normal beats immersed in bundle branch block episodes. Classifying such dissimilar beats in a common group can be a serious mistake and should not be allowed to happen. That is why the critical threshold limitation is introduced, in order

not to keep retrying the classification reaching ridiculously low values for the comparison threshold, which may ultimately end in the sorting of all beats into one single common group if isolated dissimilar beats are present that prevent the validation of templates by meeting the requirement of the minimum contributing beats.

Secondly, as the process can be too agressive, some groups might be undesirably merged and thus the number of encountered templates reduced over reasonable limits, whereas Classifier I is not only more likely to keep at least one copy of each template type, but it can also refine the coarse sorting by distinguishing slightly different groups of beats, included in the same template type but in different sets, thus providing a finer classification. In this sense, it is interesting to see the grouping of similar beats in Figures 6.18 and 6.19, which show the morphological similarity of beats sorted into a common template. Consider that the waveforms shown in those figures are merely windowed segments of samples centered around the positions of R waves, and do not possess clinically significant information; it is simply their samples values that we use to establish mathematical relations among them.

One could think of getting the best of both methods by different approaches, such as, for instance, if a robust classification with few beat types is needed, simply summing the amount of beats sorted by Classifier I into different sets with the same template name (obtained significantly from the last averaged beat that has met the comparison requirement) and merging them into one group. We would then obtain a less fine classification, but we could be sure that the different beats have been assigned to one or other type of template by exceeding the result of the comparison with a higher threshold than we would finally reach in the case of Classifier II.

Even though a shortened sample of the performance results of both methods is presented in Section 6.4, for avoiding repetitivity of report, the algorithmic differences between the two algorithms are precisely explained in Section 4.5 and the actual different behavior can be seen in Tables 6.8 and 6.9.

The field of beat and arrhythmias classification has received a great amount of attention by researchers in the last few years, as novel algorithms based on sorting by neural networks and different kinds of learning and training methods for sorting trees have been developed. The reader interested on such matter is referred to the reviews and specific articles mentioned in Section 4.5.

Turning to the very up-to-date issue of heart rate variability analysis, the results of our study showed that both time-domain and spectral measures are useful for assessing arrhythmic episodes or normal sinus rythm. HRV analysis is one of the most common signal processing application done on the ECG due to the computation simplicity of well known measures and clinical significance of the information that can be obtained from it.

The time-domain measures SDNN, RMSSD, instantaneous HR and average R-R period, are fast to calculate and can be computed in real-time from relatively few previous data samples. For SDANN and SDNN index periods of 5 minutes length records are necessary, but the measures themselves are fast to compute as well.

The spectral analysis measures require some more effort, since the PSD of the R-R periods or heart rate time series must be estimated with considerable spectral resolution in order to adequately calculate the power distribution over narrow frequency bands. However, its usefulness and the correlation of spectral measures with risk indicators justify their usage in most cases.

Other analitical parameters based on geometrical representations and more complex calculations share the disadvantage of needing a relatively big number of samples in the heart rate time series in order to construct valid statistical distributions and derivated estimates. Their usefulness is indisputable in *a-posteriori* characterization studies or periodic performance reports in Holter systems, but the real-time constrains make them unsuitable for ambulatory monitoring and critical-decision applications in implantable cardioverters.

Closing the discussion on signal processing with purely medical background and relevance, let us now move to the communication signal processing covered in this Thesis.

Out of the coding and compression systems considered for the purposes of the ULTRASPONDER project, a closed-loop DPCM scheme was chosen in light of the results of the mathematical multisignal analysis discussed above and taking into account the characteristic features of the signals to be compressed and transmitted. This was preferred over the open-loop DPCM possibility due to the better time-delay response and synchrony even though the system results slightly more expensive since two prediction filters are needed, one for the encoder and another for the decoder, plus an extra adder, whereas the open-loop scheme only needs one prediction subsystem in the decoder.

The prediction filter, implemented in our study as a linear predictor of order 2 based on a weighted linear combination of previous samples, allows to continuously refine the estimation and minimize the power of the signal to be transmitted, which is the quantized error or residual signal between the predicted sample and the actual sample value.

Regarding the quantization system, a scalar uniform approach was chosen for its simplicity and foreseeing that the dinamic range of the conversion function would not need to be too wide. Out of the mid-rise and mid-tread possibilities, we used in our experiments a mid-rise quantizer.

In [108] a slightly modified version of the mid-tread quantizer described in Section 5.6 is presented reporting satisfactory results. It is called *uniform scalar dead-zone quantizer* (USDZQ), and this dead-zone refers to the extension of the central bin around zero which is enlarged in comparison to the other bins. According to the authors, since many of the samples values the system has to deal with are very small (namely, high-frequency subband wavelet coefficients), when they are quantized to zero there is litte noticeable loss in the reconstructed signal. Therefore, and because the zero-valued quantization indices are efficiently coded using an adaptive arithmetic coder, their aim is to have a larger dead-zone than the central bin would have in a common mid-tread quantizer setting more high-frequency coefficients to zero.

This quantizer is described as follows:

$$I_k = \begin{cases} (-3\delta, -T] & \text{if } k = -1 \\ (-T, T) & \text{if } k = 0 \\ [T, 3\delta) & \text{if } k = 1 \\ [(2k-1)\delta, (2k+1)\delta) & \text{otherwise} \end{cases}$$

$$Q_k = \begin{cases} 0 & \text{if } k = 0 \\ \pm 2k\delta & \text{if } k = \pm 1, \pm 2, ... \end{cases}$$

being k the quantizer output index, I_k the k-th decision interval, R_k its corresponding quantization level, δ half the quantization step size Δ and T the threshold around zero which determines the dead-zone, with $\delta < T < 2\delta$. For any given threshold T, the quantization step size Δ can be calculated by an optimization algorithm to find the best choice in the rate distortion sense. The authors recommend, for the characteristics of the signals employed, Δ in the range of $1.2 - 1.8T$.

As said above, in the system proposed in this Thesis what is quantized is the residual signal $e(n) = x(n) - \hat{x}(n)$ between the original samples and the predicted samples. Therefore, as this does not coincide with the quantized signal in [108], we cannot assess if the employment of the USDZQ would have yielded better results in our case. A statistical analysis of signal $e(n)$ ought to be carried out in order to find out if that choice would be better or not. The spectral content could then show if one or other type of quantizer might be adviceable.

Since the original records of MIT-BIH Arrhythmia Database were digitized using A/D converters with 11-bits resolution, Table 6.12 shows that for a greater or equal number of beats, no compression ratio is achieved. In order to effectively compress the signal, the tradeoff between CR and distorsion, measured in terms of increasing PRD and decreasing SNR or others, must be assumed.

Nevertheless, we can see in the referred Table that the CR achieved by the DPCM system and quantization scheme described are not too impressive for the considerable distorsion introduced. Once the number of bits is lowered under 9, the PRD exceeds 10% and SNR decreases considerably. It is not likely that clinical systems intended for diagnosis can accept such big distorsion values with the advantage of so slightly decreased storage capacity necessities or transmission channel capacity restrictions.

Thus, a straightforward coding improvement could be implemented in our system in order to increase effective compression ratio, namely, entropy coding. This is also used in [108] by decomposing the quantized coefficients in four streams (significance value, sign, position of most significant bit, and residual stream) which have alphabets small enough that allow efficient adaptive entropy coding.

The two most common types of entropy coding are Huffman coding and arithmetic coding. An entropy coder usually exploits the symbol probability distribution of a source independently of previous symbols, which is optimal for

uncorrelated sequences. On the other hand, the DPCM system we used on ur experiments exploits redundancy in data sources in order to approximate the bits employed per sample to the bound of the sequences entropies.

Hence, combining the two systems in a cascaded fashion, the DPCM scheme performs a decorrelation process (which is imperfect) on the samples, while the entropy coding system, say, a Huffman coder, can exploit the remaining dependencies at the same time tht optimizes the average code-length by assigning shorter word-lengths to the most common symbols, and longer lengths to the most rare ones.

This combination is quite common in practical systems, taking the advantage of the consecutive compression processes. Note that entropy coding is lossless, which means that no additional distorsion is introduced to the one provoked by the previous source coding. The compression ratio of the overall system, nevertheless, is benefitted from both processes:

$$CR_{\text{Total}} = CR_{\text{DPCM}} * CR_{\text{entropy encoding}}$$

being the resultant system much more efficient in CR-distorsion terms.

Regarding the study of compressibility performed on ECG signals, blood pressure and instantaneous heart rate, there are some few remarks worth to comment.

We showed in Table 6.13 the average compressibility parameters over the set of records considered. ART and PAP are the signals with highest average entropy, as we already found in the theoretical study in Section 6.1, followed by ECG and CVP. The instantaneous heart rate shows a surprisingly low entropy, much lower than any of the other signals'. This finding suggests that compressing the heart rate should yield good results in terms of distorsion-CR.

However, it can also be seen in that Table that the instantaneous HR is the signal with highest kurtosis, meaning that its waveform peakedness is very superior to that of the other signals. Actually the PRD is also the highest and the signal-to-noise ratio is the lowest in comparison to the ECG and the pressure signals results. This contrasts with the theoretical bounds suggested by the marginal entropy. Nevertheless, this does not mean that information theory and entropy as an information measure (also, a measure for data redundancy) are failing.

On the contrary, the additional parameters we have considered in our study shed light on the answer to this behavior: it is true that the HR-series' entropy is the lowest of the signals considered, so one could expect it to be the signal which would yield the lowest PRD, which is not the case according to the results, but the entropy value only implies that the theoretical limit for the number of bits per sample that should be needed for efficiently encoding the sequence approaches the value of 3.9168. Nonetheless, the markedly peaked shape of the waveform (see Figure 6.20 for some examples of records from MIT-BIH Arrhythmia Database) reflexed in the high value of the kurtosis parameter, provokes that the linear predictor enclosed in the DPCM encoder systematically fails in providing accurate estimates for the next data samples, fact that leads to

the power of the residual signal not being adequately minimized and increasing considerably the rates of PRD.

Thus, we can conclude that although the HR sequence is theoretically highly compressible in light of its average marginal entropy value, a DPCM system proves to be quite inadequate for the purpose. This compression scheme works well for the other signals: ECG and pressures, which lower values of kurtosis make the signals much more "regular" than the instantaneous HR, thus allowing the linear predictor to lock accurately and perform in the minimum-power operation for which DPCM systems are designed and intended to function. Other coding systems should be implemented to exploit compressibility of HR.

Here we can see the usefulness of the novel measure proposed in this work, CQ, which, collecting information about the theoretical compression bounds and actual distorsion results, lowers considerably the expectances we should have about encoding certain signals with a DPCM scheme. Note that the signal with the highest CQ, namely, the arterial blood pressure, is the one with highest average SNR.

Actually, we have found a strong correlation, $\rho = 0.9624$, for such a low p-value as 0.0087 between the compressibility quotient and the signal-to-noise ratio. More exhaustive tests over bigger collections of records should be carried out to totally accept the validity of these findings.

It is also interesting to study the plots of PRD and CQ versus marginal entropy and kurtosis. The aspects of Figures 6.34 to 6.37 suggest that relations not exactly linear but close enough to linear might be found from regressive analysis of course obviating some outliers. It is likely that the instantaneous heart rate's peakedness provokes these outliers.

The finding of the correlation between SNR and CQ is encouraging and might mean that some other mathematical relations are shared between this intrinsically related parameters. One can see how records which plots show vertical concentration of kurtosis, specially around 0, yield considerably higher values of PRD, meaning that signals are not propperly compressed by the DPCM system (the predictor does not work with enough accuracy). By analyzing the scattering of these plots and related ones obtained from a bigger set of data, and computing the slopes of regressive models, etc., we might find interesting conclusions about the compressibility of the signals considered. Further research is left to future work.

Life is the art of drawing sufficient conclusions from insufficient premises.

Samuel Butler, 1835-1970

Let us leave conclusions to imbeciles.

Pío Baroja, 1872-1956

The final conclusion is that we know very little and, however, it is astonishing how much we know. Still more astonishing it is that such a little knowledge can give us so much power.

Bertrand A.W. Russell, 1872-1970

Chapter 8

Conclusion

8.1 Conclusive Comments

As a part of the study related to the ULTRASPONDER Project, we have considered the fusion and cohabitance in the implanted control unit of several signals of distinct nature, such as electrocardiogram recordings, instantaneous heart rate and several different measurements of blood pressure. We established a solid background on the human heart's anatomy and physiology, which is uncommon in many engineering texts, in the introductory chapter in order to understand the medical implications and reasons for waveforms morphology. The ECG signal, along with blood pressure records, were studied in adequate detail to provide enough knowledge to non-medical readers.

With one of our main areas of interest being to analyze how to combine signals from the different sensors and electrodes considered so far in ULTRASPONDER, some information content mathematical measures (entropy, mutual information...) were introduced and computed over the signals under test. Also, correlation among signals was analyzed in order to determine the underlying relations between them, apart from the amount of information carried by each of them alone. This theoretical study shed some light on the compression possibilities and encoding schemes to be used.

A complementary effort was put into the chapter of ECG signal processing, presenting two QRS complexes detection systems, based respectively in Zeelenberg's and Tompkins' algorithms, and a beat classification algorithm with a modified version based on template-correlation approaches. Performance reports of the detection systems were done in terms of several parametrs, namely, sensitivity, specifity, positive predictivity, false alarm rate, Q_α and Matthews correlation coefficient, the latter of which are not widely spread in biomedical literature. We also saved a section for heart rate variabilty analysis, by means of mainly time-domain and spectral measures computation.

In light of the results obtained in our theoretical study, and after providing a general insight on data compression, a DPCM system was proposed

to compress and perform source coding tasks in application to ECG signals. A closed-loop scheme with feedback around the quantizer was chosen, with a linear predictor as the estimating element and mid-rise uniform quantization of the residual signal between actual and predicted samples.

Simulations were done in order to study the effective distorsion introduced by the lossy compression method as a function of the quantizer resolution and for the different signals, employing PRD and SNR among others as performance measures. We introduced a novel parameter, the "compressibility quotient" (CQ) which relates information about the theoretical compression bounds imposed by sample entropy and at the same time, performance data in terms of the distorsion introduced by the encoding process. By studying the plots involving PRD, CQ, kurtosis and entropy, we did encouraging findings and assessed the correlation between CQ and the signal-to-noise ratio.

8.2 Limitations

As it occurs in most of the biomedical engineering field research, we have used a wide but restricted dataset from the Physiobank collection for our experiments, and thus the performance of the different methods might be favorably biased by the tuning and algorithm design that occurs during the normal phase of development. The algorithms we have presented might have worse performance if applied prospectively to different datasets.

We have tried to use as more different databases as possible during the training phase and experimental running of our modular M-functions, but it is unavoidable to let many cases not contemplated and not exploit all the possibilities of existent annotated databases in order to cover a wider sample of hypothetical situations.

Furthermore, our theoretical analysis and compression simulations have been limitted to ECG, heart rate and blood pressure signalsindexECG, not considering other different signals that might be of interest for the purposes of Ultrasponder Project, in some of which medical personnel might not yet be interested nowadays, but which application may arise after some findings or hypothesis formulation. Particularly, spatial and geometric measures such as ventricular dimension, IVS displacement, etc., are considered in up-to-date studies regarding implications in systolic function, monitoring of hemodynamic parameters, etc., but the difficulty of retrieving this kind of waveforms with correlated ECG and blood pressure signals forced us to unwillingly reject their inclusion in our studies.

Lastly, as it was commented in the Discussion chapter, the DPCM encoding system is usually cascaded with a further entropy coding stage in order to increase the compression ratio obtained by the overall system. We did not focus on that issue, understanding that Huffman or arithmetic coding provide lossless compression methods in which not much improvement or further interesting conclusions are likely to be extracted.

8.3 Future Lines of Research

Many paths are yet unopened, and many others are opened but still ought to be weeded.

Most specially, it would be interesting to perform a more complete study considering other signals of diverse nature that can be lightly processed in the control unit implanted underneat the patient's skin before being transmitted to the external PC. Time will say which parameters and signals are more worthful to be included in such a system. Time, and of course sensors technology. Ventricular displacement is being studied as a possible factor with incidence in chronic heart diseases and severe failure, so when publicly available databases include these recordings they will be undoubtlessly studied thoroughly to determine the possibility of data fusion along with the other signal types already considered nowadays.

Research described in [109] and [110] propose data fusion methods based on the value of information that data from different sensors provide. They are based on entropies, mutual information, correlation among signals, etc. as we have described in this work. Other distance measures are also considered, reporting actual results obtained in practical systems. It is encouraging to see that data and sensor fusion theory is finding support in many engineering fields, not only biomedicine.

The novel measure we introduced in this work, CQ, proved to have a strong correlation with SNR, but it double-sidedly handles theoretical information of compressibility limits and actual performance results in terms of distorsion. Its physical meaning is not clear, but it can be used after simulations on some signals are carried out, in order to more accurately determine practical compressibility of signals sharing the same nature. Regressive modelling applied to the relations between some of the parameters considered as distorsion measures or estimates of information content might shed more light on usefulness or superfluousity of this measure.

Even though all methods studied in this Thesis have been *a-posteriori* applied onto stored records wich are part of *de facto* standarized databases, the interest of biomedical signal processing are mainly real-time applications, like ambulatory monitoring, cardioverters control, etc. Thus, the usefulness of all algorithms or parameters computation should be assessed in terms of their applicablity to real-time contexts, always allowing for further optimization and improvement.

Suffice these brief and partially correlated comments to suggest only some of the infinite possible future research lines that might become interesting in a short-term perspective.

Part IV

Appendices

There are three kinds of lies: lies, damned lies, and statistics.

Attributed to Benjamin Disreali (1804-1881) by Mark Twain in 1907, but origins remain unclear

Statistics, like veal pies, are good if you know the person that made them, and are sure of the ingredients.

Abbott Lawrence Lowell, 1909

Appendix A

Basic Statistics

This brief Appendix is included as a quick reference of the statistics' formulas used through the text and some few others. Our aim is not to do a comprehensive compilation of all statistic parameters commonly employed in engineering, since each field and application usually prefers a concrete subset, neither to provide formal definitions of the statistics, but rather to specify their mathematical formulation so that the reader can know in any moment what calculation we are referring to. We limit this compilation to the most commonly used central tendency and dispersion measures.

During the Appendix we will consider the equivalence of formulas defined over a numeric data set with statistics referred to a number of sample values taken from a discrete signal, which are in the end numeric values chosen to represent the continuous signal and can be treated statistically just as any other data set.

We will assume that if x is a data set including N values, x_i, with $i = 1..N$, denote each value. Similarly, if the discrete sequence $x[n]$ (often written simply as $x(n)$) of length N is a digitized version of the continuous signal $x(t)$, the different samples are denoted by $x[n = i] = x[i]$, with again $i = 1..N$. The notation $x(i)$ is, though, more common.

A.1 Central tendency measures

An *average* or *central tendency* of a data set is a measure of the middle or expected value of the data set. Average measures include the different means, the median and the mode.

The *arithmetic mean* μ_x or $\bar{x}$ of $x(n)$ is defined as:

$$\mu_x = \bar{x} = \frac{1}{N}\sum_{i=1}^{N} x(i)$$

The *geometric mean* GM_x of $x(n)$ is defined as:

$$GM_x = \sqrt[N]{\prod_{i=1}^{N} x(i)}$$

The *harmonic mean* HM_x of $x(n)$ is defined as:

$$HM_x = \frac{1}{\frac{1}{N}\sum_{i=1}^{N}\frac{1}{x(i)}}$$

The *median* M_x of $x(n)$ is the value that lets the same number of samples $x(i)$ being greater than being smaller than it. That is to say, once the $x(i)$ are sorted, M_x is the middle value. As it does not consider the actual values of the samples for doing any calculation, but simply to sort them in the greater-than and smaller-than subsets, the median is not affected by scattering in the samples, being thus less sensitive than the means to oscillation in the values, or to far greater or smaller samples than the rest of the seat. It is calculated as:

$$M_x = \begin{cases} x(\frac{N+1}{2}) & \text{if } N \text{ is odd} \\ \frac{x(\frac{N}{2})+x(\frac{N}{2}+1)}{2} & \text{if } N \text{ is even} \end{cases}$$

The *mode* Md_x of $x(n)$ is the most repeated sample value of the set.

$$Md_x = x(i) \quad \text{, if } n_{x(i)} = \max\{f_j, j \in \{1, 2, ..., k\}\}$$

where $n_{x(i)}$ denotes the number of occurances of sample $x(i)$ in the whole sequence $x(n)$, k is the number of different samples included in the set, and f_j denotes the frequency of occurance of sample $x(j)$. The mode is then, again, independent of the bigger part of the data, thus being quite sensitive to samples values variation. If several values share the maximum frequency of appearence, they are all considered to be modes.

The *expected value* $E[X]$ or $\bar{X}$ of $x(n)$ considering the random variable X as the set of occurances of samples $x(i)$ is the integral of the random variable X with respect to its probability measure. Assuming X is a discrete random variable with probability mass function given by $p(x)$, then:

$$E[X] = \bar{X} = \sum_{i=1}^{N} x(i) \cdot p(x(i))$$

If the probability distribution of X would admit a probability density function given by $f(x)$, then:

$$E[X] = \bar{X} = \int_{-\infty}^{\infty} x f(x) dx$$

The expected value of a discrete random variable coincides with the arithmetic mean of its samples, and hence $E[X] = \mu_x$.

A.2 Dispersion measures

Dispersion measures give an idea of the variability or spread in the values of a data set, or in the samples of a discrete signal.

The *range* of $x(n)$ is the length of the smallest interval which contains all different $x(i)$ sample values.

$$Range\{x\} = \max\{x(n)\} - \min\{x(n)\}$$

The *interquartile range*, also called the *midspread*, and noted IQR is more robust than the total range and often preferred to it, since it measures the width of the two inner quarters of the points in the data set, specifying the bounds which hold half of the samples. By ignoring the outer quarters of the data set, IQR is much less sensitive to outliers. To find the IQR of a data set, the set is sorted it and the middle 50% of the points are retained and the rest are discarded. The difference between the values at either end of the middle 50% of points is the interquartile range.

The *standard deviation* $Std(X)$ or σ_x of $x(n)$ is the square root of mean squared error of each sample respect the mean of the set:

$$Std(X) = \sigma_x = \sqrt{E\left[(x(i) - \mu_x)^2\right]} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(x(i) - \mu_x)^2}$$

The *variance* $Var(X)$ or σ_x^2 of $x(n)$ is defined as the standard deviation being its positive square root. Thus:

$$Var(X) = \sigma_x^2 = E\left[(x(i) - \mu_x)^2\right] = \frac{1}{N}\sum_{i=1}^{N}(x(i) - \mu_x)^2$$

A useful property of standard deviation is that, unlike variance, it is expressed in the same units as the data. Both standard deviation and variance are independent of constant addition or substraction to the whole set of samples, meaning that they are actually independent of the central tendency or mean of the samples. This is called location-invariancy. The standard deviation is furthermore linear in scale, but the variance is not. This is summed up as follows:

$$X \text{ d.R.V.}, \quad X : i \mapsto x(i) \quad \begin{cases} \mu_x \\ \sigma_x \\ \sigma_x^2 \end{cases}$$

$$\forall a, b \in \Re$$

$$Y \text{ d.R.V.}, \quad Y : i \mapsto y(i) = ax(i) + b \quad \begin{cases} \mu_y = a\mu_x + b \\ \sigma_y = a\sigma_x \\ \sigma_y^2 = a^2\sigma_x^2 \end{cases}$$

In relation to the variance, *covariance* is a measure of how much two random variables change together. Of course, it coincides with the variance when the two variables analyzed are identical. If X and Y are two random variables with means μ_x and μ_y, then the covariance between X and Y, noted $Cov(X, Y)$ is:

$$Cov(X, Y) = E\left[(X - \mu_x)(Y - \mu_y)\right] = E\left[X \cdot Y\right] - \mu_x \mu_y$$

If X and Y are statistically independent, $E\left[X \cdot Y\right] = 0$, and $Cov(X, Y) = 0$. Random variables whose covariance is zero are called uncorrelated. If two variables tend to vary together, the covariance between them will be positive; opposedly, if one tends to be above its expected value when the other one tends to be below its expected value, the covariance will be negative.

Autocovariance is the covariance of a signal against a time-shifted version of itself. If each state of the series has a mean $E[X] = \mu_t$, then the autocovariance is defined as:

$$K_{XX}(t, s) = E\left[(X_t - \mu_t)(X_s - \mu_s)\right] = E\left[X_t \cdot X_s\right] - \mu_t \mu_s$$

Some *relative dispersion measures*, which give information about the spread of the data considering at the same time the location of the samples values include:

The *coefficient of variation*, a normalized and dimensionless measure, noted $CV(X)$ or $c_v x$, only defined for non-zero mean sets:

$$CV(X) = c_v x = \frac{\sigma_x}{\mu_x}$$

The *variation-to-mean ratio*, noted $VMR(X)$ or VMR_x, also called *index of dispersion*:

$$VMR(X) = VMR_x = \frac{\sigma_x^2}{\mu_x}$$

The *signal-to-noise ratio*, also dimensionless, noted $SNR(X)$ or SNR_x in the statistical sense:

$$SNR(X) = SNR_x = \frac{\mu_x}{\sigma_x}$$

The *Fano factor*[1], noted F_x, which is a time-windowed version of the $VMR(X)$ defined above. It is also called *windowed VMR*.

$$F_x = \frac{\sigma_{x,W}^2}{\mu_{x,W}}$$

where W denotes the window chosen. The Fano factor equals the VMR if the time window is chosen to be infinity.

[1] Named after Ugo Fano, Italo-American physicist.

Prediction is very difficult, especially about the future.

Niels Bohr, 1885-1962

Appendix B

Polynomial Predictors and Interpolators

This Appendix provides the basic knowledge on polynomial predictors and interpolators needed to follow the discussion on direct ECG data compression techniques carried out in Section 5.3. Most of its content has been extracted from [89].

Note that the order of the polynomial predictors/interpolators starts with the zero-order, whereas the order of the linear predictor starts with the first-order.

In general, either type of compression algorithms with an order higher than one are rarely used. This is due to the fact that increasing the order of the linear interpolator or predictor above two and more does not result in a significant increase in data compression of ECG's. There is still no clear answer as to whether interpolators or predictors are more efficient, since reported results found in literature are contradictory.

B.1 Polynomial Predictors

Polynomial predictors are based on a finite difference technique which constraints an nth-order polynomial to $K+1$ data points. Predicted data are obtained by extrapolating the polynomial one sample point at a time, according to:

$$\hat{y}_n = y_{n-1} + \Delta y_{n-1} + \Delta^2 y_{n-1} + \ldots + \Delta^k y_{n-1} \tag{B.1.1}$$

where:

$$\begin{aligned} \hat{y}_n &= \text{predicted sample point at time } t_n \\ y_{n-i} &= \text{sample value at time } t_{n-i} \\ \Delta y_{n-1} &= y_{n-1} - y_{n-2} \\ \Delta^k y_{n-1} &= \Delta^{k-1} y_{n-1} - \Delta^{k-1} y_{n-2} \end{aligned}$$

and k represents the order of the polynomial prediction algorithm. Only past sample values are considered to make the prediction.

***Zero-Order Predictor* (ZOP)** : The ZOP is a polynomial predictor with $k = 0$. Therefore, the predicted value is merely the previous data point.

$$\hat{y}_n = y_{n-1} \tag{B.1.2}$$

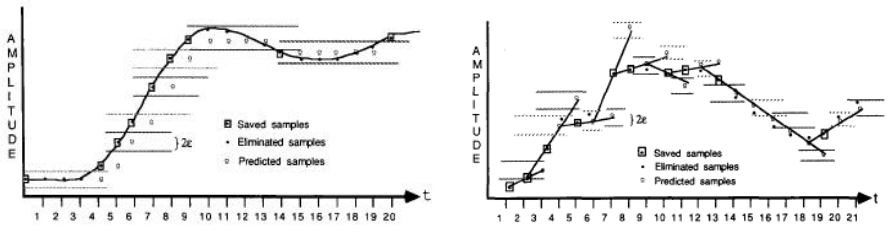
Despite its simplicity, ZOP has proven to be very efficient for step-like data. The most efficient ZOP technique uses a *floating aperture* wherein a tolerance band $\pm\epsilon$ is centered around the last saved data point. Succeeding sample points that lie in this band are not retained. These samples are approximated by a horizontal line of an amplitude equal to the previous saved sample point. Thus, the original data samples are substituted for the line parameters, amplitude and length (number of data points). Signal reconstruction is achieved by expanding the stored line parameters to discrete data points.

***First-Order Predictor* (FOP)** : The FOP is a polynomial predictor with $k = 1$. In this way, the predicted value is a point on the straight line drawn between the last two data points y_{n-1} and y_{n-2}.

$$\hat{y}_n = 2y_{n-1} - y_{n-2} \tag{B.1.3}$$

The FOP algorithm with a *floating aperture* is initiated by retaining the first two data points and drawing a straight line between them. An aperture of width $\pm\epsilon$ is centered around the obtained line. If the actual sample point y_n is within this band, the sample is not saved but approximated for the amplitude of the line in t_n. Otherwise, y_n is saved and a new prediction line is drawn through y_n and $\hat{y}_{n-1}$. Signal reconstruction requires the nonredundant sample values along with the corresponding time instants.

Figure B.1 shows the operation of the zero-order predictor and the first-order predictor over an arbitriary signal in terms of retained and eliminated samples.



(a) ZOP with floating aperture **(b)** FOP with floating aperture

Figure B.1. Illustration of the zero-order (ZOP) and first-order (FOP) prediction algorithms with floating aperture. Extracted from [89].

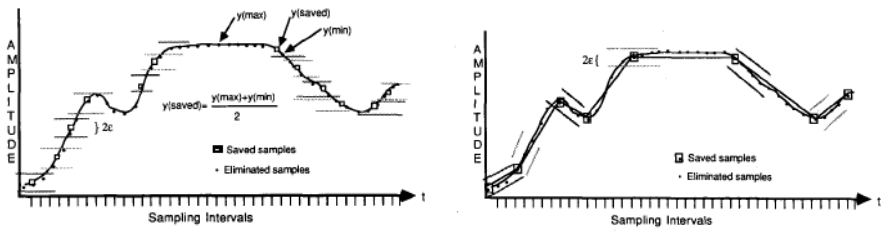
B.2 Polynomial Interpolators

Polynomial interpolators use both past and future data points to decide whether the actual sample point is redundant or not, so that all samples between the last retained sample and the present one affect the interpolation.

Zero-Order Interpolator **(ZOI)** : Similarly to the ZOP case, in the ZOI a horizontal line is used to determine the largest set of consecutive data points within a preset error threshold, but the way of selecting the sample point that represents the reundandant set changes. The interpolator retained sample is determined at the end of the redundant set, in contrast to the first sample in the case of the predictor. Moreover, the saved sample in the interpolation algorithm is computed as the average between the minumum and the maximum sample values in the redundant set. Whenever the current sample point lies outside the tolerance band, the current sample becomes the first point in a new set and the average between the largest and the smallest values of the previous redundant samples is saved.

First-Order Interpolator **(FOI)** : The FOI is also similar to its correspondant FOP case. A straight line is drawn to establish a slope, and it is assumed that data will continue in its direction. The line is drawn between the present sample and the last saved sample, so all intermediate data points are within a specified tolerance of the interpolated value given by the line. The ending point of a line is used as the starting point of the next line segment, so only one data point (the ending point) needs to be retained for each segment after the very first saved line. If the actual Kth sample value after the last retained value differs from the interpolated value by the line by a quantity greater than the preset tolerance, then the $(K-1)$th sample is saved and the process is repeated. The waveform is reconstructed by connecting the nonredundant (saved) samples with straight lines. This scheme is also called FOI-2DF, which stands for first-order interpolator with two degrees of freedom.

Figure B.2 shows the operation of the zero-order interpolator and the first-order interpolator over an arbitriary signal in terms of retained and eliminated samples.



(a) ZOI with floating aperture **(b)** FOI with floating aperture

Figure B.2. Illustration of the zero-order (ZOI) and first-order (FOI) interpolation algorithms with floating aperture. Extracted from [89].

Appendix C

Information Measures Results

In this Appendix we include the results of information measures calculated on six arbitrary records of the MGH/MF Waveform Database. For explanation of the parameters calculated and interpretation of the results, turn to Sections 6.1 and 3.1.

```
fnn =

mgh001.dat

ecg001=ecg(1,3e4:3.8e4);
art001=ecg(4,3e4:3.8e4);
pap001=ecg(5,3e4:3.8e4);
cvp001=ecg(6,3e4:3.8e4);

infoanalysis(ecg001,art001,pap001,cvp001)
entropies =
   10.0293    9.3791    9.2736    7.6777
kurtosix =
    0.5228   -0.9674    0.1595    1.4269
rho =
    1.0000   -0.0035   -0.1552    0.0545
   -0.0035    1.0000   -0.1006    0.0740
   -0.1552   -0.1006    1.0000   -0.0449
    0.0545    0.0740   -0.0449    1.0000
P =
    1.0000    0.7525    0.0000    0.0000
    0.7525    1.0000    0.0000    0.0000
    0.0000    0.0000    1.0000    0.0001
    0.0000    0.0000    0.0001    1.0000
rlo =
    1.0000   -0.0254   -0.1765    0.0326
   -0.0254    1.0000   -0.1222    0.0521
   -0.1765   -0.1222    1.0000   -0.0668
    0.0326    0.0521   -0.0668    1.0000
rup =
    1.0000    0.0184   -0.1338    0.0763
    0.0184    1.0000   -0.0788    0.0957
```

```
   -0.1338   -0.0788    1.0000   -0.0230
    0.0763    0.0957   -0.0230    1.0000
mutualI =
   10.0293    6.4782    6.3798    4.8440
    6.4782    9.3791    5.8466    4.2877
    6.3798    5.8466    9.2736    4.2071
    4.8440    4.2877    4.2071    7.6777
condentropies =
         0    3.5511    3.6495    5.1853
    3.5511         0    3.5325    5.0913
    3.6495    3.5325         0    5.0665
    5.1853    5.0913    5.0665         0
jointentropies =
   10.0293   12.9302   12.9231   12.8630
   12.9302    9.3791   12.8060   12.7691
   12.9231   12.8060    9.2736   12.7442
   12.8630   12.7691   12.7442    7.6777


fnn =

mgh002.dat

ecg002=ecg(1,1:1e4);
art002=ecg(4,1:1e4);
pap002=ecg(5,1:1e4);
cvp002=ecg(6,1:1e4);

infoanalysis(ecg002,art002,pap002,cvp002)
entropies =
    9.2460   10.0770    9.2791    8.7022
kurtosix =
    2.9387   -1.2937    4.4425   -1.2746
rho =
    1.0000    0.0103    0.1105   -0.0062
    0.0103    1.0000   -0.0400   -0.0109
    0.1105   -0.0400    1.0000    0.0044
   -0.0062   -0.0109    0.0044    1.0000
P =
    1.0000    0.3035    0.0000    0.5347
    0.3035    1.0000    0.0001    0.2768
    0.0000    0.0001    1.0000    0.6600
    0.5347    0.2768    0.6600    1.0000
rlo =
    1.0000   -0.0093    0.0911   -0.0258
   -0.0093    1.0000   -0.0595   -0.0305
    0.0911   -0.0595    1.0000   -0.0152
   -0.0258   -0.0305   -0.0152    1.0000
rup =
    1.0000    0.0299    0.1299    0.0134
    0.0299    1.0000   -0.0204    0.0087
    0.1299   -0.0204    1.0000    0.0240
    0.0134    0.0087    0.0240    1.0000
mutualI =
    9.2460    6.0625    5.2930    4.7588
    6.0625   10.0770    6.1381    5.5894
    5.2930    6.1381    9.2791    4.8803
    4.7588    5.5894    4.8803    8.7022
condentropies =
         0    3.1836    3.9530    4.4872
    3.1836         0    3.9389    4.4876
    3.9530    3.9389         0    4.3988
    4.4872    4.4876    4.3988         0
```

```
jointentropies =
    9.2460   13.2606   13.2322   13.1895
   13.2606   10.0770   13.2180   13.1899
   13.2322   13.2180    9.2791   13.1011
   13.1895   13.1899   13.1011    8.7022


fnn =

mgh003.dat

ecg003=ecg(1,1:1e4);
art003=ecg(4,1:1e4);
pap003=ecg(5,1:1e4);
cvp003=ecg(6,1:1e4);

infoanalysis(ecg003,art003,pap003,cvp003)
entropies =
    7.7573    8.4682    9.0224    7.6604
kurtosix =
    6.7320   -0.8432    0.6240    2.4318
rho =
    1.0000    0.0068   -0.1606    0.1855
    0.0068    1.0000    0.0819    0.0452
   -0.1606    0.0819    1.0000    0.1479
    0.1855    0.0452    0.1479    1.0000
P =
    1.0000    0.4942    0.0000    0.0000
    0.4942    1.0000    0.0000    0.0000
    0.0000    0.0000    1.0000    0.0000
    0.0000    0.0000    0.0000    1.0000
rlo =
    1.0000   -0.0128   -0.1796    0.1665
   -0.0128    1.0000    0.0624    0.0256
   -0.1796    0.0624    1.0000    0.1287
    0.1665    0.0256    0.1287    1.0000
rup =
    1.0000    0.0264   -0.1414    0.2043
    0.0264    1.0000    0.1013    0.0647
   -0.1414    0.1013    1.0000    0.1670
    0.2043    0.0647    0.1670    1.0000
mutualI =
    7.7573    3.2055    3.6786    2.6360
    3.2055    8.4682    4.4531    3.3178
    3.6786    4.4531    9.0224    3.7450
    2.6360    3.3178    3.7450    7.6604
condentropies =
         0    4.5518    4.0787    5.1213
    4.5518         0    4.0151    5.1504
    4.0787    4.0151         0    5.2774
    5.1213    5.1504    5.2774         0
jointentropies =
    7.7573   13.0200   13.1011   12.7818
   13.0200    8.4682   13.0375   12.8109
   13.1011   13.0375    9.0224   12.9379
   12.7818   12.8109   12.9379    7.6604


fnn =

mgh004.dat

ecg004=ecg(1,4e5:4.2e5);
```

```
art004=ecg(4,4e5:4.2e5);
pap004=ecg(5,4e5:4.2e5);
cvp004=ecg(6,4e5:4.2e5);

infoanalysis(ecg004,art004,pap004,cvp004)
entropies =
    8.1843    9.8409    9.3473    7.8830
kurtosix =
   27.0492   -1.0284   -0.5233   -0.3509
rho =
    1.0000    0.0102   -0.1097   -0.1652
    0.0102    1.0000    0.0881    0.0664
   -0.1097    0.0881    1.0000    0.2857
   -0.1652    0.0664    0.2857    1.0000
P =
    1.0000    0.1500    0.0000    0.0000
    0.1500    1.0000    0.0000    0.0000
    0.0000    0.0000    1.0000         0
    0.0000    0.0000         0    1.0000
rlo =
    1.0000   -0.0037   -0.1233   -0.1787
   -0.0037    1.0000    0.0743    0.0526
   -0.1233    0.0743    1.0000    0.2730
   -0.1787    0.0526    0.2730    1.0000
rup =
    1.0000    0.0240   -0.0959   -0.1517
    0.0240    1.0000    0.1018    0.0801
   -0.0959    0.1018    1.0000    0.2984
   -0.1517    0.0801    0.2984    1.0000
mutualI =
    8.1843    3.8614    3.4186    2.3309
    3.8614    9.8409    5.0269    3.7502
    3.4186    5.0269    9.3473    3.3375
    2.3309    3.7502    3.3375    7.8830
condentropies =
         0    4.3230    4.7657    5.8534
    4.3230         0    4.8140    6.0907
    4.7657    4.8140         0    6.0098
    5.8534    6.0907    6.0098         0
jointentropies =
    8.1843   14.1638   14.1131   13.7363
   14.1638    9.8409   14.1613   13.9737
   14.1131   14.1613    9.3473   13.8928
   13.7363   13.9737   13.8928    7.8830


fnn =

mgh005.dat

ecg005=ecg(1,7e5:7.2e5);
art005=ecg(4,7e5:7.2e5);
pap005=ecg(5,7e5:7.2e5);
cvp005=ecg(6,7e5:7.2e5);

infoanalysis(ecg005,art005,pap005,cvp005)
entropies =
    8.0793    8.8872    8.8393    7.7189
kurtosix =
    8.3313   -0.7174   -0.4821   -0.3143
rho =
    1.0000    0.1432    0.0109    0.1715
    0.1432    1.0000    0.0572    0.2908
```

```
    0.0109    0.0572    1.0000    0.3744
    0.1715    0.2908    0.3744    1.0000
P =
    1.0000    0.0000    0.1239    0.0000
    0.0000    1.0000    0.0000         0
    0.1239    0.0000    1.0000         0
    0.0000         0         0    1.0000
rlo =
    1.0000    0.1296   -0.0030    0.1580
    0.1296    1.0000    0.0434    0.2781
   -0.0030    0.0434    1.0000    0.3624
    0.1580    0.2781    0.3624    1.0000
rup =
    1.0000    0.1567    0.0247    0.1849
    0.1567    1.0000    0.0710    0.3034
    0.0247    0.0710    1.0000    0.3862
    0.1849    0.3034    0.3862    1.0000
mutualI =
    8.0793    2.9453    2.8606    2.0355
    2.9453    8.8872    3.6948    2.8138
    2.8606    3.6948    8.8393    2.7199
    2.0355    2.8138    2.7199    7.7189
condentropies =
         0    5.1340    5.2187    6.0439
    5.1340         0    5.1924    6.0733
    5.2187    5.1924         0    6.1194
    6.0439    6.0733    6.1194         0
jointentropies =
    8.0793   14.0212   14.0580   13.7628
   14.0212    8.8872   14.0317   13.7922
   14.0580   14.0317    8.8393   13.8383
   13.7628   13.7922   13.8383    7.7189


fnn =

mgh006.dat

ecg006=ecg(1,4.6e5:4.8e5);
art006=ecg(4,4.6e5:4.8e5);
pap006=ecg(5,4.6e5:4.8e5);
cvp006=ecg(6,4.6e5:4.8e5);

infoanalysis(ecg006,art006,pap006,cvp006)
entropies =
    7.8703    9.5407    9.3365    8.2787
kurtosix =
    1.1799   -1.2454    0.4125   -0.5205
rho =
    1.0000    0.0612   -0.4673   -0.0828
    0.0612    1.0000   -0.0379   -0.0046
   -0.4673   -0.0379    1.0000    0.1556
   -0.0828   -0.0046    0.1556    1.0000
P =
    1.0000    0.0000         0    0.0000
    0.0000    1.0000    0.0000    0.5113
         0    0.0000    1.0000    0.0000
    0.0000    0.5113    0.0000    1.0000
rlo =
    1.0000    0.0473   -0.4781   -0.0965
    0.0473    1.0000   -0.0517   -0.0185
   -0.4781   -0.0517    1.0000    0.1420
   -0.0965   -0.0185    0.1420    1.0000
```

```
rup =
    1.0000    0.0750   -0.4564   -0.0690
    0.0750    1.0000   -0.0240    0.0092
   -0.4564   -0.0240    1.0000    0.1690
   -0.0690    0.0092    0.1690    1.0000
mutualI =
    7.8703    3.2957    3.1189    2.2541
    3.2957    9.5407    4.6832    3.7199
    3.1189    4.6832    9.3365    3.5456
    2.2541    3.7199    3.5456    8.2787
condentropies =
         0    4.5746    4.7514    5.6162
    4.5746         0    4.8575    5.8208
    4.7514    4.8575         0    5.7909
    5.6162    5.8208    5.7909         0
jointentropies =
    7.8703   14.1153   14.0879   13.8948
   14.1153    9.5407   14.1940   14.0995
   14.0879   14.1940    9.3365   14.0696
   13.8948   14.0995   14.0696    8.2787
```

– Don't you understand? I need the codes. I have to get inside Zion, and you have to tell me how.

Dialogue between Agent Smith and Morpheus, from the film *The Matrix*, 1999

Appendix D

Code of M-functions

This Appendix lists the code of the different M-functions programmed in the MATLAB environment implementing the algorithms and methods described in their correspondent sections. Some auxiliary functions are also included, which were employed for solving the several problems approached in this Thesis. Usually the source code is commented for increasing readability.

D.1 Multisignal Information Measures

We have mainly used three own m-functions: `multisignalplot.m`, which simply plots four time series, labelling them and calculating a time array specially for the case studied in this matter in Section 6.1, `multisignalplot2.m`, a slightly more complex version which allows plotting the eight signals included in each record of the MGH/MF Waveform Database, and finally `infoanalysis.m`, that performs the matricial calculations shown in Appendix C. The latter computes the information measures parameters using the package MutualInfo 0.9 written by Hanchuan Peng and offered by its author in the MATLAB CENTRAL website. The code of those functions is not included here since the package is distributed precompiled and no source code is given.

D.1.1 multisignalplot.m

```
function multisignalplot(ecg,art,pap,cvp,fnn,w,fs)

% multisignalplot(ecg,art,pap,cvp,fnn,w,fs)
% Performs multiplot of 4 input arrays "arbitrarily" called
% ecg, art, pap, cvp.
% Optional input arguments w and fs are used for selecting windowed pieces
% of the signals. The same window will of course apply to every signal.
% For help about possible values for w, type: help chopped.
% fs is the sampling frequency at which the signals were obtained.
% fnn is a string containing the name of the record.

timearray=1:length(ecg);
```

```
if nargin==7
    [ecg,pf]=chopped(ecg,fs,w);
    art=chopped(art,fs,w);
    pap=chopped(pap,fs,w);
    cvp=chopped(cvp,fs,w);
    timearray=linspace(0,pf,length(ecg));
elseif nargin==6
    error('Incorrect number of input arguments. Valid values: 4, 5 and 7.')
end

figure();
subplot(4,1,1), plot(timearray,ecg,'m'), ylabel('ECG'), grid on
if nargin>=5, title(['MGH/MF Waveform Database, Record ',fnn]);
else title('MGH/MF Waveform Database'); end
subplot(4,1,2), plot(timearray,art,'r'), ylabel('ART'), grid on
subplot(4,1,3), plot(timearray,pap,'b'), ylabel('PAP'), grid on
subplot(4,1,4), plot(timearray,cvp,'b'), ylabel('CVP'), grid on
if nargin==7, xlabel('Time (s)');
else xlabel('Samples (n)'); end
```

D.1.2 multisignalplot2.m

```
function multisignalplot2(signal,fnn,w,fs)

% multisignalplot2(signal,fnn,w,fs)
% Performs multiplot of signal input arrays rows "arbitrarily" called
% ecg1, ecg2, ecg5, art, pap, cvp, respimp, CO2.
% Optional input arguments w and fs are used for selecting windowed pieces
% of the signals. The same window will of course apply to every signal.
% For help about possible values for w, type: help chopped.
% fs is the sampling frequency at which the signals were obtained.
% fnn is a string containing the name of the record.

ecg1=signal(1,:);
ecg2=signal(2,:);
ecg5=signal(3,:);
art=signal(4,:);
pap=signal(5,:);
cvp=signal(6,:);
respimp=signal(7,:);
CO2=signal(8,:);

timearray=1:length(ecg1);

if nargin==4
    [ecg1,pf]=chopped(ecg1,fs,w);
    ecg2=chopped(ecg2,fs,w);
    ecg5=chopped(ecg5,fs,w);
    art=chopped(art,fs,w);
    pap=chopped(pap,fs,w);
    cvp=chopped(cvp,fs,w);
    respimp=chopped(respimp,fs,w);
    CO2=chopped(CO2,fs,w);
    timearray=linspace(0,pf,length(ecg1));
elseif nargin==3
    error('Incorrect number of input arguments. Valid values: 1, 2 and 4.')
end

figure();
subplot(4,2,1), plot(timearray,ecg1,'m'), ylabel('ECG lead I'), grid on
```

```
if nargin>=2, title(['MGH/MF Waveform Database, Record ',fnn]);
else title('MGH/MF Waveform Database'); end
subplot(4,2,3), plot(timearray,ecg2,'m'), ylabel('ECG lead II'), grid on
subplot(4,2,5), plot(timearray,ecg5,'m'), ylabel('ECG lead V'), grid on
subplot(4,2,7), plot(timearray,art,'r'), ylabel('ART'), grid on
if nargin==4, xlabel('Time (s)');
else xlabel('Samples (n)'); end
subplot(4,2,2), plot(timearray,pap,'b'), ylabel('PAP'), grid on
if nargin>=2, title(['MGH/MF Waveform Database, Record ',fnn]);
else title('MGH/MF Waveform Database'); end
subplot(4,2,4), plot(timearray,cvp,'b'), ylabel('CVP'), grid on
subplot(4,2,6), plot(timearray,respimp,'k'), ylabel('Resp.Imp.'), grid on
subplot(4,2,8), plot(timearray,CO2,'k'), ylabel('CO2'), grid on
if nargin==4, xlabel('Time (s)');
else xlabel('Samples (n)'); end
```

D.1.3 infoanalysis.m

```
function [entropies,rho,P,rlo,rup,mutualI,condentropies,jointentropies] =...
    infoanalysis(ecg,art,pap,cvp)

% infoanalysis(ecg,art,pap,cvp);
% Performs information analysis over 4 input arrays "arbitrarily" called
% ecg, art, pap, cvp.

entropyecg=entropy(ecg);
entropyart=entropy(art);
entropypap=entropy(pap);
entropycvp=entropy(cvp);
entropies=[entropyecg, entropyart, entropypap, entropycvp];

[rho,P,rlo,rup]=corrcoef([ecg',art',pap',cvp']);
% P is a matrix of p-values for testing the hypothesis of no correlation.
% Each p-value is the probability of getting a correlation as large as the
% observed value by random chance, when the true correlation is zero. If
% P(i,j) is small, say less than 0.05, then the correlation R(i,j) is
% significant.
% Matrices RLO and RUP, of the same size as rho, contain lower and upper
% bounds for a 95% confidence interval for each coefficient.

mutualIecgecg=mutualinfo(ecg,ecg);
mutualIartart=mutualinfo(art,art);
mutualIpappap=mutualinfo(pap,pap);
mutualIcvpcvp=mutualinfo(cvp,cvp);
mutualIecgart=mutualinfo(ecg,art);
mutualIecgpap=mutualinfo(ecg,pap);
mutualIecgcvp=mutualinfo(ecg,cvp);
mutualIartpap=mutualinfo(art,pap);
mutualIartcvp=mutualinfo(art,cvp);
mutualIpapcvp=mutualinfo(pap,cvp);
mutualI=[mutualIecgecg, mutualIecgart, mutualIecgpap, mutualIecgcvp; ...
    mutualIecgart, mutualIartart, mutualIartpap, mutualIartcvp; ...
    mutualIecgpap, mutualIartpap, mutualIpappap, mutualIpapcvp; ...
    mutualIecgcvp, mutualIartcvp, mutualIpapcvp, mutualIcvpcvp];

condecgart=condentropy(ecg,art);
condecgpap=condentropy(ecg,pap);
condecgcvp=condentropy(ecg,cvp);
condartpap=condentropy(art,pap);
condartcvp=condentropy(art,cvp);
condpapcvp=condentropy(pap,cvp);
```

```
condentropies=[0, condecgart, condecgpap, condecgcvp; ...
    condecgart, 0, condartpap, condartcvp; ...
    condecgpap, condartpap, 0, condpapcvp; ...
    condecgcvp, condartcvp, condpapcvp, 0];

jointecgart=jointentropy(ecg,art);
jointecgpap=jointentropy(ecg,pap);
jointecgcvp=jointentropy(ecg,cvp);
jointartpap=jointentropy(art,pap);
jointartcvp=jointentropy(art,cvp);
jointpapcvp=jointentropy(pap,cvp);
jointentropies=[entropyecg, jointecgart, jointecgpap, jointecgcvp; ...
    jointecgart, entropyart, jointartpap, jointartcvp; ...
    jointecgpap, jointartpap, entropypap, jointpapcvp; ...
    jointecgcvp, jointartcvp, jointpapcvp, entropycvp];
```

D.2 QRS Detection Systems

We include here four functions: firstly, the two QRS detectors that we implemented themselves: `zeelenberg.m` and `tompkins.m`; on second place, two programs for calculating the detection performance parameters. `QRSdeteval.m` compares the positions reported by one of our algorithms, given as input, with the content of the annotations file, calculating Se, PP, Sp, FPR, Q_α and MCC. `QRSdetevalaux.m` launches the detection automatically on a specified number of signals that the user will select, either employing Zeelenberg method or Tokpins', and calculates the performance parameters by calling `QRSdeteval.m`

D.2.1 zeelenberg.m

```
function [QRS,num_qrs]=zeelenberg(ecg,fs,n_signals,der)

% [QRS,num_qrs]=zeelenberg(ecg,fs,n_signals,der)
% Performs the Zeelenberg QRS complexes detection algorithm.
% User will be asked about which derivation to employ, if applicable.
%
% INPUT:    ecg: ECG data. Can be a 2-derivation matrix as most.
%           fs: sample frequency to which the data has been obtained.
%           n_signals: number of derivations contained in ecg.
%                      size(ecg,1) could also work, but the parameter is
%                      prefered.
%           der: 1 or 2. optional. Derivation over which the processing
%                will be performed.
% OUTPUT:   QRS: positions (samples domain) in which QRS complexes are
%                detected.
%           num_qrs: number of QRS complexes detected.
%

if (n_signals > 2) || (n_signals < 1)
    error('Number of derivations is not supported');
end;

% FILTERING:
% We will design our filters for a concrete sample frequency: 250 Hz.
% However the input signals can be sampled at different rates, so we will
% need to decimate or interpolate the samples so that the behaviour of the
```

```
% filters is the one we expect and require.
% REMARK: the m-function 'filter' always operates along the first
% non-singleton dimension, namely dimension 1 for column vectors and
% non-trivial matrices, and dimension 2 for row vectors. Usually it should
% receive a one-dimensional array as input argument, to avoid mistakes. If
% it receives a matrix, like, for instance, a multi-derivation ECG signal,
% it will only work along the first row.

% Resampling
% The sample frequency used in the ECG signals of the MIT DB is frequently
% fs = 360Hz (for all the "small signals" set), but we need fs2 = 250Hz
% (for our signal processing).
% y = resample(x,p,q), where fs2 = fs*p/q;
fs2=250;
ecg1=resample(ecg(1,:),fs2,fs);
ecg2=resample(ecg(2,:),fs2,fs);

% Notch Filter
% Should be designed for deleting the electrical net frequency.
% This is 50 Hz in Europe, 60 Hz in the U.S.
% This filter is desgined for attenuating the 60 Hz component, since the
% signals used in the MIT-BIH database come from records obtained
% in the U.S.
b1=[1 0 0 0 -1];
a1=1;
Gd1=2;  % Group delay introduced by this filter
% [H1,W1]=freqz(b1,a1);
y_d1_0=filter(b1,a1,ecg1(1,:)); % filtering 1st derivation
y_d2_0=filter(b1,a1,ecg2(1,:)); % filtering 2nd derivation

% Low Pass Filter
% Should be designed to delete the higher frequency information, since the
% significant power of QRS complexes occurs between 5 and 20Hz. Any higher
% component is, hence, considered noise.
b2=[1 4 6 4 1];
a2=1;
Gd2=2;  % Group delay introduced by this filter
% [H2,W2]=freqz(b2,a2);
y_d1_1=filter(b2,a2,y_d1_0); % filtering 1st derivation
y_d2_1=filter(b2,a2,y_d2_0); % filtering 2nd derivation

%%%%%%% Plotting of the signals in the different steps (can be commented)
% figure(1)
% title('DERIVATION 1')
% subplot(3,1,1),plot(ecg1(10:1000),'b'),xlabel('samples'),...
% ylabel('unfiltered ECG')
% subplot(3,1,2),plot(y_d1_0(10:1000),'b'),xlabel('samples'),...
% ylabel('notch filtered')
% subplot(3,1,3),plot(y_d1_1(10:1000),'b'),xlabel('samples'),...
% ylabel('low-pass filtered')
% figure(2)
% title('DERIVATION 2')
% subplot(3,1,1),plot(ecg2(10:1000),'g'),xlabel('samples'),...
% ylabel('unfiltered ECG')
% subplot(3,1,2),plot(y_d2_0(10:1000),'g'),xlabel('samples'),...
% ylabel('notch filtered')
% subplot(3,1,3),plot(y_d2_1(10:1000),'g'),xlabel('samples'),...
% ylabel('low-pass filtered')
%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%% Derivation election by the user (can be commented)
% Election of the derivation could be done based on formal criteria such as
```

```
% performance results, minimizing noise, etc.
if nargin == 3
    der = questdlg('Choose the best derivation to perform the detection:',...
        'QRS Complexes Detection: Zeelenberg algorithm','Derivation 1',...
        'Derivation 2','Derivation 1');
end
if strcmp(der,'Derivation 1')|strcmp(der,'1')|(der==1)
    disp('Detection performed over Derivation 1.');
    signalut=y_d1_1-round(mean(y_d1_1));
else
    disp('Detection performed over Derivation 2.');
    signalut=y_d2_1-round(mean(y_d2_1));
end
%%%%%%%%%%%%%%%%%%%%%%%%

% Restriction rules and Initialization
%threshold=0.4*max(signalut);
% Literature recommends thresholds around that value.
%utime_update=1600; % Threshold updated each 1600ms (each 1.6s, or 10
% windows).
windowlength=160;   % Window length in which to look for crossings: 160ms
sepQRS=200;         % Minimum separation between QRS complexes: 200ms
GdTOT=Gd1+Gd2;
totalsamples=length(signalut);
window=floor(windowlength*fs2/1000); % window length (in samples)
%utime=round(utime_update*fs2/1000); % Threshold updated each utime samples.
% The next lines are for initialization of the threshold value.
% Literature recommends thresholds between -21 and 21.
signalpiece=fix(totalsamples/100); % We divide the signal in 100 pieces
vecmaxs(1,1:100)=0;
for piecenr=0:99
    vecmaxs(1,piecenr+1)=max(signalut((1+piecenr*signalpiece):...
        (signalpiece+piecenr*signalpiece)));
end
threshold=0.3*mean(vecmaxs); % This is the initial value of the threshold

% DECISION RULES
crossings=find(signalut>threshold); % candidadates to be labeled as qrs.
j=1;
qrs=0*crossings;        % qrs is now a zeros array of same length as crossings.
%wcnter=1;           % window counter. each 10 windows, threshold will be
% updated.
for i=1:length(crossings)
    if crossings(i)+window<length(signalut); % this is not the last window,
        windowend=crossings(i)+window;
        aux=signalut(crossings(i):windowend); % piece of signal we'll search
        pos=find(aux>threshold);
        if pos<=1
            qrs(j)=crossings(i);
            i=i+pos;
            j=j+1;
        end
     % We implement the adaptation of threshold so that it is updated each
     % 10 windows.
%       if wcnter == 10
%           wcnter=1;
%           max1=max(aux(1:floor(length(aux)/2)));
%           max2=max(aux(floor(length(aux)/2)+1:length(aux)));
%           threshold=0.3*(max1+max2)/2;
%       end
%       wcnter=wcnter+1;
    else    % we're placed in the last window,
```

```
        aux=signalut(crossings(i):length(signalut)); % piece of signal we'
        % ll search
        pos=find(aux>threshold);
        if pos<=1
            qrs(j)=crossings(i);
            i=i+pos;
            j=j+1;
        end
    end
end

qrs=qrs(:);
finalqrs=find(qrs==0);
QRS=round(qrs(1:finalqrs-1).*fs/fs2)';  % we resample the QRS onsets to fs
dif=diff(QRS);

% for i=1:length(dif) % Implementation of refractary periods
%     if dif(i)<sepQRS*fs/1000 % this means the QRS complexes are spaced
% less than 200ms
%         QRS=[QRS(1:i),QRS(i+2:end)]; % we delete this impossible detection
%     end
% end

QRS=QRS-GdTOT; % we must perform this substraction to adapt the indexes to
% the delay introduced by the lineal filtering
num_qrs=length(QRS);
disp([num2str(num_qrs),' QRS complexes detected.']);
%disp(['out of ',num2str(length(crossings)),' candidates.']);
```

D.2.2 tompkins.m

```
function [QRS,num_qrs]=tompkins(ecg,fs,n_signals,der)

% [QRS,num_qrs]=tompkins(ecg,fs,n_signals)
% Performs the Tompkins QRS complexes detection algorithm.
% User will be asked about which derivation to employ, if applicable.
%
% INPUT:    ecg: ECG data. Can be a 2-derivation matrix as most.
%           fs: sample frequency to which the data has been obtained.
%           n_signals: number of derivations contained in ecg.
%                      size(ecg,1) could also work, but the parameter is
%                      prefered.
%           der: 1 or 2. optional. Derivation over which the processing
%                will be performed.
% OUTPUT:   QRS: positions (samples domain) in which QRS complexes are
%                detected.
%           num_qrs: number of QRS complexes detected.
%

if (n_signals > 2) || (n_signals < 1)
    error('Number of derivations is not supported');
end;

% FILTERING:
% We will design our filters for a concrete sample frequency: 250 Hz.
% However the input signals can be sampled at different rates, so we will
% need to decimate or interpolate the samples so that the behaviour of the
% filters is the one we expect and require.
% REMARK: the m-function 'filter' always operates along the first
% non-singleton dimension, namely dimension 1 for column vectors and
```

```
% non-trivial matrices, and dimension 2 for row vectors. Usually it should
% receive a one-dimensional array as input argument, to avoid mistakes. If
% it receives a matrix, like, for instance, a multi-derivation ECG signal,
% it will only work along the first row.

% Resampling
% The sample frequency used in the ECG signals of the MIT DB is frequently
% fs = 360Hz (for all the "small signals" set), but we need fs2 = 250Hz
% (for our signal processing).
% y = resample(x,p,q), where fs2 = fs*p/q;
fs2=250;
ecg1=resample(ecg(1,:),fs2,fs);
ecg2=resample(ecg(2,:),fs2,fs);

% Low Pass Filter
% Should be designed to delete the higher frequency information, since the
% significant power of QRS complexes occurs between 5 and 20Hz. Any higher
% component is, hence, considered noise.
a1=[1 -2 1];
b1=[1 0 0 0 0 0 0 -2 0 0 0 0 0 0 1];
Gd1=6;  % Group delay introduced by this filter
% [H1,W1]=freqz(b1,a1);
y_d1_0=filter(b1,a1,ecg1(1,:)); % filtering 1st derivation
y_d2_0=filter(b1,a1,ecg2(1,:)); % filtering 2nd derivation

% High Pass Filter
% Should be designed to delete the lower frequency information, such as
% that from P- and T- waves and DC.
a2=32*[1 -1];
b2=[-1, zeros(1,15), a2, zeros(1,14), 1];
Gd2=16; % Group delay introduced by this filter
% [H2,W2]=freqz(b2,a2);
y_d1_1=filter(b2,a2,y_d1_0); % filtering 1st derivation
y_d2_1=filter(b2,a2,y_d2_0); % filtering 2nd derivation

% Derivative Filter
% Designed to calculate the differences in the signal, and emphasize the
% peaks so that the peak detector can do its task in a better condition.
a3=1;
b3=0.1*[2 1 -1 -2];
Gd3=2;  % Group delay introduced by this filter
% [H3,W3]=freqz(b3,a3);
y_d1_2=filter(b3,a3,y_d1_1); % filtering 1st derivation
y_d2_2=filter(b3,a3,y_d2_1); % filtering 2nd derivation

%%%%%%% Plotting of the signals in the different steps (can be commented)
% figure(1)
% title('DERIVATION 1')
% subplot(4,1,1),plot(ecg1(10:1000),'b'),xlabel('samples'),...
% ylabel('unfiltered ECG')
% subplot(4,1,2),plot(y_d1_0(10:1000),'b'),xlabel('samples'),...
% ylabel('low-pass filtered')
% subplot(4,1,3),plot(y_d1_1(10:1000),'b'),xlabel('samples'),...
% ylabel('high-pass filtered')
% subplot(4,1,4),plot(y_d1_2(10:1000),'b'),xlabel('samples'),...
% ylabel('differenciated')
% figure(2)
% title('DERIVATION 2')
% subplot(4,1,1),plot(ecg2(10:1000),'g'),xlabel('samples'),...
% ylabel('unfiltered ECG')
% subplot(4,1,2),plot(y_d2_0(10:1000),'g'),xlabel('samples'),...
% ylabel('notch filtered')
```

```
% subplot(4,1,3),plot(y_d2_1(10:1000),'g'),xlabel('samples'),...
% ylabel('low-pass filtered')
% subplot(4,1,4),plot(y_d2_2(10:1000),'b'),xlabel('samples'),...
% ylabel('differenciated')
%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%% Derivation election by the user (can be commented)
% Election of the derivation could be done based on formal criteria such as
% performance results, minimizing noise, etc.
if nargin == 3
    der = questdlg('Choose the best derivation to perform the detection:'...
        ,'QRS Complexes Detection: Zeelenberg algorithm','Derivation 1',...
        'Derivation 2','Derivation 1');
end
if strcmp(der,'Derivation 1')|strcmp(der,'1')|(der==1)
    disp('Detection performed over Derivation 1.');
    signalut=y_d1_2-round(mean(y_d1_2));
else
    disp('Detection performed over Derivation 2.');
    signalut=y_d2_2-round(mean(y_d2_2));
end
%%%%%%%%%%%%%%%%%%%%%%%%


signalut=signalut.^2;   % 2-Powering furtherly emphatizes the peaks and
% makes them positive.


% INTEGRAL MOVING WINDOW:
for n=1:(length(signalut)-40)
    yinteg(n)=(1/40).*sum(signalut(n:40+n));
end
% yinteg(1) will contain the samples 1 to 41
% yinteg(2) will contain the samples 2 to 42, etc

% The first samples are not valid due to the group delay introduced by the
% filtering.
GdTOT=Gd1+Gd2+Gd3;
y=yinteg(1+GdTOT:end);

% Inizialization
ye=[-inf;y(:);-inf]; % We need an array with 2 very small
                     % values in the extremes.
d1=diff(ye);         % We calculate the derivative.
ld1=length(d1);
% We will search in the derivative subsequent values with altern signs
% to find the positions of the maximums, which will locate the peaks of
% the signal
% (PEAKI)
ppeaks=find(sign(d1(2:ld1)).*sign(d1(1:(ld1-1)))<0 & sign(d1(1:ld1-1))>0);
paux=diff(ppeaks);
p=0*ppeaks;             % space allocation
i=1;
for cnt=1:length(paux)
    if (paux(cnt)> 25) % con 25 funciona bien
        p(i)=ppeaks(cnt);
        i=i+1;
    end
end
firstnull=find(p==0);
p=p(1:(firstnull(1)-1)); % we remove the nulls in the end of p,
% being false QRS peaks.
```

```
p(end+1)=ppeaks(end);
totalsamples=length(y);
% The next lines are for initialization of the threshold value.
signalpiece=fix(totalsamples/100); % We divide the signal in 100 pieces
vecmaxs(1,1:100)=0;
for piecenumber=0:99
    vecmaxs(1,piecenumber+1)=max(y((1+piecenumber*signalpiece):...
        (signalpiece+piecenumber*signalpiece)));
end
threshold=0.8*mean(vecmaxs); % This is the initial value of the threshold

% Now we have to check if the peaks overcome the threshold or not:
% PEAKI-> signal peaks after integration,
% SPKI-> if PEAKI is signal peak
% NPKI-> if PEAKI is noise peak
PEAKI=y(p);
j=1;

% DECISION RULES
SPKI(1)=threshold*4;
NPKI(1)=threshold;
for i=1:length(PEAKI)
    if PEAKI(i)>threshold
        SPKI(i+1)=(0.125*PEAKI(i)) + (0.875*SPKI(i));
        thresholdI1(i)= NPKI(i) + 0.25*(SPKI(i+1)-NPKI(i));
        NPKI(i+1)=NPKI(i);
        threshold=thresholdI1(i);
        QRS(j)=p(i); % stores positive positions
        j=j+1;
    else
        NPKI(i+1)=(0.125*PEAKI(i)) + (0.875*NPKI(i));
        thresholdI1(i)= NPKI(i+1) + 0.25*(SPKI(i)-NPKI(i+1));
        SPKI(i+1)=SPKI(i);
        threshold=thresholdI1(i);
        if j>1  % we must go back, threshold was too high
            if p(i)-QRS(j-1)>360 % if after 360ms we didn't detect
               % a new complex,
               thresholdI2= 0.5*thresholdI1(i); % we go back and check with
               % thresholdI2
               i=find(p==QRS(j-1))-1;
               threshold=thresholdI2;
            end
        end
    end
end

%figure, plot(1:length(y),y,'r',p(1:length(thresholdI1)),...
%           thresholdI1,'g',QRS,y(QRS),'+b')
QRS=QRS+GdTOT;
num_qrs=length(QRS);
disp([num2str(num_qrs),' QRS complexes detected.']);
```

D.2.3 QRSdeteval.m

```
function [pos,globalpar,numbers,performance]=QRSdeteval(QRS,fnn,fpp)

% [pos,globalpar,percentages,performance]=QRSdeteval(QRS,fnn,fpp)
% Calculates detection performance results.
%
% INPUTS: QRS = sequence of positions where QRS complexes were found by a
```

```
%               certain algorithm.
%         fnn and fpp = name and folder name of the file which contains the
%                       record in question.
%
% OUTPUTS: pos = Atrreader([fpp,fnnc]); sequence of positions where QRS
%                complexes are located according to the record annotations.
%          globalpar = [DetectedBeats,AnnotatedBeats,N];
%          numbers = [TP,FP,TN,FN];
%          performance = [100*[Se,PP,Sp,FPR,Q_alpha],MCC];

tolerance = 15;
% number of samples to be tolerated in order to accept detection and
% annotation as location-matched.

% With these lines we remove the '.DAT' string from the variable fnn and we
% store the content in fnnc.
auxstring = fliplr(fnn);
auxstring = auxstring(5:end);
fnnc = fliplr(auxstring);
% Example: fnn = 103yflwhatever103.DAT; fnnc = 103yflwhatever103;

pos = Atrreader([fpp,fnnc]);
% Now, in pos we have the positions of QRS complexes according to
% annotation files, and in QRS we have the positions detected by a custom
% algorithm. We must compare the arrays, giving a tolerance.
DetectedBeats = length(QRS);
AnnotatedBeats = length(pos);

auxarrayALL = sort([pos,QRS]);
% It contains all positions, reported either by the annotations or by the
% custom algorithm.
auxdiff=diff(auxarrayALL);
% Any element which is > tolerance indicates that next element must be kept
% in the final sorting.
auxcorr=[auxarrayALL(auxdiff>tolerance),auxarrayALL(end)];
% It's a correction of auxarrayALL in which only distinct elements are
% kept.

% Now we produce a corrected version of the QRS array: QRSc
QRSc=QRS;
for i=1:DetectedBeats
    tolvalues = QRS(i)-tolerance:QRS(i)+tolerance;
    j=1; found=0;
    while j<=(2*tolerance+1) && found==0 %length(tolvalues)
       found = sum(tolvalues(j) == auxcorr);
       j = j+1;
    end
    if found == 1
        QRSc(i) = auxcorr(find(auxcorr == tolvalues(j-1)));
    end
end
% Now we produce a corrected version of the pos array: posc
posc=pos;
for i=1:AnnotatedBeats
    tolvalues = pos(i)-tolerance:pos(i)+tolerance;
    j=1; found=0;
    while j<=(2*tolerance+1) && found==0 %length(tolvalues)
       found = sum(tolvalues(j) == auxcorr);
       j = j+1;
    end
    if found == 1
        posc(i) = auxcorr(find(auxcorr == tolvalues(j-1)));
```

```
    end
end

N = length(auxcorr); % N = TP + TN + FP + FN
Nc = N + abs(N-round(auxcorr(end)/mode(diff(auxcorr))));
% corrected N, according to the average heart rate...
% this might be tuned.
TP = sum(ismember(posc,QRSc));
FP = DetectedBeats - TP;
FN = AnnotatedBeats - TP;
TN = Nc - TP - FP - FN;
%TN = N - TP - FP - FN;
numbers = [TP,FP,TN,FN];
globalpar = [DetectedBeats,AnnotatedBeats,N];

Se = TP/(TP+FN); % Sensitivity, Hit Rate.
PP = TP/(TP+FP); % Positive Predictivity (often called Specificity, too).
FPR = FP/(FP+TN); % False Alarm Rate, or False Positive Rate.
Sp = TN/(FP+TN); % Specifity, Sensitivity of the negative category.
% Correction for infinite case:
Se(isnan(Se)==1) = 1;
PP(isnan(PP)==1) = 1;
FPR(isnan(FPR)==1) = 0;
Sp(isnan(Sp)==1) = 1;

Q_alpha = mean([Se,Sp]); % Q_alpha.
% Matthews Correlation Coefficient:
MCC = (TP*TN-FP*FN)/(sqrt((TP+FP)*(TP+FN)*(TN+FP)*(TN+FN)));
MCC(isnan(MCC)==1) = 0;
performance = [100*[Se,PP,Sp,FPR,Q_alpha],MCC];
```

D.2.4 QRSdetevalaux.m

```
function [recs,mimat] = QRSdetevalaux(fpp,howmany,flag)

% [recs,mimat] = QRSdetevalaux(fpp,howmany,flag)
% Launches QRS detection and performance evaluation over a set of signals.
%
% INPUTS: fpp = folder name in which record files are stored.
%         howmany = number of signals that will be analyzed.
%         flag = if its value is 'z', Zeelenberg algorithm will be used. If
%                it is 't', Tompkins algorithm will be used.
%
% OUTPUTS: recs = column array containing the names of processed records.
%          mimat = matrix containing performance measures. Each row
%                  corresponds to a record (an element of recs). Columns
%                  are: [DetectedBeats,AnnotatedBeats,Se,PP,Q_alpha,MCC].

mimat = zeros(howmany,8);
%recs = zeros(howmany,7);

if flag == 'z'
    disp('Zeelenberg algorithm is being used');
    for i=1:howmany
        [ecg,fs,n_signals,fnn,fpp]=Leedat(fpp);
        QRS=zeelenberg(ecg,fs,n_signals,1);
        [pos,globalpar,numbers,performance]=QRSdeteval(QRS,fnn,fpp);
        mimat(i,:) = [globalpar(1:2),performance];
        recs(i,:) = fnn;
    end
elseif flag == 't'
```

```
    disp('Tompkins algorithm is being used');
    for i=1:howmany
        [ecg,fs,n_signals,fnn,fpp]=Leedat(fpp);
        QRS=tompkins(chopped(ecg,fs,'1min'),fs,n_signals,1);
        [pos,globalpar,numbers,performance]=QRSdeteval(QRS,fnn,fpp);
        mimat(i,:) = [globalpar(1:2),performance];
        recs(i,:) = fnn;
    end
else
    error('flag input argument must be either "z" or "t"');
end
% Now we introduce the name of the records in the results matrix:
aux=fliplr(recs);
if length(aux) == 7         % fnn is of the type '100.dat'
    recs=str2num(fliplr(aux(:,5:end)));
elseif length(aux) == 12    % fnn is of the type '100small.dat'
    recs=str2num(fliplr(aux(:,10:end)));
end
mimat = [recs, mimat];
%latex(mimat,'nomath') % displays the matrix in LaTeX tabular format.
```

D.3 Beats Classifier

For the beats classifiers we include the two methods described in Section 4.5: `beatclassifier.m` and `beatclassifiermod.m`. They both share the common principle of dataset scanning, segmenting, and template correlation for decision if beats should be sorted out together. The former, besides, plots the templates morphological analysis, which is useful for studying templates similarity or dissimilarity.

D.3.1 beatclassifier.m

```
function [template,template_nom,class] =...
    beatclassifier(ecg,fs,n_signals,pos,type,threshold);

% [template,template_nom,class] = ...
%        beatclassifier(ecg,fs,n_signals,pos,type,threshold);
% Classifies beats based on templates correlation.
%
% INPUT:    ecg: ECG data. Can be a 2-derivation matrix as most.
%           fs: sample frequency to which the data has been obtained.
%           n_signals: number of derivations contained in ecg.
%                      size(ecg,1) could also work, but the parameter is
%                      prefered.
%           pos: positions of the QRS complexes according to the MIT
%                annotation. length(pos)=length(type).
%           type: type of beats according to the MIT annotation (coded).
%           threshold: optional. Threshold for the correlation coefficient
%                      comparison. If not specified, 0.9699 is assumed.
%
% OUPUT:    template: matrix containing a QRS template in each row. Its
%                     dimension is: N x L
%           template_nom: MIT's coded name associated to each different
%                         template.
%           class: array containing the type of beat, expressed as an index
```

```
%                   of the template matrix. Length: Q (equal to the number
%                   of QRS complexes).
%
% Example:
%
% template = [ t11 t12 t13 t14;
%              t21 t22 t23 t24;
%              t31 t32 t33 t34; ]
% template_nom = [ 'LBBB' 'NORMAL' 'RBBB' ];
% class    = [ 1 2 1 1 1 2 1 3 1 1 1 1 3 3 3 3 3 3 3 3 3 3 ];
%
% Input arguments pos and type can be obtained as:
% >> [pos,type] = atrreader('smallsignals\102small') % for instance.
%

% Parameters. Adjusted for better performance.
if nargin < 6
    threshold=0.9699;
end

% Resampling
fs2=250;
%ecg1=resample(ecg(1,:),fs2,fs);
%ecg2=resample(ecg(2,:),fs2,fs);
ecg1=ecg(1,:);
%ecg2=ecg(2,:);

% High-pass filtering to delete baseline drift.
fpnorm1=1/(fs2/2);
fsnorm1=1.5/(fs2/2);
[N1,Wn1] = buttord(fpnorm1,fsnorm1,3,20);
[B1,A1] = butter(N1,Wn1,'high');
% Wn = 1 means half the sample rate
%[H1,W1]=freqz(B1,A1);

% Low-pass filtering to delete other noises
fpnorm2=15/(fs2/2);
fsnorm2=25/(fs2/2);
[N2,Wn2] = buttord(fpnorm2,fsnorm2,3,20);
[B2,A2] = butter(N2,Wn2,'low');
% Wn = 1 means half the sample rate
%[H2,W2]=freqz(B2,A2);

% Filtering
y_der1=filtfilt(B1,A1,ecg1);     % first derivation
%y_der2=filtfilt(B1,A1,ecg2);     % second derivation
der1=filtfilt(B2,A2,y_der1);   % first derivation
%der2=filtfilt(B2,A2,y_der2);   % second derivation

der=der1(:)';   % for ensuring it's a row-array

% We will use windows of 140ms, 35 samples if fsample is 250Hz,
% 15 samples before the R wave, and 20 after it.

% Filling with zeros to avoid length problems
if pos(1)<15 % problematic case
    fillingBEG=15-pos(1);
    der=[zeros(1,fillingBEG) der];
    pos=pos+fillingBEG;
    pos_last=pos(end);
    isend=length(der(pos_last:end));
    fillingEND=35-isend;
```

```
    der=[der zeros(1,fillingEND)];
else
    pos_last=pos(end);
    isend=length(der(pos_last:end));
    fillingEND=21-isend;
    der=[der zeros(1,fillingEND)];
end

% Inicialization
init=pos(1)-14;     % first sample of the window
final=pos(1)+20;    % last sample of the window
template(1,:)=der(1,init:final);
template_nom(1)=type(1);
class(1)=1;
% First beat is automatically stored as a reference template.

for i=2:length(pos)
    ini1=pos(i)-14;
    final1=pos(i)+20;
    [rows,cols]=size(template);
    %plot([ini1 final1],der(1,[ini1 final1]),'*')
    %plot([ini1 final1],der(1,[ini1 final1]),'*')
    for j=1:rows % we search along the templates, calculating the
        % correlation for each complex
        A=corrcoef(template(j,:),der(1,ini1:final1));
        correl(j)=abs(A(1,2));  % correlation array of the current QRS
        % complex with every template available so far.
    end
    maximum=max(correl);
    if abs(maximum) < threshold % New template must be created
        template(rows+1,:)=der(1,ini1:final1);
%       ij=[i j]
        template_nom(rows+1)=type(i);   % Uncomment for debugging
        class(i)=rows+1;
%       disp('New template created')    % For debugging purposes.
    else    % current template must be updated by averaging with this beat
        k=find(correl==maximum);
        template(k,:)=((template(k,:))+(der(1,ini1:final1)))/2;
        aux=size(template);
%       ij=[i j]
        template_nom(k)=type(i);        % Uncomment for debugging
        class(i)=k;
%       disp(['Template ',num2str(k),' updated'])  % For debugging purposes.
    end
end

% Displaying results
[rows,cols]=size(template); % This could have been changed in the very
% last beat...
unik=unique(type);
disp([ ]);
disp(['Classification according to MIT annotations:']);
for j=1:length(unik)
    disp(['  Beats of type: ',unik(j),': ',...
        num2str(sum(strcmp(type,unik(j))))]);
end
disp([]);
disp(['Results of classification:']); [aux1,aux2]=sortarray(class);
if rows <=4
    xx=2; yy=2;
elseif rows <=6
    xx=3; yy=2;
```

```
elseif rows<=9
    xx=3; yy=3;
elseif rows<=12
    xx=4; yy=3;
elseif rows<=16
    xx=4; yy=4;
elseif rows<=20;
    xx=5; yy=4;
elseif rows<=25
    xx=5; yy=5;
elseif rows<=30
    xx=6; yy=5;
elseif rows<=36;
    xx=6; yy=6;
else xx=0; yy=0;
end
% Warning: the length of the window for the template plotting might have to
% be tuned for different signals, depending on pos(1) and pos(end), in
% order not to exceed the dimensions... (this could be solved by
% zero-filling or by tuning automatically as a function of those values).
figure;
for j=1:rows
    disp(['  Beats of type: ',template_nom(j),': ',...
        num2str(length(find(j==class)))]);
    if xx~=0
        subplot(xx,yy,j), hold on, ylabel([template_nom(j), num2str(j)])
        mipos=pos(find(class==j));
        for rr=1:aux2(j)
            % aux2(j) is the number of repetitions of each template.
            % same as length(find(j==class))
            plot(der(1,mipos(rr)-10:mipos(rr)+50),'r'),
        end
        plot(mean(der(1,mipos-10:mipos+50),1),'m')
    end
end
```

D.3.2 beatclassifiermod.m

```
function [template,template_nom,class] = beatclassifiermod(ecg,fs,...
    n_signals,pos,type,threshold,signbeats,criticalthreshold);

% [template,template_nom,class] = beatclassifiermod(ecg,fs,...
%           n_signals,pos,type,threshold,signbeats,criticalthreshold);
% Classifies beats based on templates correlation.
%
% INPUT:    ecg: ECG data. Can be a 2-derivation matrix as most.
%           fs: sample frequency to which the data has been obtained.
%           n_signals: number of derivations contained in ecg.
%                      size(ecg,1) could also work, but the parameter is
%                      prefered.
%           pos: positions of the QRS complexes according to the MIT
%                annotation. length(pos)=length(type).
%           type: type of beats according to the MIT annotation (coded).
%           threshold: optional. Threshold for the correlation coefficient
%                      comparison. If not specified, 0.9699 is assumed.
%           signbeats: optional. Number of significant beats demanded for
%                      accepting a template. If not met, reclassification
%                      will be retried lowering the threshold. If not
%                      specified, 5 is assumed.
%           criticalthreshold: optional. Critical threshold for which the
%                              reclassification will be retried. If not
```

```
%                              specified, 0.8 is assumed.
%
% OUPUT:    template: matrix containing a QRS template in each row.
%                     Dimension: N x L
%           template_nom: MIT's coded name associated to each different
%                         template.
%           class: array containing the type of beat, expressed as an index
%                  of the template matrix. Length: Q (equal to the number
%                  o QRS complexes).
%
% Example:
%
% template = [ t11 t12 t13 t14;
%              t21 t22 t23 t24;
%              t31 t32 t33 t34; ]
% template_nom = [ 'LBBB' 'NORMAL' 'RBBB' ];
% class    = [ 1 2 1 1 1 2 1 3 1 1 1 1 3 3 3 3 3 3 3 3 3 3 ];
%
% Input arguments pos and type can be obtained as:
% >> [pos,type] = atrreader('smallsignals\102small') % for instance.
%

% Parameters. Adjusted for better performance.
if nargin < 8
    criticalthreshold=0.8;
    if nargin < 7
        signbeats = 5;
        if nargin < 6
            threshold=0.9699;
        end
    end
end

% Resampling
fs2=250;
%ecg1=resample(ecg(1,:),fs2,fs);
%ecg2=resample(ecg(2,:),fs2,fs);
ecg1=ecg(1,:);
%ecg2=ecg(2,:);

% High-pass filtering to delete baseline drift.
fpnorm1=1/(fs2/2);
fsnorm1=1.5/(fs2/2);
[N1,Wn1] = buttord(fpnorm1,fsnorm1,3,20);
[B1,A1] = butter(N1,Wn1,'high');
% Wn = 1 means half the sample rate
%[H1,W1]=freqz(B1,A1);

% Low-pass filtering to delete other noises
fpnorm2=15/(fs2/2);
fsnorm2=25/(fs2/2);
[N2,Wn2] = buttord(fpnorm2,fsnorm2,3,20);
[B2,A2] = butter(N2,Wn2,'low');
% Wn = 1 means half the sample rate
%[H2,W2]=freqz(B2,A2);

% Filtering
y_der1=filtfilt(B1,A1,ecg1);     % first derivation
%y_der2=filtfilt(B1,A1,ecg2);     % second derivation
der1=filtfilt(B2,A2,y_der1);   % first derivation
%der2=filtfilt(B2,A2,y_der2);   % second derivation
```

```
der=der1(:)';   % for ensuring it's a row-array

% We will use windows of 140ms, 35 samples if fsample is 250Hz,
% 15 samples before the R wave, and 20 after it.

% Filling with zeros to avoid length problems
% pos_last=pos(end);
% isend=length(der(pos_last:end));
% filling=20-isend;
% der=[der zeros(1,filling)];
if pos(1)<15 % problematic case
    fillingBEG=15-pos(1);
    der=[zeros(1,fillingBEG) der];
    pos=pos+fillingBEG;
    pos_last=pos(end);
    isend=length(der(pos_last:end));
    fillingEND=35-isend;
    der=[der zeros(1,fillingEND)];
else
    pos_last=pos(end);
    isend=length(der(pos_last:end));
    fillingEND=21-isend;
    der=[der zeros(1,fillingEND)];
end

% Inicialization
init=pos(1)-14;     % first sample of the window
final=pos(1)+20;    % last sample of the window
template(1,:)=der(1,init:final);
template_nom(1)=type(1);
template_weight(1)=1;
class(1)=1;
% First beat is automatically stored as a reference template.
retry=1;

while retry==1
    for i=2:length(pos)
        ini1=pos(i)-14;
        final1=pos(i)+20;
        [rows,cols]=size(template);
        %plot([ini1 final1],der(1,[ini1 final1]),'*')
        %plot([ini1 final1],der(1,[ini1 final1]),'*')
        for j=1:rows % we search along the templates, calculating the
            % correlation for each complex
%           ij=[i j] % ij = [Number of beat, Number of template] that we
            % are comparing. For debugging purposes.
            A=corrcoef(template(j,:),der(1,ini1:final1));
            correl(j)=abs(A(1,2));  % correlation array of the current QRS
            % complex with every template available so far.
            % correl takes temporarily corrupt values. should only be taken
            % into account when the for j=1:rows loop is completed.

        end
        maximum=max(correl);
        if abs(maximum) < threshold % New template must be created
            template(rows+1,:)=der(1,ini1:final1);
            template_nom(rows+1)=type(i);
            template_weight(rows+1)=1;
            class(i)=rows+1;
%           disp('New template created')   % For debugging purposes.
        else % current template must be updated by averaging with this beat
            k=find(correl==maximum);
```

```
            template(k,:)=((template(k,:))+(der(1,ini1:final1)))/2;
            aux=size(template);
            template_nom(k)=type(i);
            template_weight(k)=template_weight(k)+1;
            class(i)=k;
%           disp(['Template ',num2str(k),' updated'])  % For debugging
            % purposes.
        end
    end

    if (sum(template_weight<signbeats) ~= 0) &&...
            (threshold > criticalthreshold)
        % this means SOME of the templates has NOT been validated by a
        % number of beats >= signbeats. Reclassification will be tried,
        % lowering the threshold, until the value of criticalthreshold.
        threshold = threshold - 0.01;
        %         init=pos(1)-14;     % first sample of the window
        %         final=pos(1)+20;    % last sample of the window
        template=der(1,init:final);
        template_nom=type(1);
        template_weight=1;
        class=1;
        correl=0;
    else retry=0; % all templates are robust enough,
        % or the threshold has reached its critical value.
        % Stop reclassification.
    end
end

% Displaying results
[rows,cols]=size(template); % This could have been changed in the very
% last beat...
unik=unique(type);
disp([ ]);
disp(['Classification according to MIT annotations:']);
for j=1:length(unik)
    disp(['  Beats of type: ',unik(j),': ',...
        num2str(sum(strcmp(type,unik(j))))]);
end
disp([]);
disp(['Results of classification:']);
for j=1:rows
    disp(['  Beats of type: ',template_nom(j),': ',...
        num2str(length(find(j==class)))]);
end
disp(['Final threshold was: ',num2str(threshold)])
```

D.4 HRV Analysis

We include here the code of function `HRV.m`, which realizes the time-domain analysis of heart rate variability and plots the HR time series with considering corrected N-N intervals.

D.4.1 HRV.m

```
function [HR,HRm,T,B,SDNN,SDANN,SDNNindex,RMSSD] =...
    HRV(pos,plotme,fnn,fs,HRcritT,HRcritB)
```

```
% [HR,HRm,T,B,SDNN,SDANN,SDNNindex,RMSSD] =...
%       HRV(pos,plotme,fnn,fs,HRcritT,HRcritB);
% Analysis of the Heart Rate Variability
% Optional plot of the HR with marked arrhythmic events, and computation of
% HRV measures.
%
% INPUT: pos: array containing the positions of the R-R onsets.
%        plotme: optional string indicating if a plot is wanted or not.
%                Possible values are 'yes' (by default) and 'no'.
%        fnn: optional string containing the name of the record employed.
%        fs: sample frequency for the ECG. If not specified, 360 Hz is
%            assumed.
%        HRcritT: critical HR above which tachycardia is detected.
%                 Optional. If not specified, 100bpm is assumed.
%        HRcritB: critical HR below which bradycardia is detected.
%                 Optional. If not specified, 50bpm is assumed.
%
% OUTPUT: HR: array containing the instantaneous heart rate values.
%         HRm: mean of HR.
%         T, B: matrixes which first and second rows contain the samples
%               and instants for which tachycardias or bradicardias are
%               detected, and which third rows contain the value of the HR
%               in such moments.
%         optional outputs containing diverse statistics:
%         SDNN; SDANN; SDNNindex; RMSSD (all given in ms !!!)

threshup = 1.2;
threshdown = 0.8;

if nargin < 6
    HRcritB = 50;  % critical HR below which bradycardia is detected.
    if nargin < 5
        HRcritT = 100; % critical HR above which tachycardia is detected.
        if nargin < 4
            fs = 360;
            if nargin < 3
                fnn = 'unspecified';
                if nargin < 2
                    plotme='yes';
                end
            end
        end
    end
end
if strcmp(plotme,'yes')==0 && strcmp(plotme,'no')==0
    error('Input argument plotme must be either ''yes'' or ''no''.');
end

HR = 60*fs./diff(pos(2:end));
% we obtain the array of instantaneous heart rate.
% HR = HR(find(HR<max(150,threshup*HRcritT) & ...
%              HR>min(20,threshdown*HRcritB)));
% With the previous line we discard the aberrant HR samples, due to
% premature beats or missed beats. Here we contemplate the elimination of
% HRs both heavily above HRcritT or heavily below HRcritB.
% Alternative option:
HR(HR>max(120,threshup*HRcritT))=max(120,threshup*HRcritT);
HR(HR<min(40,threshdown*HRcritB))=min(40,threshdown*HRcritB);
RRintervals = 60./HR;
%RRintervals = diff(pos(2:end))./fs; % Alternative option
%HR = 1./RRintervals; % Alternative option
```

```
time = pos(3:end)/fs; % we obtain the time array.
% After deleting aberrant samples in line 43, the time array must be
% recalculated. But with the option of lines 48 and 49 this is not
% necessary, since samples are not eliminated, but redefined.
HRl = length(HR);
HRm = mean(HR);
% Actually what we have now as RRintervals are the NNintervals, since the
% abnormal beats have already been removed.

% We define two matrixes, T and B, containing the samples and instants for
% which tachycardias or bradicardias are detected, and the value of the HR
% in such moments.
T = find(HR>=HRcritT);
T = [T; pos(T+2)/fs; HR(T)];
B = find(HR<=HRcritB);
B = [B; pos(B+2)/fs; HR(B)];

% Plotting
if strcmp(plotme,'yes')==1
    figure(), plot(time,HR,'m-',time,HRm*ones(1,HRl),'g-');
    legend(['Heart Rate, rec. ',fnn],['Mean HR = ',num2str(HRm)])
    % Instantaneous heart rate is plotted in magenta, mean HR in green,
    % and tachycardia and bradycardia points, if any, are marked in red
    % and blue along with the thresholds.
    if isempty(T) == 0
        hold on, plot(time,HRcritT*ones(1,HRl),'r-.');
        plot(T(2,:),HR(T(1,:)),'r*');
    end
    if isempty(B) == 0
        hold on, plot(time,HRcritB*ones(1,HRl),'b-.');
        plot(B(2,:),HR(B(1,:)),'b*');
    end
    title('Instantaneous Heart Rate');
    xlabel('Time (s)'); hold off;
%    grid on;
end

% Computation of statistics
% SDNN is the standard deviation of the R-R period.
SDNN = std(RRintervals);
% SDANN is the SDNN calculated over 5-min periods.
dursampl5min = 5*60*fs;
if pos(end)>=dursampl5min
    nm5minter = floor(pos(end)/dursampl5min);
    SDANN = zeros(1,nm5minter);
    aux2=find(pos>=dursampl5min);
    for i=1:nm5minter-1
        aux1=aux2;
        aux2=find(pos>=(i+1)*dursampl5min);
        SDANN(i) = std(RRintervals(aux1(1):aux2(1)));
    end
    SDANN(end+1) = std(RRintervals(aux2(1):end));
    % SDNNindex is the mean SDANN over 24 hours.
    % 24 hours equals 288 times 5 minutes.
    if pos(end)>=288*dursampl5min
        SDNNindex = mean(SDANN);
    else SDNNindex = 'not possible to calculate. record is too short';
    end
else
    SDANN = 'not possible to calculate. record is too short';
    SDNNindex = 'not possible to calculate. record is too short';
end
```

```
% RMSDD is the root mean squared deviation difference...
RMSSD = sqrt(mean((diff(RRintervals)).^2));
% HRV triangular index
% not implemented yet.

% Normalization so that the stats are given in ms instead of seconds:
SDNN = 1000*SDNN;
if ischar(SDANN)==0
    SDANN = 1000*SDANN;
end
if ischar(SDNNindex)==0
    SDNNindex = 1000*SDNNindex;
end
RMSSD = 1000*RMSSD;
```

D.5 DPCM Compression

Several functions employed for the DPCM tests are included in this Section. The main function is `DPCMexp.m`, which performs the experiments of DPCM compression, and calls `dpcm_encoder.m` and `\dpcm_decoder` and plots their results, which in turn employ the quantization scheme listed in `quantizer_encoder.m` and `quantizer_decoder.m`. See also auxiliary functions `dbc.m` and `bdc.m` in Section D.6.

Lastly, `compquotient.m` performs the calculation of entropies, kurtosis, PRD, SNR and CQ on the different signals, having to use for it many of the previously introduced functions. It also displays the plots shown in the end of Section 6.6.

D.5.1 DPCMexp.m

```
function [xrec,errormeasures]=DPCMexp(xorg,fs,bitnr,lpclen,fnn)

% [xrec,errormeasures]=DPCMexp(xorg,fs,bitnr,lpclen,fnn);
% Performs DPCM experiments, including codification and decodification,
% plotting the difference between the original and the reconstructed signal
% and calculating several distorsion measures.
% Quantization is done with a uniform quantizer.
%
% INPUT: xorg: original signal to be compressed.
%        fs: sampling frequency to which xorg was obtained.
%        bitnr: number of bits desired for the quantization.
%        lpclen: length of the linear prediction coefficients array.
%        fnn: string containing the name of the record, for plotting
%             purpose.
%
% OUTPUT: xrec: reconstructed signal after compression - decompression.
%         errormeasures: array containing several distorsion measures:
%                        [PRD; rms; SNR; MAX]


if exist('lpclen')== 0 lpclen = 5; end % by default
xorg=xorg./max(abs(xorg))/2; % normalization
% Caution! Normalization may affect error measures.
```

```
[eq,binstream,lpccoeffs] = dpcm_encoder(xorg,lpclen,bitnr);
[xrec] = dpcm_decoder(binstream,lpccoeffs,bitnr);

%tap=100;
%cf=.15;
%aux=fir1(tap,cf);
%Sa=conv2(xrec,aux,'same');

error=xorg-xrec;
PRD=sqrt(sum(error.^2)/sum((xorg-mean(xorg)).^2))*100;
rms=sqrt(sum(error.^2)/length(xorg));
SNR=10*log(sum((xorg-mean(xorg)).^2)/sum(error.^2));
MAX=max(abs(error));
errormeasures = [PRD; rms; SNR; MAX];

figure();
xlabel('samples');
subplot(3,1,1), plot(xorg,'r'); grid on; %xlabel('samples');
ylabel('Original Signal'); title(['DPCM compression: Record ',fnn]);
subplot(3,1,2), plot(xrec,'m'); grid on; %xlabel('samples');
ylabel('Reconstructed Signal');
subplot(3,1,3), plot(error,'b--'); grid on; xlabel('samples');
ylabel('Error');
%subplot(4,1,4), plot(Sa,'m'); grid on; xlabel('samples');
%ylabel('LPF output');
```

D.5.2 dpcm_encoder.m

```
function [eq,binstream,lpccoeffs] = dpcm_encoder(xorg,lpclen,bitnr)

N = length(xorg);
e = zeros(1,N); % preallocation for speed
eq = e;         % preallocation for speed
xtil = e;       % preallocation for speed

e(1) = xorg(1);
[eq(1),binstream(1,:)] = quantizer_encoder(e(1),bitnr);
%[eq(1),binstream(1,:)] = pcm_quan_enco(e(1),bitnr);
xtil(1) = eq(1);
% For the first sample there is no prediction available.

for i=2:N
   if i<=lpclen
      e(i) = xorg(i)-xtil(i-1);
      % error signal
      [eq(i),binstream(i,:)] = quantizer_encoder(e(i),bitnr);
      % quantized error signal
      xtil(i) = xtil(i-1) + eq(i);
      % predicted signal
   else
      [a,E]=lpc(xtil(i-lpclen:i-1),lpclen);
      % coefficients of the linear predictor
      a = a*E;
      m = 1:lpclen;
      j=2:lpclen + 1;
      xtilL = sum(a(j).*xtil(i-m));
      lpccoeffs(i,:)=a;
      e(i) = xorg(i)-xtilL;
      [eq(i),binstream(i,:)] = quantizer_encoder(e(i),bitnr);
      xtil(i) = xtilL + eq(i);
```

```
   end
end
binstream=binstream';
binstream=binstream(:)';
```

D.5.3 dpcm_decoder.m

```
function [xtil] = dpcm_decoder(binstream,lpccoeffs,bitnr)

[jj,size_lpc] = size(lpccoeffs);
eq = quantizer_decoder(binstream,bitnr);
xtil=cumsum(eq(1:size_lpc-1));

m=1:size_lpc-1;
j=2:size_lpc;
for i=size_lpc:length(eq)
   xtilL = sum(lpccoeffs(i,j).*xtil(i-m));
   xtil(i) = eq(i) + xtilL ;
end
```

D.5.4 quantizer_encoder.m

```
function [eq,binstream] = quantizer_encoder(e,bitnr)

% Quantizing with bitnr bits, we obtain M = 2^bitnr quantization levels.
% Signal to be quantized must be in the interval [-1,1], so it should be
% normalized.
% The quantization step will be: delta = D = 2^(-bitnr)

%e = e./max(abs(e));     % normalization

if bitnr<4
    Lgthe = length(e);      % loop counter. though inefficient...
    eq = zeros(1,Lgthe);    % preallocation for speed,
    %b = zeros(bitnr,Lgthe); % preallocation for speed,
    D = 2^(-bitnr);         % quantization step.

    switch bitnr
        case 1 % Quantization with 1 bit, 2 levels.
            for i=1:Lgthe
                if e(i) < 0
                    eq(i) = -(1/2)*D;
                    b(i,:) = [ 0 ];
                elseif e(i) >= 0
                    eq(i) = (1/2)*D;
                    b(i,:) = [ 1 ];
                end
            end
            b=b';
            binstream=b(:)';
        case 2 % Quantization with 2 bits, 4 levels.
            for i=1:Lgthe
                if e(i) < -D
                    eq(i) = -(3/2)*D;
                    b(i,:) = [ 0 0 ];
                elseif e(i) >= -D && e(i) < 0
                    eq(i) = -(1/2)*D;
                    b(i,:) = [ 0 1 ];
                elseif e(i) >= 0 && e(i) < D
```

```
            eq(i) = (1/2)*D;
            b(i,:) = [ 1 0 ];
        elseif e(i) >= D
            eq(i) = (3/2)*D;
            b(i,:) = [ 1 0 ];
        end
    end
    b=b';
    binstream=b(:)';
case 3 % Quantization with 3 bits, 8 levels.
    for i=1:Lgthe
        if e(i) < -3*D
            eq(i) = -(7/2)*D;
            b(i,:) = [ 0 0 0 ];
        elseif e(i) >= -3*D && e(i) < -2*D
            eq(i) = -(5/2)*D;
            b(i,:) = [ 0 0 1 ];
        elseif e(i) >= -2*D && e(i) < -D
            eq(i) = -(3/2)*D;
            b(i,:) = [ 0 1 0 ];
        elseif e(i) >= -D && e(i) < 0
            eq(i) = -(1/2)*D;
            b(i,:) = [ 0 1 1 ];
        elseif e(i) >= 0 && e(i) < D
            eq(i) = (1/2)*D;
            b(i,:) = [ 1 0 0 ];
        elseif e(i) >= D && e(i) < 2*D
            eq(i) = (3/2)*D;
            b(i,:) = [ 1 0 1 ];
        elseif e(i) >= 2*D && e(i) < 3*D
            eq(i) = (5/2)*D;
            b(i,:) = [ 1 1 0 ];
        elseif e(i) >= 3*D
            eq(i) = (7/2)*D;
            b(i,:) = [ 1 1 1 ];
        end
    end
    b=b';
    binstream=b(:)';
    %     case 4 % Quantization with 4 bits, 16 levels.
    %         for i=1:Lgthe
    %             if e(i) < -7*D
    %                 eq(i) = -(15/2)*D;
    %                 b(i,:) = [ 0 0 0 0 ];
    %             elseif e(i) >= -7*D && e(i) < -6*D
    %                 eq(i) = -(13/2)*D;
    %                 b(i,:) = [ 0 0 0 1 ];
    %             elseif e(i) >= -6*D && e(i) < -5*D
    %                 eq(i) = -(11/2)*D;
    %                 b(i,:) = [ 0 0 1 0 ];
    %             elseif e(i) >= -5*D && e(i) < -4*D
    %                 eq(i) = -(9/2)*D;
    %                 b(i,:) = [ 0 0 1 1 ];
    %             elseif e(i) >= -4*D && e(i) < -3*D
    %                 eq(i) = -(7/2)*D;
    %                 b(i,:) = [ 0 1 0 0 ];
    %             elseif e(i) >= -3*D && e(i) < -2*D
    %                 eq(i) = -(5/2)*D;
    %                 b(i,:) = [ 0 1 0 1 ];
    %             elseif e(i) >= -2*D && e(i) < -D
    %                 eq(i) = -(3/2)*D;
    %                 b(i,:) = [ 0 1 1 0 ];
```

```
            %             elseif e(i) >= -D && e(i) < 0
            %                 eq(i) = -(1/2)*D;
            %                 b(i,:) = [ 0 1 1 1 ];
            %             elseif e(i) >= 0 && e(i) < D
            %                 eq(i) = (1/2)*D;
            %                 b(i,:) = [ 1 0 0 0 ];
            %             elseif e(i) >= D && e(i) < 2*D
            %                 eq(i) = (3/2)*D;
            %                 b(i,:) = [ 1 0 0 1 ];
            %             elseif e(i) >= 2*D && e(i) < 3*D
            %                 eq(i) = (5/2)*D;
            %                 b(i,:) = [ 1 0 1 0 ];
            %             elseif e(i) >= 3*D && e(i) < 4*D
            %                 eq(i) = (7/2)*D;
            %                 b(i,:) = [ 1 0 1 1 ];
            %             elseif e(i) >= 4*D && e(i) < 5*D
            %                 eq(i) = (9/2)*D;
            %                 b(i,:) = [ 1 1 0 0 ];
            %             elseif e(i) >= 5*D && e(i) < 6*D
            %                 eq(i) = (11/2)*D;
            %                 b(i,:) = [ 1 1 0 1 ];
            %             elseif e(i) >= 6*D && e(i) < 7*D
            %                 eq(i) = (13/2)*D;
            %                 b(i,:) = [ 1 1 1 0 ];
            %             elseif e(i) >= 7*D
            %                 eq(i) = (15/2)*D;
            %                 b(i,:) = [ 1 1 1 1 ];
            %             end
            %         end
            %         b=b';
            %         binstream=b(:)';
    end
else % Quantization with 5 or more bits
    % We will use the subroutines dbc and bdc.
    [b0,b,bb] = dbc(e,bitnr);
    [eq] = bdc(b0,b);
    binstream = [b0 b];
end
```

D.5.5 quantizer_decoder.m

```
function [eq,binstream] = quantizer_encoder(e,bitnr)

% Quantizing with bitnr bits, we obtain M = 2^bitnr quantization levels.
% Signal to be quantized must be in the interval [-1,1], so it should be
% normalized.
% The quantization step will be: delta = D = 2^(-bitnr)

%e = e./max(abs(e));     % normalization

if bitnr<4
    Lgthe = length(e);      % loop counter. though inefficient...
    eq = zeros(1,Lgthe);    % preallocation for speed,
    %b = zeros(bitnr,Lgthe); % preallocation for speed,
    D = 2^(-bitnr);         % quantization step.

    switch bitnr
        case 1 % Quantization with 1 bit, 2 levels.
            for i=1:Lgthe
                if e(i) < 0
                    eq(i) = -(1/2)*D;
```

```
            b(i,:) = [ 0 ];
        elseif e(i) >= 0
            eq(i) = (1/2)*D;
            b(i,:) = [ 1 ];
        end
    end
    b=b';
    binstream=b(:)';
case 2 % Quantization with 2 bits, 4 levels.
    for i=1:Lgthe
        if e(i) < -D
            eq(i) = -(3/2)*D;
            b(i,:) = [ 0 0 ];
        elseif e(i) >= -D && e(i) < 0
            eq(i) = -(1/2)*D;
            b(i,:) = [ 0 1 ];
        elseif e(i) >= 0 && e(i) < D
            eq(i) = (1/2)*D;
            b(i,:) = [ 1 0 ];
        elseif e(i) >= D
            eq(i) = (3/2)*D;
            b(i,:) = [ 1 0 ];
        end
    end
    b=b';
    binstream=b(:)';
case 3 % Quantization with 3 bits, 8 levels.
    for i=1:Lgthe
        if e(i) < -3*D
            eq(i) = -(7/2)*D;
            b(i,:) = [ 0 0 0 ];
        elseif e(i) >= -3*D && e(i) < -2*D
            eq(i) = -(5/2)*D;
            b(i,:) = [ 0 0 1 ];
        elseif e(i) >= -2*D && e(i) < -D
            eq(i) = -(3/2)*D;
            b(i,:) = [ 0 1 0 ];
        elseif e(i) >= -D && e(i) < 0
            eq(i) = -(1/2)*D;
            b(i,:) = [ 0 1 1 ];
        elseif e(i) >= 0 && e(i) < D
            eq(i) = (1/2)*D;
            b(i,:) = [ 1 0 0 ];
        elseif e(i) >= D && e(i) < 2*D
            eq(i) = (3/2)*D;
            b(i,:) = [ 1 0 1 ];
        elseif e(i) >= 2*D && e(i) < 3*D
            eq(i) = (5/2)*D;
            b(i,:) = [ 1 1 0 ];
        elseif e(i) >= 3*D
            eq(i) = (7/2)*D;
            b(i,:) = [ 1 1 1 ];
        end
    end
    b=b';
    binstream=b(:)';
    %     case 4 % Quantization with 4 bits, 16 levels.
    %         for i=1:Lgthe
    %             if e(i) < -7*D
    %                 eq(i) = -(15/2)*D;
    %                 b(i,:) = [ 0 0 0 0 ];
    %             elseif e(i) >= -7*D && e(i) < -6*D
```

```
            %                 eq(i) = -(13/2)*D;
            %                 b(i,:) = [ 0 0 0 1 ];
            %             elseif e(i) >= -6*D && e(i) < -5*D
            %                 eq(i) = -(11/2)*D;
            %                 b(i,:) = [ 0 0 1 0 ];
            %             elseif e(i) >= -5*D && e(i) < -4*D
            %                 eq(i) = -(9/2)*D;
            %                 b(i,:) = [ 0 0 1 1 ];
            %             elseif e(i) >= -4*D && e(i) < -3*D
            %                 eq(i) = -(7/2)*D;
            %                 b(i,:) = [ 0 1 0 0 ];
            %             elseif e(i) >= -3*D && e(i) < -2*D
            %                 eq(i) = -(5/2)*D;
            %                 b(i,:) = [ 0 1 0 1 ];
            %             elseif e(i) >= -2*D && e(i) < -D
            %                 eq(i) = -(3/2)*D;
            %                 b(i,:) = [ 0 1 1 0 ];
            %             elseif e(i) >= -D && e(i) < 0
            %                 eq(i) = -(1/2)*D;
            %                 b(i,:) = [ 0 1 1 1 ];
            %             elseif e(i) >= 0 && e(i) < D
            %                 eq(i) = (1/2)*D;
            %                 b(i,:) = [ 1 0 0 0 ];
            %             elseif e(i) >= D && e(i) < 2*D
            %                 eq(i) = (3/2)*D;
            %                 b(i,:) = [ 1 0 0 1 ];
            %             elseif e(i) >= 2*D && e(i) < 3*D
            %                 eq(i) = (5/2)*D;
            %                 b(i,:) = [ 1 0 1 0 ];
            %             elseif e(i) >= 3*D && e(i) < 4*D
            %                 eq(i) = (7/2)*D;
            %                 b(i,:) = [ 1 0 1 1 ];
            %             elseif e(i) >= 4*D && e(i) < 5*D
            %                 eq(i) = (9/2)*D;
            %                 b(i,:) = [ 1 1 0 0 ];
            %             elseif e(i) >= 5*D && e(i) < 6*D
            %                 eq(i) = (11/2)*D;
            %                 b(i,:) = [ 1 1 0 1 ];
            %             elseif e(i) >= 6*D && e(i) < 7*D
            %                 eq(i) = (13/2)*D;
            %                 b(i,:) = [ 1 1 1 0 ];
            %             elseif e(i) >= 7*D
            %                 eq(i) = (15/2)*D;
            %                 b(i,:) = [ 1 1 1 1 ];
            %             end
            %         end
            %         b=b';
            %         binstream=b(:)';
    end
else % Quantization with 5 or more bits
    % We will use the subroutines dbc and bdc.
    [b0,b,bb] = dbc(e,bitnr);
    [eq] = bdc(b0,b);
    binstream = [b0 b];
end
```

D.5.6 compquotient.m

```
function [labels,entropies,kurtosix,PRDs,SNRs,CQ] =...
    compquotient(ecg,fs,n_signals,fnn,bitnr,plotme,pch)
```

```
% [labels,entropies,kurtosix,PRDs,SNRs,CQ] =
%         compquotient(ecg,fs,n_signals,fnn,bitnr,plotme,pch)

disp(' ');
disp('Wait please... Calculations can take some seconds. Be patient...');
disp(' ');
if nargin == 7
    ecg = chopped(ecg,fs,pch);
else
    pch = '15min'; % for multisignalplot in line 46
end

QRS = zeelenberg(ecg(1:2,:),fs,2,1);
% QRS contains the positions of QRS waves detected by Zeelenberg algorithm.
[iHR,HRm] = HRV(QRS,plotme,fnn,fs,160,40);
disp(['The average HR is: ', num2str(HRm),' bpm.']);
% iHR contains the instantaneous HR, while HRm is the mean (avg.) HR.
entropyecg = entropy(ecg(1,:));
entropyiHR = entropy(iHR);
entropies = [entropyecg, entropyiHR];
% entropies array
kurtecg = kurtosis(ecg(1,:))-3;
kurtiHR = kurtosis(iHR)-3;
kurtosix = [kurtecg, kurtiHR];
% excess kurtosis array
labels = ['ECG'; 'iHR'];

if n_signals == 8
    % Records from MGH/MF Waveform Database
    art = ecg(4,:);
    pap = ecg(5,:);
    cvp = ecg(6,:);
    entropyart = entropy(art);
    entropypap = entropy(pap);
    entropycvp = entropy(cvp);
    kurtart = kurtosis(art)-3;
    kurtpap = kurtosis(pap)-3;
    kurtcvp = kurtosis(cvp)-3;
    entropies = [entropies, entropyart, entropypap, entropycvp];
    kurtosix = [kurtosix, kurtart, kurtpap, kurtcvp];
    labels = [labels; 'ART'; 'PAP'; 'CVP'];
    if strcmp(plotme,'yes')==1
        multisignalplot(ecg(1,:),art,pap,cvp,fnn,pch,fs);
    end
end

ecg = chopped(ecg,fs,'10s');
%iHR = chopped(iHR,fs,'10s'); % Must not be chopped since the QRS positions
% series is much shorter than the ecg signal.
[ecgrec,ecgerrormeasures] = DPCMexp(ecg(1,:),fs,bitnr,2,fnn,'no');
[iHRrec,iHRerrormeasures] = DPCMexp(iHR,fs,bitnr,2,fnn,'no');
% errormeasures contains [PRD(%); rms; SNR(dB); MAX]
PRDecg = ecgerrormeasures(1);
PRDiHR = iHRerrormeasures(1);
PRDs = [PRDecg, PRDiHR];
% PRDs array
SNRecg = ecgerrormeasures(3);
SNRiHR = iHRerrormeasures(3);
SNRs = [SNRecg, SNRiHR];
% SNRs array

if n_signals == 8
```

```
    % Records from MGH/MF Waveform Database
    art = chopped(art,fs,'10s');
    pap = chopped(pap,fs,'10s');
    cvp = chopped(cvp,fs,'10s');
    [artrec,arterrormeasures] = DPCMexp(art,fs,bitnr,2,fnn,'no');
    [paprec,paperrormeasures] = DPCMexp(pap,fs,bitnr,2,fnn,'no');
    [cvprec,cvperrormeasures] = DPCMexp(cvp,fs,bitnr,2,fnn,'no');
    PRDart = arterrormeasures(1);
    SNRart = arterrormeasures(3);
    PRDpap = paperrormeasures(1);
    SNRpap = paperrormeasures(3);
    PRDcvp = cvperrormeasures(1);
    SNRcvp = cvperrormeasures(3);
    PRDs = [PRDs, PRDart, PRDpap, PRDcvp];
    SNRs = [SNRs, SNRart, SNRpap, SNRcvp];
    if strcmp(plotme,'yes')==1
        multisignalplot(ecg(1,:),art,pap,cvp,fnn,'10s',fs);
    end
end

CQ = 10./entropies./PRDs;
if strcmp(plotme,'yes')==1
    figure,
    subplot(2,2,1), plot(entropies,PRDs,'b-',entropies,PRDs,'r*'),
    xlabel('Signal entropy'), title('PRD'),
    subplot(2,2,2), plot(entropies,CQ,'m-',entropies,CQ,'r*');
    xlabel('Signal entropy'), title('Compression Quotient'),
    subplot(2,2,3), plot(kurtosix,PRDs,'b-',kurtosix,PRDs,'r*');
    xlabel('Kurtosis'), title('PRD'),
    subplot(2,2,4), plot(kurtosix,CQ,'m-',kurtosix,CQ,'r*');
    xlabel('Kurtosis'), title('Compression Quotient'),
end
```

D.6 Auxiliary Functions

In this Section several auxuliary m-functions with diverse purposes are listed.

Function `chopped.m` is used to select a windowed portion of a signal.
Function `sortarray.m` is employed to list and sort the distinct elements of an input array omitting redundancies.
Function `see_filter.m` calculates the impulse response and the frequency response of a filter, plotting them and calculating if the filter is stable or not. It was used to produce the plots shown in Section 6.2.
Function `miboxplot.m` displays boxplot of input data samples.
Functions `dbc.m` and `bdc.m` perform respectively decimal-to-binary and binary-to-decimal conversion. They are credited to their author, Sabri Gurbuz, from Drexel University.

D.6.1 chopped.m

```
function w = chopped(x,fs,p)

% w = chopped(x,fs,p);
% Calculates a chopped portion (window) of the input signal.
%
```

```
% INPUT: x = sequence to chop
%        fs = sampling frequency of x.
%        p = piece to select. it can be given in minutes, or as a string
%            which values can be: '1s', '10s', '30s', '60s', '1min', '5min',
%            '15min', '30min', '60min', '1h'.
%
% OUTPUT: w = selected portion of the original signal.

if ischar(p)==1
    if strcmp(p,'1s')==1, pf = 1;
    elseif strcmp(p,'5s')==1, pf = 5;
    elseif strcmp(p,'10s')==1, pf = 10;
    elseif strcmp(p,'30s')==1, pf = 30;
    elseif strcmp(p,'60s')==1 || strcmp(p,'1min')==1, pf = 60;
    elseif strcmp(p,'5min')==1, pf = 5*60;
    elseif strcmp(p,'15min')==1, pf = 15*60;
    elseif strcmp(p,'30min')==1, pf = 30*60;
    elseif strcmp(p,'60min')==1 || strcmp(p,'1h'), pf = 3600;
    else error('incorrect p value. Type: help chopped');
    end
else pf=60*p;
end

windowlength = fs*pf;
if windowlength<= length(x)
    disp(['A window of ',num2str(windowlength),...
            ' samples has been selected.']);
    window = 1:windowlength; % One sample is odd, but it's ok.
    % This could be improved giving the user the possibility of
    % not choosing only the first "p" piece of the signal, but
    % being able to choose...
    w = x(window);
else
    error(['incorrect p value. demanded window is too large.'...
            'Type: help chopped']);
    w=x;
end
```

D.6.2 sortarray.m

```
function [X,Y]=sortarray(x,s)

% [X,Y]=sortarray(x,s)
% Sorts the array given as an input.
%
% INPUTS: x: array to be sorted.
%         s: optional. must be = 1 if elements are strings or chars.
%
% OUPUTS: X: array containing the sorted elements of x
%         Y: array containing the number of repetitions of each element of
%            X found in x.
%
% Examples:
% >> vec = [20 4 5 10 35 20.4 30 5 15 30 20];
% >> VEC = sortarray(vec);
% Z = [X',Y'] =
%     4.0000    1
%     5.0000    2
%    10.0000    1
%    15.0000    1
%    20.0000    2
```

```
%    20.4000    1
%    30.0000    2
%    35.0000    1
%
% >> [pos,type]=atrreader('102');
% >> [X,Y]=sortarray(type,1)
% X =
%     'NORMAL'    'PACE'    'PFUS'    'PVC'    'RC(N'    'RC(P'
% Y =
%     99        2028          56           4           2           3
%

if nargin<2
    s=0;
end
X=unique(x);
Y=zeros(1,length(X));
if s==0
    disp('Numeric array')
    for j=1:length(X)
        Y(j)=sum(x==X(j));
    end
    size(X)
    size(Y)
    Z = [X',Y'] % useful for displaying in columns
elseif s==1
    disp('String array')
    for j=1:length(X)
        Y(j)=sum(strcmp(x,X(j)));
    end
end
```

D.6.3 see_filter.m

```
function [h,H,W,Gd] = see_filter(B,A,fs);
%
% [h,H,W,Gd] = see_filter(B,A,fs)
%
% Calculates the impulse response and the frequency response of a filter,
% plotting them and stating if the filter is stable or not.
% INPUT:  B,A: coefficients of the polinomials describing the filter.
%         fs: sample frequency for the filter. It's optional. If not
%         specified, fs = 250Hz will be assumed.
% OUTPUT: h, impulse response of the filter defined by B and A,
%         H, frequency response of the filter defined by B and A,
%         W, vector of digital frequencies, from 0 to pi, for which the
%         response is calculated, and plotted.
%         Gd: group delay of the filter.
%
% Remember: H(z) = Y(z)/X(z), where B are the coefficients for Y(z) and
% A are the ones for X(z).
% Remember: the point where W=pi corresponds to f/fsample=0.5.

if nargin == 2
    fs = 250;
end;
disp(['fs = ',num2str(fs),'Hz.']); % For confirming sample frequency.
h = impz(B,A);
[H,W] = freqz(B,A);
Gd = grpdelay(B,A);
subplot(2,2,1), plot(W*fs/(2*pi),20*log10(abs(H)),'r'), grid on,...
```

```
    title('Freq. Response, Amplitude (dB)'), xlabel('Frequency (Hertz)');
subplot(2,2,2), plot(W*fs/(2*pi),Gd,'r'), grid on,...
    title('Group Delay'), xlabel('Frequency (Hertz)');
subplot(2,2,3), plot(W*fs/(2*pi),180*angle(H)/pi,'r'), grid on,...
    title('Freq. Response, Phase (degrees)'), xlabel('Frequency (Hertz)');
subplot(2,2,4), stem(0:1:length(h)-1,h,'r','filled'), grid on,...
    title('Impulse Response'), xlabel('Time (samples)');
if isfinite(sum(h)),
    disp('The filter is stable, its impulse response is summable.')
else disp('The filter is not stable, its impulse response diverges.')
end
shg
```

D.6.4 miboxplot.m

```
function miboxplot(matrix,labels,tit)

% miboxplot(matrix,labels,tit)
% Displays boxplot of data samples.
% Data in matrix are assumed to be presented in columns. For instance:
%   Se          PP          Sp          FPR         Q_alpha
%   94.44       97.14       66.67       33.33       80.56
%   96.97       100.00      100.00      0.00        98.49
%   34.29       54.55       97.72       2.28        66.00
%   ...         ...         ...         ...         ...
%
% labels must be a column array containing equally dimensioned labels (add
% blank spaces if they are not naturally equal). For instance:
% Se
% PP
% Sp
% FPR
% Q_alpha
%
% tit, optional, specifies the title for the plot

[rows,cols] = size(matrix);
if cols ~= size(labels,1)
    error('Incorrect number of labels, or incorrect format of labels.')
end
mataux = [];
for j=1:cols
    for i=1:rows
        labaux(i+(j-1)*rows,:) = labels(j,:);
    end
    mataux = [mataux;matrix(:,j)];
end
figure;
boxplot(mataux,labaux);
if nargin == 3
    title(tit);
end
```

D.6.5 dbc.m

```
function [b0,b,bb]=dbc(x,B);
% [b0,b,bb]=dbc(x,B)
%
% Decimal-to-Binary conversion using B+1 bit precision
```

```
%
% x  : a constant in decimal
% B  : number of the precision bit
% b0 : sign bit ( 0 represent + sign, and 1 represents - sign)
% b  : binary bits after the binary point
% bb : (B+1)st bit
% ------------------------------------------------------------------------
% Example: x=-0.778; b=4;
%          [b0,b,bb]=dbc(x,B);    % returns binary result.
%          [b]=qround(b,bb);      % rounds off the binary input, returns binary.
%          [Y]=bdc(b0,b);         % returns decimal result.
%
%      see also QROUND and BDC

%
% Sabri Gurbuz, 7-27-98
% Drexel University
% E-mail:sabrig@cbis.ece.drexel.edu
% ************************************************************************
B = B-1;
if x>1, error(' x is not normilized.'); end
if x>=0
        b0=0;           % + sign is assigned.
        z=x;

else
        b0=1;           % - sign is assigned.
        z=-x;

end

if z >= 0

    for i=1:B,
        a=2*z;

     if a>=1
        b(i)=1;
        z=a-1;
     else
        b(i)=0;
        z=a;
     end
    end

        a=2*z;        % finds B+1 th bit in the binary point part

     if a>=1
        bb=1;
      else
        bb=0;
     end

end
```

D.6.6 bdc.m

```
function [x]=bdc(b0,b);
```

```
%  [X]=BDC(b0,b)
%
%  Binary-to-Decimal  conversion
%
% b0 : sign bit ( 0 represent + sign, and 1 represents - sign)
% y  : binary bits after the binary point
% x  : a constant in decimal
% ------------------------------------------------------------------------
% Example: x=-0.778; b=4;
%          [x]=bdc(b0,b);    % returns decimal result.
%
%      see also QROUND and DBC
%

% Sabri Gurbuz, 7-27-98
% Drexel University
% E-mail:sabrig@cbis.ece.drexel.edu
% ************************************************************************

N=length(b);      % finds the bit precision, B

y=0;

for i=1:N,
y=y+b(i)*2^(-i);  % for  x >0, converts from binary to decimal
end

x=-b0+y;

if x < 0

[b0,b,bb]=dbc(x,N+1); %+1 bit is for sign
y=0;

  for i=1:N,
  y=y+b(i)*2^(-i);  % for x < 0, converts from binary to decimal
  end
end
x=-b0+y;
```

References

[1] MedicineNet.com, "Medical Dictionary MedTerms." `http://www.medterms.com/script/main/hp.asp`, Consulted 2009.

[2] Wikimedia Foundation, Inc., "Wikimedia Commons." `http://commons.wikimedia.org/wiki/Main_Page`, Consulted 2009.

[3] Kaiser Permanente Medical Care Program, The Permanente Medical Group, "Regional Cardiac Electrophysiology Service." `http://www.kpep.org/normal_heart_function/index.htm`, Consulted 2009.

[4] A. C. Guyton and J. E. Hall, *Textbook of Medical Physiology*. Elsevier Saunder, 11th ed., 2006.

[5] J. M. Ferrero Corral, *Bioelectrónica. Señales Bioeléctricas*. S. P. Universidad Politécnica de Valencia, 1994.

[6] E. Koch, *Electrocardiografía General - con un Apéndice: Directrices para la valoración de un electrocardiograma, por el Dr. Elsbeth Koch-Momm*. Espasa-Calpe, S.A., 1942.

[7] S. Tawara, *Das Reizleitungssystem des Säugetierherzens. Eine anatomische Studie über das atrioventrikuläre Bündel und die Purkinjeschen Fäden. Mit einem Vorwort von L. Aschoff*. Gustav Fischer Verlag, 1906.

[8] E. Patterson and B. J. Scherlag, "Decremental conduction in the posterior and anterior AV nodal inputs," *Journal of Interventional Cardiac Electrophysiology*, vol. 7, pp. 137–148, Oct. 2002.

[9] D. Sánchez-Quintana and S. Y. Ho, "Anatomía de los nodos cardíacos y del sistema de conducción específico auriculoventricular," *Revista Española de Cardiología*, vol. 56, pp. 1085–92, nov 2003.

[10] S. S. Barold, "Willem Einthoven and the birth of clinical electrocardiography a hundred years ago," *Cardiac Electrophysiology Review*, vol. 7, pp. 99–104, jan 2003.

[11] W. Einthoven, "Le telecardiogramme," *Arch Int Physiol*, no. 4, pp. 132–164, 1906.

[12] E. Besterman and R. Creese, "Waller – Pioneer of electrocardiography," *British Heart Journal*, no. 42, pp. 61–64, 1979.

[13] M. Rivera-Ruiz, C. Cajavilca, and J. Varon, "Einthoven's string galvanometer: The first electrocardiograph," *Texas Heart Institute Journal*, vol. 35, no. 2, pp. 174–178, 2008.

[14] G. E. Burch and N. P. DePasquale, *A History of Electrocardiography*. Year Book Medical Publishers, 1964.

[15] F. A. Schaub, *Grundriss der klinischen Elektrokardiographie*. Documenta Geigy. Wissenschaftliche Tabellen. Supplementum ; 1, 1965.

[16] C. Alvarado Serrano, *Análisis de la variabilidad de intervalos de tiempo del ECG*. PhD thesis, Instituto de Ingeniería Electrónica. Universitat Politècnica de Catalunya, Nov. 2001.

[17] J. P. Martínez Cortés, *Análisis de alternancias de onda T en la señal electrocardiográfica*. PhD thesis, Departamento de Ingeniería Electrónica y Comunicaciones. Universidad de Zaragoza, May 2005.

[18] A. R. Pérez Riera, C. Ferreira, C. Ferreira Filho, M. Ferreira, A. Meneghini, A. Hiroshi Uchida, E. Schapachnik, S. Dubner, and L. Zhang, "The enigmatic sixth wave of the electrocardiogram: The U wave," *Cardiology Journal*, no. 5, pp. 408–421, 2008.

[19] C. R. Vázquez Seisdedos, *Análisis del ECG en monitorización ambulatoria para el diagnóstico cardíaco y la predicción de muerte cardíaca súbita*. PhD thesis, D.E.E.., Universitat Politècnica de Catalunya, 2001.

[20] G. Burch and T. Winsor, *A Primer Of Electrocardiography*. Lea & Febiger, 1946.

[21] R. C. Bourge, W. T. Abraham, P. B. Adamson, M. F. Aaron, J. M. A. Jr, A. Magalski, M. R. Zile, A. L. Smith, F. W. Smart, M. A. O'Shaughnessy, M. L. Jessup, B. Sparks, D. L. Naftel, and L. W. Stevenson, "Randomized controlled trial of an implantable continuous hemodynamic monitor in patients with advanced heart failure: The COMPASS-HF Study," *Journal of the American College of Cardiology*, vol. 51, no. 11, pp. 1073 – 1079, 2008.

[22] F. Braunschweig, B. Kjellström, M. Söderhäll, N. Clyne, and C. Linde, "Dynamic changes in right ventricular pressures during haemodialysis recorded with an implantable haemodynamic monitor," *Nephrology Dialysis Transplantation*, vol. 21, no. 1, pp. 176–183, 2006.

[23] W. Kong, D. Rollins, R. Ideker, and W. Smith, "Design and initial evaluation of an implantable sonomicrometer and cw doppler flowmeter for simultaneous recordings with a multichannel telemetry system," *Biomedical Engineering, IEEE Transactions on*, vol. 52, pp. 1365–1367, July 2005.

[24] D. W. Reynolds, N. Bartelt, R. Taepke, and T. D. Bennett, "Measurement of pulmonary artery diastolic pressure from the right ventricle," *Journal of the American College of Cardiology*, vol. 25, no. 5, pp. 1176 – 1182, 1995. must-read.

[25] R. McCraty, M. Atkinson, D. Tomasino, and H. R. Center, "Science of the heart: Exploring the role of the heart in human performance - an overview of research conducted by the institute of heartmath." e-book, `http://www.heartmath.org/research/science-of-the-heart.html`, 2001.

[26] S. C. Malpas, "Neural influences on cardiovascular variability: possibilities and pitfalls," *American Journal of Physiology - Heart and Circulatory Physiology*, vol. 282, no. 1, pp. H6–20, 2002.

[27] C. Guger, G. Edlinger, R. Leeb, G. Pfurtscheller, A. Antley, M. Garau, A. Brogni, D. Friedman, and M. Slater, "Heart-rate variability and event-related ECG in virtual environments," in *Presence 2004: The 7th Annual International Workshop on Presence*, pp. 240–245, 2004.

[28] M. L. Cendrero Mateo, "Phase-rectified signal averaging for circadian analysis of atrial fibrillation," Master's thesis, Dpt. of Electrical and Information Technology, Lunds Tekniska Högskola - E.T.S.I.T., Universidad Politécnica de Valencia, 2008.

[29] J. Francés Monllor, "Diseño de un sistema de detección y seguimiento de la frecuencia principal para fibrilación auricular basado en modelos ocultos de Markov y filtrado de Kalman," Master's thesis, E.T.S.I.T., Universidad Politécnica de Valencia, 2006.

[30] C. Mora Mora, "Análisis e identificación mediante algoritmos tiempo-frecuencia de patrones de conducta de la fibrilación auricular paroxística a partir del ECG de superficie," Master's thesis, E.T.S.I.T., Universidad Politécnica de Valencia, 2003.

[31] N. Thakor, J. Webster, and W. Tompkins, "Estimation of QRS complex power spectrum for design of a QRS filter," *IEEE Trans. Biomed. Eng.*, vol. 31, no. 11, pp. 702–706, 1984.

[32] S. Ding, X. Zhu, W. Chen, and D. Wei, "Derivation of respiratory signal from single-channel ECGs based on source statistics," *International Journal of Bioelectromagnetism*, vol. 6, no. 1, 2004.

[33] G. B. Moody, R. G. Mark, A. Zoccola, and S. Mantero, "Derivation of respiratory signals from multilead ECGs," *Computers in Cardiology*, vol. 12, pp. 113–116, 1985.

[34] J. O. Wisbeck, A. K. Barros, and R. G. Ojeda, "Application of ica in the separation of breathing artifacts in ecg signals," 1998.

[35] B. Hollý and L. Spilák, *Digital Filtration of Artifacts in ECG*. Radioelektronika, 2004.

[36] M. Vargas Drechsler, *Contribución al estudio de los micropotenciales cardíacos*. PhD thesis, Instituto de Ingeniería Electrónica. Universitat Politècnica de Catalunya, July 1996.

[37] C. E. Shannon and W. Weaver, *The Mathematical Theory of Communication*. The University of Illinois Press, 1949.

[38] M. Costa, A. L. Goldberger, and P. C.-K., "Multiscale entropy analysis of biological signals," *Physical Review E*, vol. 71, 2005.

[39] J. S. Richman and J. R. Moorman, "Physiological time-series analysis using approximate entropy and sample entropy," *Am J Physiol Heart Circ Physiol*, vol. 278, no. 6, pp. H2039–2049, 2000.

[40] D. E. Lake, J. S. Richman, M. P. Griffin, and J. R. Moorman, "Sample entropy analysis of neonatal heart rate variability," *Am J Physiol Regul Integr Comp Physiol*, vol. 283, no. 3, pp. R789–797, 2002.

[41] S. M. Pincus, "Approximate entropy as a measure of system complexity," *Proc Natl Acad Sci USA*, pp. 2297–2301, march 1991.

[42] L. A. Nunes Amaral, A. L. Goldberger, P. C. Ivanov, and H. E. Stanley, "Modeling heart rate variability by stochastic feedback," 1999.

[43] C. Letellier, "Estimating the shannon entropy: Recurrence plots versus symbolic dynamics," *Physical Review Letters*, vol. 96, no. 25, 2006.

[44] A. Kaltchenko, "Algorithms for estimating information distance with application to bioinformatics and linguistics," tech. rep., Department of Physics and Computing, Wilfrid Laurier University, April 2004.

[45] H.-X. Zhang, Y.-S. Zhu, and Z.-M. Wang, "Complexity measure and complexity rate information based detection of ventricular tachycardia and fibrillation," *Medical and Biological Engineering and Computing*, vol. 38, pp. 553–557, 2000.

[46] M. Doob, "A Gentle Introduction to TEX. A Manual for Self-study." `http://ftp.udc.es/CTAN/info/gentle/gentle.pdf`, Jan. 2002.

[47] E. F. Cassiraga, *Aprendiendo LaTeX*. Ediciones VJ, 2006.

[48] T. Oetiker, H. Partl, I. Hyna, and E. Schlegl, "The Not So Short Introduction to LaTeX 2_ε. Or LaTeX 2_ε in 141 minutes"." `http://tobi.oetiker.ch/lshort/lshort.pdf`, Sept. 2008.

[49] N. Markey, "Tame the Beast. The B to X of BibTeX." `http://tug.ctan.org/tex-archive/info/bibtex/tamethebeast/ttb_en.pdf`, Oct. 2005.

[50] A. L. Goldberger, L. A. N. Amaral, L. Glass, J. M. Hausdorff, P. C. Ivanov, R. G. Mark, J. E. Mietus, G. B. Moody, C.-K. Peng, and H. E. Stanley, "PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals," *Circulation*, vol. 101, no. 23, pp. e215–e220, 2000 (June 13). Circulation Electronic Pages: `http://circ.ahajournals.org/cgi/content/full/101/23/e215`.

[51] F. Gritzali, "Towards a generalized scheme for QRS detection in ECG waveforms," *Signal Processing*, vol. 15, pp. 183–192, Sept. 1988.

[52] B.-U. Köhler, C. Hennig, and R. Orglmeister, "The principles of software QRS detection," *Engineering in Medicine and Biology Magazine, IEEE*, vol. 21, no. 1, pp. 42–57, 2002.

[53] R. Jané Campos, *Noves tècniques en l'anàlisi del senyal electrocardiogràfic: Aplicació a l'ECG d'alta resolució*. PhD thesis, Escola Tècnica Superior d'Enginyers Industrials de Barcelona. Institut de Cibernètica. Universitat Politècnica de Catalunya, June 1989.

[54] E. Skordalakis, "Syntactic ECG processing: A review," *Pattern Recognition*, vol. 19, no. 4, pp. 305 – 313, 1986.

[55] R. Poli, S. Cagnoni, and G. Valli, "Genetic design of optimum linear and non-linear QRS detectors," *Biomedical Engineering, IEEE Transactions on*, vol. 42, pp. 1137–1141, Nov 1995.

[56] L. Sörnmo, O. Pahlm, and M.-E. Nygårds, "Adaptive QRS detection: A study of performance," *Biomedical Engineering, IEEE Transactions on*, vol. BME-32, pp. 392–401, Jun 1985.

[57] R. Murray, S. Kadambe, and G. F. Boudreaux-Bartels, "Extensive analysis of a QRS detector based on the dyadic wavelet transform," in *Time-Frequency and Time-Scale Analysis, 1994., Proceedings of the IEEE-SP International Symposium on*, pp. 540–543, Oct 1994.

[58] N. Arzeno, Z.-D. Deng, and C.-S. Poon, "Analysis of first-derivative based QRS detection algorithms," *Biomedical Engineering, IEEE Transactions on*, vol. 55, pp. 478–484, Feb. 2008.

[59] M. Aboy, J. McNames, T. Thong, D. Tsunami, M. S. Ellenby, B. Goldstein, and A. Member, "An automatic beat detection algorithm for pressure signals," *IEEE Transactions on Biomedical Engineering*, october 2005.

[60] M. Okada, "A digital filter for the QRS complex detection," *Biomedical Engineering, IEEE Transactions on*, vol. BME-26, pp. 700–703, Dec 1979.

[61] F. Chiarugi, V. Sakkalis, D. Emmanouilidou, T. Krontiris, M. Varanini, and I. Tollis, "Adaptive threshold QRS detector with best channel selection based on a noise rating system," in *Computers in Cardiology, 2007*, pp. 157–160, 30 2007-Oct. 3 2007.

[62] I. I. Christov, "Real time electrocardiogram QRS detection using combined adaptive threshold," *BioMedical Engineering OnLine*, vol. 3, no. 1, p. 28, 2004.

[63] W. A. H. Engelse and C. Zeelenberg, "A single scan algorithm for QRS-detection and feature extraction," *Computers in Cardiology*, vol. 6, pp. 37–42, 1979.

[64] J. Pan and W. J. Tompkins, "A real-time QRS detection algorithm.," *IEEE Trans. Biomed. Eng.*, vol. 32, pp. 230–236, March 1985.

[65] P. Baldi, S. Brunak, Y. Chauvin, C. A. F. Andersen, and H. Nielsen, "Assessing the accuracy of prediction algorithms for classification: an overview," *Bioinformatics*, vol. 16, no. 5, pp. 412–424, 2000.

[66] J. Thomas, Lewis J., K. W. Clark, C. N. Mead, K. L. Ripley, B. F. Spenner, and J. Oliver, G. Charles, "Automated cardiac dysrhythmia analysis," *Proceedings of the IEEE*, vol. 67, pp. 1322–1337, Sep 1979.

[67] M. Zweig and G. Campbell, "Receiver-operating characteristic (ROC) plots: a fundamental evaluation tool in clinical medicine [published erratum appears in clin chem 1993 aug;39(8):1589]," *Clinical Chemistry*, vol. 39, no. 4, pp. 561–577, 1993.

[68] J. Hanley and B. McNeil, "A method of comparing the areas under receiver operating characteristic curves derived from the same cases.," *Radiology*, vol. 148, no. 3, pp. 839–843, 1983.

[69] D. Cuesta Frau, *Estudio de métodos para procesamiento y agrupación de señales electrocardiográficas.* PhD thesis, D.I.S.C.A., Universidad Politécnica de Valencia, 2001.

[70] M. Botsivaly, C. Koutsourakis, and B. Spyropoulos, "Evaluation of a new technique for the detection of ventricular fibrillation and ventricular tachycardia," in *22^{nd} Annual EMBS International Conference*, pp. 938–941, 2000.

[71] J. Millet-Roig, J. López-Soriano, A. Mocholí, R. Ruiz-Granell, and R. García-Civera, "An additional and easy algorithm based on morphological analysis of unipolar electrogram: missclassification in implantable cardioverters-defibrillators," *Computers in Cardiology*, vol. 25, pp. 561–564, 1998.

[72] M. Engin, "ECG beat classification using neuro-fuzzy network," *Pattern Recognition Letters*, vol. 25, pp. 1715–1722, 2004.

[73] M. Bahmanyar and W. Balachandran, "Beat to beat classification of long electrocardiograms using entropies and hierarchical clustering," in *2005 IEEE Engineering in Medicine and Biology 27th Annual Conference*, pp. 5579–5581, 2005.

[74] T. F. o. t. E. S. o. C. t. N. A. S. o. P. Electrophysiology, "Heart rate variability: Standards of measurement, physiological interpretation, and clinical use," *Circulation*, vol. 93, no. 5, pp. 1043–1065, 1996.

[75] J. Nolan, P. D. Batin, R. Andrews, S. J. Lindsay, P. Brooksby, M. Mullen, W. Baig, A. D. Flapan, A. Cowley, R. J. Prescott, J. M. Neilson, and K. A. Fox, "Prospective study of heart rate variability and mortality in chronic heart failure: Results of the united kingdom heart failure evaluation and assessment of risk trial (UK-Heart)," *Circulation*, vol. 98, no. 15, pp. 1510–1516, 1998.

[76] G. Zuanetti, J. M. Neilson, R. Latini, E. Santoro, A. P. Maggioni, and D. J. Ewing, "Prognostic significance of heart rate variability in post-myocardial infarction patients in the fibrinolytic era: The GISSI-2 results," *Circulation*, vol. 94, no. 3, pp. 432–436, 1996.

[77] H. V. Huikuri, T. Seppanen, M. J. Koistinen, K. J. Airaksinen, M. Ikaheimo, A. Castellanos, and R. J. Myerburg, "Abnormalities in beat-to-beat dynamics of heart rate before the spontaneous onset of life-threatening ventricular tachyarrhythmias in patients with prior myocardial infarction," *Circulation*, vol. 93, no. 10, pp. 1836–1844, 1996.

[78] M. Brennan, M. Palaniswami, and P. Kamen, "Do existing measures of poincare plot geometry reflect nonlinear features of heart rate variability?," *Biomedical Engineering, IEEE Transactions on*, vol. 48, pp. 1342–1347, Nov. 2001.

[79] J. Bigger, J. Thomas, J. L. Fleiss, R. C. Steinman, L. M. Rolnitzky, R. Kleiger, and J. Rottman, "Frequency domain measures of heart period variability and mortality after myocardial infarction," *Circulation*, vol. 85, no. 1, pp. 164–171, 1992.

[80] J. Bigger, J. Thomas, R. C. Steinman, L. M. Rolnitzky, J. L. Fleiss, P. Albrecht, and R. J. Cohen, "Power law behavior of RR-interval variability in healthy middle-aged persons, patients with recent acute myocardial infarction, and patients with heart transplants," *Circulation*, vol. 93, no. 12, pp. 2142–2151, 1996.

[81] H. V. Huikuri, T. H. Makikallio, K. E. J. Airaksinen, T. Seppanen, P. Puukka, I. J. Raiha, and L. B. Sourander, "Power-law relationship of heart rate variability as a predictor of mortality in the elderly," *Circulation*, vol. 97, no. 20, pp. 2031–2036, 1998.

[82] K. K. L. Ho, G. B. Moody, C.-K. Peng, J. E. Mietus, M. G. Larson, D. Levy, and A. L. Goldberger, "Predicting survival in heart failure case and control subjects by use of fully automated methods for deriving nonlinear and conventional indices of heart rate dynamics," *Circulation*, vol. 96, no. 3, pp. 842–848, 1997.

[83] A. L. Goldberger, L. A. N. Amaral, J. M. Hausdorff, P. C. Ivanov, C.-K. Peng, and H. E. Stanley, "Fractal dynamics in physiology: Alterations with disease and aging," *PNAS*, vol. 99, no. 90001, pp. 2466–2472, 2002.

[84] H. V. Huikuri, T. H. Makikallio, C.-K. Peng, A. L. Goldberger, U. Hintze, and M. Moller, "Fractal correlation properties of R-R interval dynamics and mortality in patients with depressed left ventricular function after an acute myocardial infarction," *Circulation*, vol. 101, no. 1, pp. 47–53, 2000.

[85] R. W. Glenny, H. T. Robertson, S. Yamashiro, and J. B. Bassingthwaighte, "Applications of fractal analysis to physiology," *J Appl Physiol*, vol. 70, no. 6, pp. 2351–2367, 1991.

[86] P. C. Ivanov, L. A. Nunes Amaral, A. L. Goldberger, S. Havlin, M. G. Rosenblum, H. E. Stanley, and Z. R. Struzik, "From 1/f noise to multifractal cascades in heartbeat dynamics," *Chaos*, vol. 11, no. 3, pp. 641–652, 2001.

[87] U. Rajendra Acharya, K. Paul Joseph, N. Kannathal, C. M. Lim, and J. S. Suri, "Heart rate variability: a review.," *Med Biol Eng Comput*, vol. 44, pp. 1031–1051, December 2006.

[88] M. C. Teich, S. B. Lowen, B. M. Jost, K. Vibe-Rheymer, and C. Heneghan, "Heart rate variability: Measures and models," in *Dynamic Analysis and Modeling* (. M. Akay (IEEE Press, New York, ed.), vol. II, pp. 159–213, 2001.

[89] S. Jalaleddine, C. Hutchens, R. Strattan, and W. Coberly, "ECG data compression techniques–a unified approach," *Biomedical Engineering, IEEE Transactions on*, vol. 37, pp. 329–343, April 1990.

[90] R. McCaughern, A. Rosie, and F. Monds, "Two methods of asynchronous data compression," *Electronics Letters*, vol. 4, pp. 363–364, 23 1968.

[91] H. Imai, N. Kimura, and Y. Yoshida, "An efficient encoding method for electrocardiography using spline functions," *Systems and Computers in Japan*, vol. 16, no. 3, pp. 85–94, 1985.

[92] M. Shridhar and N. Mohankrishnan, "Data compression techniques for electrocardiograms," *Canadian Electrical Engineering Journal*, vol. 9, no. 4, pp. 126–131, 1984.

[93] M. Ishijima, "Fundamentals of the decision of optimum factors in the ECG data compression (special section on ECG data compression)," *IEICE Transactions on Information and Systems*, vol. E76-D, pp. 1398–1403, dec 1993.

[94] R. Nygaard and D. Haugland, "Multiple error measures in ECG data compression," in *Norwegian Signal Processing Symposium, 1997*, pp. 134–139, May 1997.

[95] Y. Zigel, A. Cohen, and A. Katz, "A diagnostic meaningful distortion measure for ECG compression," in *19th Conf. IEEE in Israel*, pp. 117–120, 1996.

[96] Y. Zigel, A. Cohen, and A. Katz, "The weighted diagnostic distortion (WDD) measure for ECG signal compression," *IEEE Transactions on Biomedical Engineering*, vol. 47, pp. 1422–1430, nov 2000.

[97] Y. Zigel, A. Cohen, and A. Katz, "ECG signal compression using analysis by synthesis coding," *IEEE Transactions on Biomedical Engineering*, vol. 47, pp. 1308–1316, oct 2000.

[98] Y. Zigel, A. Cohen, A. Abu-Ful, A. Wagshal, and A. Katz, "Analysis by synthesis ECG signal compression," in *Computers in Cardiology 1997*, vol. 24, pp. 279–282, Sep 1997.

[99] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, pp. 1098–1101, Sept. 1952.

[100] D. A. Lelewer and D. S. Hirschberg, "Data compression," *ACM Computing Surveys*, vol. 19, pp. 261–296, 1987.

[101] N. Jayant and P. Noll, *Digital Coding of Waveforms: Principles and Applications to Speech and Video*. Prentice-Hall Signal Processing Series, 1984.

[102] D. Stewart, G. E. Dower, and O. Suranyi, "An ECG compression code," *Journal of Electrocardiology*, vol. 6, no. 2, pp. 175–176, 1973.

[103] A. M. Kim and T. A. Ramstad, "Improving the rate-distortion performance of DPCM using multirate processing with application in low-rate image coding," *IEEE transactions on signal processing*, vol. 55, pp. 4958–4968, Oct 2007.

[104] G. E. Øien and T. A. Ramstad, "On the role of Wiener filtering in quantization and DPCM," in *Proc. of IEEE Norwegian Symposium on Signal Processing*, pp. 70–72, Oct 2001.

[105] R. J. Mammone, X. Zhang, and R. P. Ramachandran, "Robust speaker recognition: A feature-based approach," *Signal Processing Magazine, IEEE*, vol. 13, pp. 58–71, Sep 1996.

[106] X. Huang, A. Acero, and H.-W. Hon, *Spoken Language Processing: A Guide to Theory, Algorithm and System Development*. Prentice Hall PTR, April 2001.

[107] A. Amann, R. Tratnig, and K. Unterkofler, "Reliability of old and new ventricular fibrillation detection algorithms for automated external defibrillators," *BioMedical Engineering OnLine*, vol. 4, no. 1, p. 60, 2005.

[108] J. Chen, F. Wang, Y. Zhang, and X. Shi, "Ecg compression using uniform scalar dead-zone quantization and conditional entropy coding," *Medical Engineering & Physics*, no. 30, pp. 523–530, 2008.

[109] R. Aguilar-Ponce, J. McNeely, A. Baker, A. Kumar, and M. Bayoumi, "Multisensor data fusion schemes for wireless sensor networks," *Application-Specific Systems, Architectures and Processors, IEEE International Conference on*, pp. 136–141, 2007.

[110] S. Kadambe and C. Daniell, "Sensor/data fusion based on value of information," in *Information Fusion, 2003. Proceedings of the Sixth International Conference of*, vol. 1, pp. 25– 32, 2003.

[111] L. J. T. Jr., K. W. Clark, C. N. Mead, K. L. Ripley, B. F. Spenner, and G. C. O. Jr., "Automated cardiac dysrhythmia analysis," *Proceedings of the IEEE*, vol. 67, pp. 1322–1337, september 1979.

[112] T. Dahan and A. Sade, "Application of ICA in Removing Artifacts from the ECG," Master's thesis, Israel Institute of Technology, 2004-2005.

[113] M. Warren Rodríguez, *Electrical impedance of normal and ischemic myocardium. Role on the genesis of ST segment changes and ventricular arrhythmias*. PhD thesis, Universitat Autònoma de Barcelona, 1999.

[114] C. S. Gutiérrez Perera, "Principios de Electrocardiografía. Introducción a la Interpretación del ECG." Slideshow presentation, 2006.

[115] S. Barro, R. Ruiz, D. Cabello, and J. Mira, "Algorithmic sequential decision-making in the frequency domain for life threatening ventricular arrhythmias and imitative artefacts: a diagnostic system," *J Biomed Eng*, vol. 11, no. 4, pp. 320–8, 1989.

[116] N. Thakor, Y. Zhu, and K. Pan, "Ventricular tachycardia and fibrillation detection by a sequential hypothesis testing algorithm," *IEEE Trans Biomed Eng*, vol. 37, no. 9, pp. 837–43, 1990.

[117] G. Ersoy and M. Holewijn, "Development of a QRS detector algorithm for ecg's measured during centrifuge training," in *Proceedings of the 18th Annual International Conference of the IEEE*, vol. 4, pp. 1390–1391 vol.4, Oct-3 Nov 1996.

[118] W. Kong, J. Huang, D. L. Rollins, R. E. Ideker, and W. M. Smith, "A semi-implantable multichannel telemetry system for continuous electrical, mechanical and hemodynamical recordings in animal cardiac research," *Physiological Measurement*, vol. 28, no. 3, pp. 249–257, 2007.

[119] C. Killingsworth, D. Ritscher, G. Walcott, D. Rollins, R. Ideker, and W. Smith, "Continuous telemetry from a chronic canine model of sudden cardiac death," *Journal of Cardiovascular Electrophysiology*, vol. 11, pp. 1333–1341(9), December 2000.

[120] R. F. Appleyard and S. A. Glantz, "Two dimensions describe left ventricular volume change during hemodynamic transients," *Am J Physiol Heart Circ Physiol*, vol. 258, no. 1, pp. H277–284, 1990.

[121] M. Visner, C. Arentzen, M. O'Connor, E. Larson, and R. Anderson, "Alterations in left ventricular three-dimensional dynamic geometry and systolic function during acute right ventricular hypertension in the conscious dog," *Circulation*, vol. 67, no. 2, pp. 353–365, 1983.

[122] A. Magalski, P. Adamson, F. Gadler, M. Böehm, D. Steinhaus, D. Reynolds, K. Vlach, C. Linde, B. Cremers, B. Sparks, and T. Bennett, "Continuous ambulatory right heart pressure measurements with an implantable hemodynamic monitor: A multicenter, 12-month follow-up study of patients with chronic heart failure," *Journal of Cardiac Failure*, vol. 8, no. 2, pp. 63 – 70, 2002. not-read.

[123] F. Braunschweig, C. Linde, M. Eriksson, C. Hofman-Bang, and L. Ryden, "Continous haemodynamic monitoring during withdrawal of diuretics in patients with congestive heart failure," *Eur Heart J*, vol. 23, no. 1, pp. 59–69, 2002. not-read.

[124] J. Leypoldt, A. Cheung, J. Delmez, J. Gassman, N. Levin, J. Lewis, J. Lewis, and M. Rocco, "Relationship between volume status and blood pressure during chronic hemodialysis.," *Kidney Int*, vol. 61, no. 1, pp. 266–75, 2002. not-read.

[125] P. B. Adamson, A. Magalski, F. Braunschweig, M. Bohm, D. Reynolds, D. Steinhaus, A. Luby, C. Linde, L. Ryden, B. Cremers, T. Takle, and T. Bennett, "Ongoing right ventricular hemodynamics in heart failure: clinical value of measurements derived from an implantable monitoring system," *J Am Coll Cardiol*, vol. 41, no. 4, pp. 565–571, 2003.

[126] C. M. van Ravenswaaij-Arts, L. A. Kollée, J. C. Hopman, G. B. Stoelinga, and H. P. van Geijn, "Heart rate variability.," *Ann Intern Med*, vol. 118, pp. 436–447, March 1993.

[127] K. Urbanowicz, J. J. Zebrowski, R. Baranowski, and J. A. Holyst, "How random is your heart beat?," Nov 2006.

[128] C.-K. Peng, S. Havlin, J. Hausdorff, J. Mietus, H. Stanley, and A. Goldberger, "Fractal mechanisms and heart rate dynamics: Long-range correlations and their breakdown with disease," *Journal of Electrocardiology*, vol. 28, no. Supplement 1, pp. 59 – 65, 1995. Research and Technology Transfer in Computerized Electrocardiology.

Index