

UNIVERSIDAD COMPLUTENSE DE MADRID

FACULTAD DE INFORMÁTICA



TESIS DOCTORAL

**Robust modeling for information acquisition in biophysical
and critical scenarios**

**Modelado robusto para la extracción de información
en entornos biofísicos y críticos**

MEMORIA PARA OPTAR AL GRADO DE DOCTOR

PRESENTADA POR

Josué Pagán Ortiz

Directores

**José Luis Ayala Rodrigo
José Luis Risco Martín
José Manuel Moya Fernández**

Madrid
Ed. electrónica 2019



Universidad Complutense de Madrid

Facultad de Informática

**Modelado Robusto para la Extracción de
Información en Entornos Biofísicos y Críticos**

**Robust Modeling for Information Acquisition in
Biophysical and Critical Scenarios**

AUTHOR:

JOSUÉ PAGÁN ORTIZ

ADVISORS:

Dr. JOSÉ LUIS AYALA RODRIGO

Dr. JOSÉ LUIS RISCO MARTÍN

Dr. JOSÉ MANUEL MOYA FERNÁNDEZ

Robust Modeling for Information Acquisition in Biophysical and Critical Scenarios

by

Josué Pagán Ortiz



Thesis submitted to the Universidad Complutense de Madrid in fulfillment of the requirements for the degree of

Doctor en Ingeniería Informática

Facultad de Informática

Advisors:

Professor Dr. José Luis Ayala Rodrigo

Professor Dr. José Luis Risco Martín

Professor Dr. José Manuel Moya Fernández

Universidad Complutense de Madrid

Madrid

Dedication

*I dedicate this thesis to my mother and my father,
to my sister and my brother,
to my niece. I dedicate this thesis to all my friends; they know who they are.
I dedicate this thesis to my supervisors
and all my teachers who have taught me since I started to learn.
I dedicate this thesis to the time I owe to all of them.*

* * *

*Dedico esta tesis a mi madre y mi padre,
a mi hermana y mi hermano,
a mi sobrina.
Dedico esta tesis a todos mis amigos; ellos saben quiénes son.
Dedico esta tesis a mis directores de tesis
y a todos mis profesores que me han enseñado desde que empecé a aprender.
Dedico esta tesis al tiempo que les debo a todos ellos.*

Acknowledgement

I would like to thank to all the people involved in this research, without which this thesis would not had been possible. I want to many thank to the clinical team Ana Gago, Mónica Sobrado, Patricia Heredia, Mercedes Gallego and José Vivancos; to the bachelor, master students and developers Irene, Kevin, Álvaro, Almudena and many others.

I want to thank also to my external advisors during my research stays in Germany and the US, Björn Eskofier and Hassan Ghasemzadeh for their time and help to improve my research skills. And, of course I want to thank to my office partners in the UCM Carlos González, Juan Carlos Salinas, Sergio Bernabé and Alberto del Barrio, and all the GreenDISC and GreenLSI members to make research times easier.

* * *

Financial support

This work was supported in part by the EU (FEDER) and the Ministerio de Economía y Competitividad de España under Research Grants TIN2015-65277-R and TEC2012-33892, and the Spanish national project FIS PI15/01976.

Declaration of authorship

The author, Josué Pagán Ortiz, hereby declares and confirms that this thesis is entirely the result of the work carried out in the Department of Architecture and Technology of Computing Systems of the School of Computer Science at the Complutense University of Madrid. This thesis contains original contribution by the author unless otherwise indicated.

Josué Pagán Ortiz,

July 20, 2018

Table of Contents

List of Tables	xvi
List of Figures	xx
Abbreviations	xxiv
Abstract	xxvi
Resumen	xxviii
1 Introduction	1
1.1 Mobile Cloud Computing and health	3
1.1.1 Ambulatory monitoring, MCC and WBSN	6
1.2 The migraine disease	10
1.2.1 Migraine Prediction	12
1.3 Energy efficiency in WBSN	14
1.3.1 Energy efficiency through computation in the node	16
1.3.2 Energy efficiency in the radio link	16
1.3.3 Holistic energy optimization	18
1.4 Publications	20
1.4.1 Journal papers	20
1.4.2 Conference papers	21

1.4.3	Other publications	22
1.4.4	Intellectual property	22
2	Architecture of the system	23
2.1	Robust predictive scenarios	24
2.2	Model of the system	26
2.2.1	Data Acquisition System	27
2.2.2	Robust Prediction System	28
2.2.2.1	Data Driver	28
2.2.2.2	Sensor Status Detector	29
2.2.2.3	Prediction System	32
2.2.3	Decider	34
2.2.4	Expert Decision System	35
2.3	Energy efficiency problem	36
3	Implementation	39
3.1	Overview	43
3.2	Sub-system I. Fine-grained modeling	46
3.2.1	Hemodynamic data acquisition	46
3.2.1.1	Overview of the monitoring experiments	49
3.2.1.2	Conditioning circuits of hemodynamic signals	51
3.2.1.3	Pain objectification	55
3.2.2	Migraine predictive modeling	57
3.2.2.1	The training and validation blocks	62
3.2.2.2	Set of models	65
3.2.3	Improvement of the prediction and decider	66
3.2.3.1	Sensor-Dependent Model Selection System	69

3.3	DEVS-based migraine predictor simulator	71
3.3.1	DEVS formalization of the conceptual model	73
3.3.1.1	Coupled models	73
3.3.1.2	Atomic models	76
3.4	Sub-system II. Prediction support	79
3.4.1	Environmental data acquisition	80
3.4.2	Migraine predictive modeling from environmental data . . .	83
3.5	Sub-system III. Prediction support	84
3.5.1	Acquisition of qualitative variables	86
3.5.2	Classification of qualitative variables	87
3.6	System implementation: CERPS	87
3.7	Energy aware prediction system	89
3.7.1	Energy aware predictive models	89
3.7.2	Compression techniques for energy saving	95
3.7.3	Workload balancing in an MCC system	97
4	Results	113
4.1	Fine-grained migraine predictive modeling	114
4.1.1	Preprocessing block	116
4.1.2	Training block	118
4.1.3	Validation block	120
4.1.3.1	Model Repair: improving predictions	123
4.1.4	Model Selection block	129
4.1.4.1	Sensor-Dependent Model Selection System	130
4.1.4.2	Advanced Sensor-Dependent Model Selection System	131
4.1.5	Test of the Sub-system I	133
4.1.6	Fine-grained modeling using GE algorithms	138

4.1.6.1	Training of models	138
4.1.6.2	Validation of models	141
4.1.6.3	Test results	144
4.2	Advanced migraine prediction simulation system	147
4.3	Sub-systems II and III. Prediction support	150
4.3.1	Sub-system II. Prediction support using environmental data	150
4.3.2	Sub-system III. Prediction support using qualitative data	151
4.4	Energy aware prediction system	153
4.4.1	Compression techniques for energy saving	153
4.4.1.1	Discrete Wavelet Transform (DWT)	154
4.4.1.2	Discrete Fourier Transform (DFT)	155
4.4.1.3	Compressed Sensing	157
4.4.2	Energy savings	159
4.4.3	Energy aware predictive models	162
4.4.3.1	The concern of the energy consumption	165
4.4.4	Workload balancing in an MCC system	166
4.4.4.1	Energy results in WBSN	166
4.4.4.2	Workload off-loading policies	168
4.4.4.3	Economic benefits	172
5	Conclusions and future work	175
	Bibliography	179
A	Modeling and simulation background	199
A.1	State-space models	199
A.2	GE algorithms	201
A.3	Time series analysis	208

A.4 DEVS simulation	213
B Experimental set-up. On-body channel transmission modeling	217
C Sizing of HPC and Cloud clusters	221
D Costs of the migraine disease in Europe	223
E Ethical consent	227

List of Tables

3.1	Data acquisition parameters.	51
3.2	Environmental data acquisition parameters.	81
3.3	Premonitory symptoms for decision making	86
3.4	Parameters for the GE modeling with energy constraints.	91
3.5	Average #clk of all branches of <i>__fixunssfsi</i>	93
3.6	The acquisition parameters in the Shimmer nodes	103
3.7	Probability of sensor data loss.	103
3.8	Data transmission properties for Shimmer nodes.	103
3.9	Parameters in the Castalia radio simulator	107
3.10	Tasks that each level of the network is able to perform.	110
3.11	Five scenarios for the workload balancing policies.	111
4.1	Training results for 40 minutes forward horizon	119
4.2	# Useful models after validation	120
4.3	SCS. Validation for models of Patient A	124
4.4	SCS. Prediction horizons for all the combinations of features	125
4.5	Results of the prediction repair methodologies	126
4.6	SCS. Robustness analysis	129
4.7	SCS. Test results	133
4.8	Generalization of models for other patients	134

4.9	CCS. Test results	137
4.10	GE. Validation results for Patient A	141
4.11	GE. Validation results for Patient B	141
4.12	Comparing GE and ZOH	144
4.13	CCS. Test results	145
4.14	Accuracy of average models using local environmental data	151
4.15	Classification of migraines with predictive premonitory symptoms	152
4.16	Best fit solutions in the Pareto fronts for Patient A.	163
4.17	Best fit solutions in the Pareto fronts for Patient B.	164
4.18	Sensing nodes energy consumption	167
4.19	Energy consumption breakdown in HPC and Cloud Data Center	170
4.20	Energy consumption for various workload off-loading scenarios	171
4.21	Total energy savings in nodes, Data Centers, and coordinators	172
4.22	Average saving for a federation of Data Centers in Europe	173
A.1	Parameters for the GE experiments.	208
A.2	Characteristics to identify pure AR and MA time series models	211
D.1	Average direct and indirect costs of migraine	223
D.2	Migraine sufferers in Europe	224
D.3	Availability of models depending on features and sensors' status	224

List of Figures

1.1	Research objectives	2
1.2	Aspects and abstraction levels treated in the thesis	3
2.1	Robust predictive paradigms in the IoT	25
2.2	Model of the system architecture	27
2.3	Robust Prediction System architecture	28
2.4	Data Driver	29
2.5	Sensor Status Detector	30
2.6	Signal Repair module	32
2.7	Prediction system	33
2.8	Decider	35
2.9	Examples of core functions for the Decider module	35
2.10	Energy and workload.	36
3.1	Equivalence graph architecture-implementation	40
3.2	Implementation of the system architecture	41
3.3	Implementation of the Sub-system I	47
3.4	Monitoring devices used to monitor hemodynamic variables	48
3.5	Patient wearing the monitoring system	49
3.6	Scheme of the monitoring study	50
3.7	Evolution of the pain marked in the smartphone	52

3.8	Hemodynamic signals: ECG and PPG	53
3.9	Hemodynamic variables after synchronization and preprocessing . .	54
3.10	Stages of the migraine	55
3.11	Modeling of subjective pain evolution curve	56
3.12	Training and validation diagram	58
3.13	Data Driver	59
3.14	Cross-validation	64
3.15	Schemes proposed to improve migraine predictions	67
3.16	Improvement of the predictions	67
3.17	<i>SDMS</i> ² design and usage for the real-time applications with the SCS	70
3.18	Conceptual system diagram	72
3.19	Representation of simulated system.	73
3.20	The <i>RootCoupled</i> model	74
3.21	Three coupled models that the <i>EFsys</i> hosts	77
3.22	Implementation of the Sub-system II	80
3.23	SensorTag CC2650 for local environmental monitoring	81
3.24	Global meteorological variables during pain episodes	82
3.25	Local environmental variables during a pain episode	82
3.26	Implementation of the Sub-system III	85
3.27	Expert Decision System implemented as a ANFIS model	88
3.28	Overview of the energy optimization workflow	90
3.29	Optimal approach of low-power migraine prediction models	90
3.30	Unwrapped branches possibilities for <i>__fixunssfsi</i>	92
3.31	Optimized data transmission in an MCC architecture	95
3.32	Energy saving through data compression	96
3.33	Workload balancing optimization in the CERPS	98

3.34	Scheme representing the structure of the system	99
3.35	Streaming and processing working modes in the sensing nodes . . .	104
3.36	Gradual inclusion of migraine sufferers in the system	110
4.1	Example of signal repair using GPML and ARX	117
4.2	N4SID. Average fits for training.	119
4.3	Validation for models from Patient A. 20 minutes forward	121
4.4	Validation for models from Patient B. 20 minutes forward	122
4.5	Validation applying the average model	124
4.6	Comparison of SCS and CSS results	127
4.7	$SDMS^2$ design and usage for the real-time applications with the SCS	130
4.8	Hierarchies of sets of models using the SCS.	130
4.9	Board of strategies for model selection	132
4.10	SCS. Test results	135
4.11	CSS. Test results	139
4.12	Training results	140
4.13	CCS with GE algorithms	144
4.14	CCS. Robustness	146
4.15	Evaluation of the basic simulator using GE models.	148
4.16	Complete simulator. Error induction.	149
4.17	ECG and PPG signals before applying transformations.	154
4.18	DWT. Results of the optimization process	156
4.19	3D decision curves for the compression of the DWT	157
4.20	DFT. Results of the optimization process	158
4.21	3D decision curves for the compression of the DFT	159
4.22	CS. Results of the optimization process	160
4.23	2D decision curves for the CS	161

4.24	Savings in transmission.	161
4.25	3D and 2D views of the Pareto Fronts	163
4.26	Energy savings: clk cycles VS sensors consumption	165
4.27	Utilization of the HPC Data Center	170
4.28	Utilization of the Cloud cluster.	171
A.1	GE population evolution	203
A.2	Mix and mutation of individuals	204
A.3	Example of a 2D Pareto front	204
A.4	BNF grammar used for migraine prediction	206
A.5	Decoding process in Grammatical Evolutionary algorithms	207
A.6	Body skin temperature (TEMP) and its first derivative	209
A.7	ACF and PACF functions for TEMP and its first derivative	211
B.1	On-body channel transmission architecture.	218
B.2	On-body channel transmission scheme	218

Abbreviations

ACF	AutoCorrelation Function
AEMET	Agencia Estatal de Meteorología
AIC	Akaike Information Criterion
ANFIS	Adaptive Neuro-Fuzzy Inference System
ANN	Artificial Neural Network
ARIMA	AutoRegressive Moving-Average integrated model
ARIMAX	AutoRegressive Moving-Average integrated model with eXogenous inputs
ARMA	AutoRegressive Moving-Average model
AR	AutoRegressive model
ARX	AutoRegressive model with eXogenous inputs
BLE	Bluetooth Low Energy
BNF	Backus Naur Form
CDF	Cohen–Daubechies–Feauveau wavelet transform
CERPS	Critical-Events Robust Prediction System
CLT	Central Limit Teorem
CPS	Cyber Physical System
CS	Compressed Sensing
CSS	Complete-model repair Strategic Study

DAS	Data Acquisition System
DEVS	Discrete Event Systems
DFT	Discrete Fourier Transform
DIST	Distributed computing
DSS	Decision Support System
DWT	Digital Wavelet Transform
ECG	Electrocardiogram
EDA	Electrodermal Activity
EDS	Expert Decision System
FFT	Fast Fourier Transform
FIFO	First In First Out
FOG	Fog computing
FP	False Positive
FPGA	Field Programmable Gate Array
GA	Genetic Algorithms
GE	Grammatical Evolutionary algorithms
GP	Genetic Programming
GPL	General Public License
GPML	Gaussian Process Machine Learning
GUI	Graphical User Interface
HIL	Hardware-in-the-loop
ICT	Information and Communications Technologies
IDFT	Inverse Discrete Fourier Transform
IDWT	Inverse Digital Wavelet Transform
IoT	Internet of Things
IT	Information Technology

LTI	Linear Time-Invariant
MA	Moving-Average model
MAX	Moving-Average model with eXogenous inputs
MBSE	Model-Based Systems Engineering
M&SBSE	Modeling and Simulation-based Systems Engineering
MCC	Mobile Cloud Computing
MDE	Model-Driven Engineering
MLP	Multilayer Perceptron
MSE	Model-Based Engineering
N4SID	Subspace State-Space System IDentification
NSGA-II	Non-dominated Sorting Genetic Algorithm II
NTC	Negative Temperature Coefficient
PACF	Partial AutoCorrelation Function
PPG	Photoplethysmography
PPV	Predictive Positive Value
PS	Prediction System
PUE	Power Usage Effectiveness
ROC	Region Of Convergence
RPS	Robust Prediction System
RSSI	Received Signal Strength Indicator
SARMA	Seasonal AutoRegressive Moving-Average model
SCS	Simple-model repair Conservative Study
SpO2	Peripheral oxygen saturation
SR	Symbolic Regression
SVM	Support Vector Machine

TP	True Positive
TPR	True Positive Rate
UML	Unified Modeling Language
VHDL	VHSIC Hardware Description Language
WSN	Wireless Sensor Network
WBSN	Wireless Body Sensor Network

Abstract

The era of information and Big Data is an environment where multiple devices, always connected, generate huge volumes of information (paradigm of the Internet of Things). This paradigm is present in different areas: the Smart Cities, sport tracking, lifestyle, or health. The goal of this thesis is the development and implementation of a Robust predictive modeling methodology using low cost wearable devices in biophysical and critical scenarios. In this manuscript we present a multilevel architecture that covers from the on-node data processing, up to the data management in Data Centers. The methodology applies energy aware optimization techniques at each level of the network. And the decision system makes use of data from different sources leading to expert decision system.

The architecture and methodology developed have been implemented in a real environment for the prediction of neurological diseases, specifically in the prediction of symptomatic migraine attacks. Migraine is one of the most disabling neurological diseases; it affects 12-15% population worldwide, and its most remarkable symptom is an intense headache. Migraine is a social illness that affects the daily life of people who suffer from it. It leads to costs of € 1200 per patient and year for public and private health systems, due to direct and indirect costs, such as visits to the emergencies and absenteeism. The prediction of the onset of pain suppose a before and after in the life of these people, being able to act in advance to avoid it.

In this thesis, it has been shown for the first time in the literature that it is possible to model and predict a symptomatic crisis of migraine. The proof of concept has been carried out with real patients in a clinical study. Preliminary results have shown that, by monitoring biomedical variables (such as skin temperature, heart rate or sweating), it is possible to predict the beginning of a crisis—within the medication action window current—with up to 47 minutes in advance with high precision, and 76% hit rate on average. For the generation of these models, it has been used state spaces and grammatical evolutionary algorithms. The methodology is robust against sensor failures—very common in

ambulatory monitoring wearable devices—and it is able of temporarily regenerating a signal when a failure of data occurs. Otherwise, it is automatically able to adapt the prediction models to use others that do not make necessitate of the damaged sensor, thus maintaining certain prediction horizon.

In this research, a real time simulator of a migraine crisis predictor has been developed, and it is reactive to sensor failures. The methodology presented contemplates the use of environmental variables, and symptomatic information of the patient for improvement and prediction system. In the economic plane, it has been glimpsed that, before a possible energy-efficient scenario in which 2% of the European migraineurs make use of the prediction system, leads savings of € 288 million in the electricity bill, and up to € 1272 million for the health entities.

Resumen

La era de la información y el Big Data, se sustenta en un entorno en el que múltiples dispositivos, siempre conectados, generan ingentes volúmenes de información (paradigma del Internet de las Cosas). Este paradigma ha llegado diversos entornos: las denominadas ciudades inteligentes, monitorización deportiva, estilo de vida, o salud. El objetivo de esta tesis es el desarrollo e implementación de una metodología de modelado predictivo robusto mediante dispositivos *wearable* de bajo coste en entornos biofísicos y críticos. A lo largo de este manuscrito se presenta una arquitectura multinivel que abarca desde el tratamiento de los datos en los dispositivos sensores hasta el manejo de éstos en centros de datos. La metodología cubre la optimización energética a todos los niveles con consciencia del estado de la red. Y el sistema de decisión hace uso de datos de distintas fuentes para conformar un sistema experto de decisión.

La arquitectura y metodología desarrolladas se han implementado en un entorno real de predicción de enfermedades neurológicas, en concreto en la predicción de crisis sintomáticas de migraña. La migraña es una de las enfermedades neurológicas más discapacitantes; afecta a entre un 12 y un 15% de la población mundial, y se caracteriza principalmente por intensos dolores de cabeza. La migraña es una enfermedad social que afecta a la vida diaria de quienes la sufren. Asimismo, supone unos costes de unos 1200 euros por paciente y año a las arcas de salud públicas y privadas debido a costes directos e indirectos, como las visitas a urgencias y el absentismo laboral. La predicción de la llegada del dolor suponer un antes y un después en la vida de estas personas, pudiendo actuar con antelación para evitarlo.

En esta tesis se ha demostrado por primera vez en la literatura que es posible el modelado y predicción de una crisis sintomática de migraña. La prueba de concepto se ha realizado con pacientes reales en un estudio clínico. Los resultados preliminares han demostrado que, mediante la monitorización de variables biomédicas (como temperatura, ritmo cardíaco o sudoración), es posible predecir la llegada de una crisis—dentro de la ventana de actuación de los medicamentos actuales—con hasta 47 minutos de antelación con alta precisión, y

un promedio de acierto del 76%. Para la generación de estos modelos se han usado algoritmos de espacios de estado y evolución gramatical. La metodología es robusta ante fallos de sensor—muy comunes en dispositivos *wearable* de monitorización ambulatoria—y es capaz de regenerar temporalmente una señal ante pérdida o fallo de datos. En caso contrario, de manera automática es capaz de adaptar los modelos de predicción a otros que no hagan uso del sensor dañado, manteniendo así cierto horizonte de predicción.

En el transcurso de esta investigación, se ha desarrollado además un simulador de predicción de crisis de migrañas en tiempo real, reactivo ante fallos de sensor. La metodología presentada contempla el uso de variables atmosféricas y sintomáticas del paciente para mejorar y ayudar al sistema de predicción. En el ámbito económico, se ha vislumbrado que, ante un posible escenario energéticamente eficiente en el que un 2% de la población migrañosa europea hace uso del sistema de predicción, se generarían ahorros energéticos de 288 millones de euros y hasta 1272 millones de euros para las entidades sanitarias.

Chapter 1

Introduction

Over the last two decades there have emerged hundred of problems derivated of the new era of the Information and Communications Technologies (ICT). Zettabytes of data are generated by billions of heterogeneous devices that acquire all kind of information from innumerable sources, leading to a new industrial era revolutionizing everything: education, food and agriculture, health, transportation *etc.* This abundance of data can lead to an exponential increase of unprecedentedly knowledge generation. To achieve this knowledge, humanity requires of big data analytic solutions. The Internet of Things (IoT) embraces in a vertical way the architectures, methodologies and elements of many different scenarios which encompass always-connected devices that can acquire, process and transmit data.

This thesis presents a robust methodology for predictive modeling and optimization applied to critical scenarios presenting symptomatic crises. The objectives of the present manuscript (see Figure 1.1) depict a framework to generate knowledge from multi-source data to predict and actuate in biophysical and critical scenarios. This is made through and application-independent data-fusion [91] predictive modeling technique and it is applied to a real study case. This study case focuses on the prediction of critical events in the migraine diseases, which is called to become a hot-topic research in near future of neurology.

The methodology presented is not constrained by data, and the reader can easily extrapolate the framework to many other specific study cases that distributed and collaborative data acquisition networks from heterogeneous data sources that may be correlated or uncorrelated. These networks, in IoT, conceive Wireless Sensor Networks (WSNs). People are currently used to use these networks in their daily life, such as smart watches. These device lack of the precision and reliability of traditional laboratory instrumentation, however they

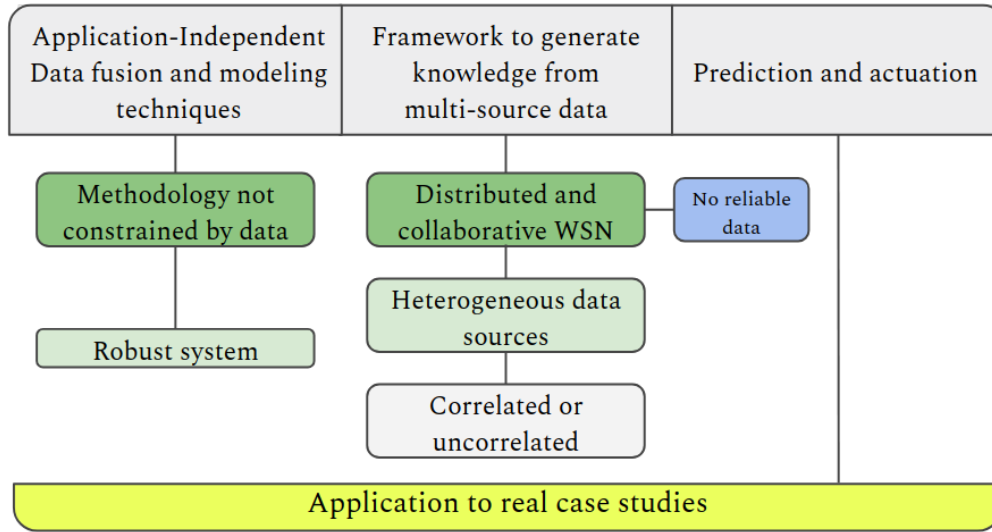


Figure 1.1: Research objectives. Robust methodology for predictive modeling and optimization applied to critical scenarios. This scheme summarizes the different issues surrounding the research objectives.

are inexhaustible sources of data. It is proposed a robust system to deal with these unreliable data, so that this problem affects minimally the decision making over the prediction of critical events.

This thesis goes through many aspects, such as (i) Model-Based Systems Engineering (MBSE), (ii) predictive modeling, (iii) simulation, (iv) energy efficiency, and (v) optimization, in different abstraction levels, from (i) personalized and ubiquitous monitoring, to (ii) local area networks, and (iii) big centralized infrastructures. These different aspects and abstraction levels are depicted in Figure 1.2. In this Figure it can be seen how there is an optimization layout underlining and giving support to each aspect of the thesis: from energy to predictive modeling. At the same time, the energetic face is also present over each level: data processing and transmission in small monitoring devices and intelligent devices, or workload balancing in an IoT scenario including federated Cloud data centers. Simulation is a key tool to better understand and smooth development of complex systems. In this research it has been used as check-point of the methodology of the robust predictive system prior to a final physical implementation. Predictive modeling covers every facet of this thesis, making use of information extraction from heterogeneous data sources with different semantic representation: analog continuous/ discrete data, partial subjective information, dichotomic information from log forms *etc.*

This Chapter serves as introduction to the problem to be solved in further Chapters, and as background support of the state of the art and current techniques

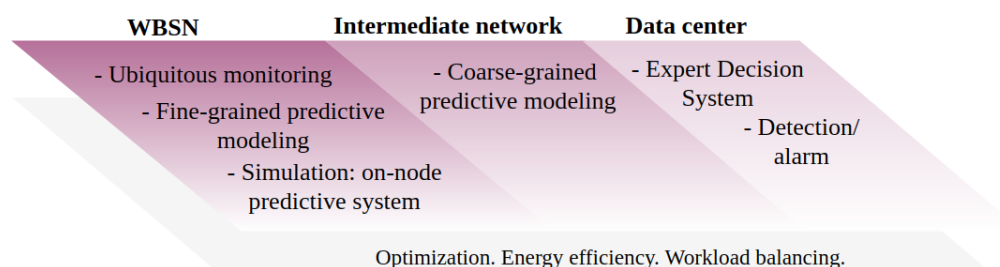


Figure 1.2: Different aspects and abstraction levels treated in the thesis: ubiquitous WBSN, intermediate local area networks, and big centralized infrastructures. There is an optimization layout underlining and giving support to each aspect: from energy to predictive modeling. The energetic optimization is present over each level: data processing and transmission in small monitoring devices and intelligent devices, and workload balancing in an IoT scenario.

existing to deal with problems of each aspect described in thesis.

1.1 Mobile Cloud Computing and health

1 Introduction

1.1 Mobile Cloud Computing and health

1.1.1 Ambulatory monitoring, MCC and WBSN

1.2 The migraine disease

1.3 Energy efficiency in Wireless Body Sensor Networks

1.4 Publications

IoT devices are always connected and generate a very large mass of data that must be processed. This thesis is framed in a Mobile Cloud Computing (MCC) environment [92]. MCC is the combination of Cloud computing, mobile computing and wireless networks to transport abundant computational resources to mobile users, network operators, as well as Cloud computing providers. The resources are sent to computer units with higher capacities (Data Centers) which are powered with high levels of energy for their operation.

Large scale population monitoring systems in MCC scenarios are starting to become a reality, specially under the paradigm of Smart Cities. Many different and vague definitions of Smart Cities are described in the literature. According to the authors of the ranking of middle-sized smart European Cities reported for the Austrian Centre of Regional Science [81], a Smart City is understood as a certain ability of a city and not focusing on single aspects and certain characteristics are

required for an evaluation. The authors of this study also indicate that the term is not used in a holistic way describing a city with certain attributes, but is used for various aspects, that Deakin *et al.* [37] describe in another study. It may be said that Smart Cities are based on the application of ICT through IoT, and they merge a set of services and resources in the urban environment to make citizen's life better. The services applied over the health care lead to the eHealth scenario that targets this research work. Training sport systems or healthcare ambulatory monitorization systems foresee a future where data and communication management—between the monitoring systems and the data bases and computing centers—will be crucial to reduce the energetic impact that these already cause in our days.

Among proposed Smart Cities services devoted to monitorization of daily activity and healthcare (*eHealth* systems), there are several research areas that exhibit a high interest for future applications. As an example, tracking prediction to optimize tactics in team sports [65], or mobile applications for sport training [96]. Further, we can find dozens of applications focused on the medical environment, such as: remote diagnosis, disease alarms generation [3], prediction of atrial fibrillation [119], or arrhythmia detection using mobile devices [104].

Health care in unobtrusive monitoring scenarios has become one of the most challenging areas of the Internet of Things. Health care applications are being exploited in both commercial [69] and research environments [131]. The distribution and specificity of clinical use of these devices remains in the category Wellness & Life Style category, trying to serve mass market. These Wellness & Life Style devices are not designed around a specific customer's need or medical condition, thus, are solely used for customer's self-tracking of variables like calories burned, exercise and sleep time, heart rate, among others. Recent proliferation of wireless monitoring devices has brought big opportunities to the industry of healthcare and personal well-being. This has become a major concern in the paradigm of proactive personal eHealth [191] in the era of the IoT.

Self-tracking device creates an opportunity among individuals who are able and willing to participate in the management of their disease. In the general context, 9% of the population use wearable to monitor health conditions and 36% of doctors think that monitor pain would help patients and their practice¹. On the healthcare side, a growing number of hospitals are using (or piloting the use of) self-tracking technologies to monitor conditions, which have to be instrumented (real time monitor), interconnected (smart phone, hubs), intelligent (provide recommendations). Reuters predicted that 70% of healthcare

¹Fashioningtech: <http://fashioningtech.com/profiles/blogs/fashioning-health-wellness-an-interview-with-misfit-wearables> (accessed March 2018)

organizations globally will invest in wearables, apps and related technologies² in the next three years. The global market for wearable medical devices is projected to reach 30 billion by 2020, with a compounded annual growth rate of 42.9% between 2014 and 2019³. Wearable device shipments worldwide will grow from 23 million in 2015 to 175 million units in 2020, and pass from 9% usage to 38% by 2020⁴.

In the market, during the last years, dozens of companies have developed a large number of monitoring devices. Most of them are not oriented to the science but mostly to the personal well-being. They usually incorporate movement sensors and rarely others such as oxygen saturation, electrodermal activity or skin surface temperature. The accuracy of these devices—generally presented as wristbands—is not enough for medical purposes in a continuous ambulatory study.

Different medical bio-monitors can be found in the state-of-the-art, such as the IMEC's devices *Smart ECG necklace* for ambulatory cardiac monitoring, or the Health Patch⁵ developed in collaboration between the Holst Centre and SHINKO⁶. Another example are the medical devices for rehabilitation and muscle strain diagnosis developed by TMG⁷. Wristband Empatica E4⁸ is a commercial device that, unlike Surge (FitBit⁹), also commercial, is thought for scientific usage. Empatica E4 integrates several sensors, but suffer from accuracy problems due to plastic belt of the wristband format makes it move. This company has recently released the Embrace¹⁰, a commercial device for seizure detection. For this massive consuming device, they have improved the movement problems with an adjustable textile belt. There are also others less portable devices, such as the well known Shimmer device¹¹. This device allows wireless communication and control of the whole acquisition system to enlarge the battery life. Shimmer nodes are thought for non-invasive biomedical research [24].

²Reuters' press note: <https://www.reuters.com/article/us-apple-hospitals-exclusive/exclusive-apples-health-tech-takes-early-lead-among-top-hospitals-idUSKBN0L90G920150205> (accessed March 2018)

³RockHeal+h: <https://rockhealth.com/reports/the-future-of-biosensing-wearables/> (accessed March 2018)

⁴eMarketer:<http://www.emarketer.com/article.aspx?R=1012756&RewroteTitle=1> (accessed March 2018)

⁵IMEC devices: <https://www.holstcentre.com/news---press/2016/health-patch/> (accessed March 2018)

⁶SHINKO: <http://www.shinko.co.jp/english/index.html> (accessed March 2018)

⁷TMG: <http://www.tmg-bodyevolution.com/medical> (accessed March 2018)

⁸Empatica: <https://www.empatica.com> (accessed March 2018)

⁹FitBit: <https://www.fitbit.com> (accessed March 2018)

¹⁰Embrace: <https://www.empatica.com/embrace/> (accessed March 2018)

¹¹Shimmer: <http://www.shimmersensing.com> (accessed March 2018)

1.1.1 Ambulatory monitoring, MCC and WBSN

The continuous development of high-performance microprocessors and novel communication protocols has stimulated great interest in the research of wireless sensor nodes for wireless body sensor network (WBSN) applications [151]. They allow physiological signals to be easily monitored.

The most common WBSN application areas are, as aforementioned, healthcare in the elderly population, remote medical diagnosis or sport training mobile applications. But there are also others such as disease alarm notifications as Alemda *et al.* relate in [3] or rehabilitation [133]. There are other regarding monitoring of illnesses related to functional impairment [161] or Parkinson disease [93], and it also plays a crucial role in Ambient Assisted Living tools designed for elders [137]. Among all these applications, the core functionality is detection and classification of activities or events such as heart failure.

The importance of promoting the use of these networks is shown in the amount of inversions made in projects like the “HealthService24 Project”, most of them in countries with elderly populations with chronic diseases [118]. Predictive models in the *eHealth* scenario using wearable monitoring devices have increased rapidly—from activity recognition [45] to event detection in neurological diseases [74]. One of the biggest areas of interest, because of its major impact on the user’s satisfaction, is the use of data collected by WBSNs for prediction in chronic diseases. Some studies, for instance, have tackled the predictive functionality for epileptic seizures [102] or anomalies in blood glucose levels. Another example of prediction system is the work of Hidalgo *et al.* in [78] for modeling of glycemia in humans.

Many different modeling techniques have been applied in these researches: time series algorithms, state-space, or Grammatical Evolutionary (GE) algorithms among others. Time series models are polynomial functions based on past data of the series. Time series models can be used to predict data by recursive iteration over past data. On the other hand, state-space models represent the behavior of a signal through the relation with system’s states and other. Alternatively to these classic techniques, GE algorithms are a heuristic approach to the model discovery problem. GE reduce the model complexity and help on the automatic feature selection automatically by means of symbolic regressions, as opposed to the time costly manual feature selection of classic modeling methods [7]. As an example, Cescon [28] presents in her thesis different modeling techniques using both state-space models and time series algorithms for prediction in diabetes. GE and Genetic Programming, thanks to the improvement of computational resources have faced to many real world applications in different areas, such as prediction of economic events in financial applications, ecological simulation [4] or modeling

of dynamic power consumption of servers [149] in data centers. Related to health problems, GE has been used for classification purposes such as fetal heart rate in [60] or to model glycemia in humans [78].

However, most of the aforementioned studies have not dealt with the issues of real-time data acquisition by WBSNs, but from off-line processed databases. These scenarios present big challenges in data acquisition and processing in ambulatory environments. Unfortunately, event detection in neurological diseases and diagnosis are still in their early stages of research and with limited commercial examples. In chronic diseases with symptomatic events the prediction of incoming events is crucial. Per-patient modeling of the behavior of chronic diseases has proved that the implementation of such kind of methodologies is applicable in a real ubiquitous health care system. Despite this thesis describes a framework and an architecture, this has been entirely implemented in a real study case of a chronic disease with symptomatic crises, the migraine, where it has been considered the real-time issue.

Remote health monitoring systems are divided into two main layers: the front-end and the back-end, to acquire, store and analyze patient data [131]. Despite there have been discussed different topology for healthcare monitoring systems [54], and there are many possible diagnostic systems and methodologies for several chronic diseases (such as diabetes [49] or asthma [55]) at the architectural level there are few real implementations working in such a fashion. There are a lot of business and research opportunities hanging on big data analytics and WBSNs, but there are still few companies that use the information both facets: predictive purposes an ambulatory monitoring, *i.e.* 24 hours a day. However, in the private plane recently there have appeared companies combining ambulatory monitoring and prediction. MedicSen¹² that predicts future glucose values, or Diagnostix¹³ for cardiovascular risks. Cardiovascular diseases have been studied from decades; from a patent in 1985 [139] up to research works [36] like WANDA [99, 164]—an end-to-end remote health monitoring and analytics system designed and implemented for heart failure in a real-time and automated way.

One of the reasons why there has not been an explosion of ambulatory predictive systems in healthcare are two of the major problems with WBSNs: (i) data loss due to disconnections, sensor problems or sensor losses and (ii) the energy constrains of these networks due to the battery life; very common problems in ambulatory studies for ambient assisted living (AAL) [6]. Data losses pose strong demands on the real-time analysis of data, but the prediction

¹²MedicSen: <https://www.mediczen.com> (accessed March 2018)

¹³Diagnostix: <https://www.diagnostix.com/cardiovascular-risk/> (accessed March 2018)

capability should not be affected. Different alternatives have been proposed in the literature to tackle the energetic problem as shown further (see Section 1.3). However, there have not been found automatic intelligent monitoring alternatives to solve the problem of data losses. During the research it has not been found either implementations that combine both intelligent energy efficient implementations and autonomous decisions systems to manage damaged data. In this thesis it has been considered these situations and it has been tackled in an optimized way.

Prediction of critical events in chronic diseases

Three are the main challenges of the intelligent algorithms in health data mining systems [14]: prediction, anomaly detection and diagnosis. For instance, support vector machines (SVM) have been used in arrhythmia and seizure detection [102]; hidden Markov models (HMM) have been applied for detecting anomalies in blood glucose levels [192]; decision trees and Bayes nets have been used in stress studies [165]; and artificial neural networks (ANN) in diabetes works [29]. Using Artificial Neural Networks (ANNs), Babušiak *et al.* [12] predict the ECG signal for arrhythmia detection. With the same purpose, Grandl *et al.* use these algorithms in WBSNs in [64]. Other work tries to predict epileptic seizures through EEG [121] using damaged data, but these have not been gathered from a WBSN.

This thesis focuses on the algorithms of prediction and correlation among signals. Both kinds of algorithms are frequently applied in the literature to static datasets, either clinical or online databases of sensor data [84, 97], where there is no data loss and signals are less noisy. Therefore, the first difficulty in this study is to deal with a noisy real scenario from which the gathered data come. Thus, the robustness will be a very important requirement of the developed data processing. Mobile health (*m-Health*) enables monitoring the type, quantity and quality of everyday activities improving daily care and the design of more clinically meaningful interventions to establish evidence-based practices [44]. Within *m-Health*, WBSNs are one of the most promising tools for unobtrusive healthcare monitoring. Wearable sensors placed on the body surface enable registering physiological and environmental human conditions, such as electroencephalogram [41], blood pressure [47] and accelerometry [31], among others.

The majority of health case studies with WBSNs in the literature are performed under a controlled setting [1, 5, 155, 165]. On the contrary, the data collected for this paper were acquired from an ambulatory monitoring scenario,

where real migraineurs wore the sensors almost 24 hours per day. Thus, this study captures the full complexity and multi-dimensionality of a real health scenario. Healthcare data volumes are massive, and as a consequence, the focus of health monitoring systems is progressively shifting from the mere sensor reading to the treatment and processing of the data [14] and the development of intelligent algorithms.

Simulation frameworks for healthcare systems

As stated before, there are not many implementations of systems for decision support in ambulatory predictive systems, to generate recommendations for patients and clinicians. There are solutions to tackle the problem of prediction in healthcare as Djitong *et al.* present in [43], however they are not oriented to personalized, and thus, ambulatory medicine but on health problems related to infections. In addition, the magnitude of these problems and the diversity of possible scenarios—such as different outbreaks distributed in time and space—lead to first developments in simulated frameworks.

Development of devices for diagnosis and detection is a time consuming process. Simulation experiments may help speed up the engineering process, especially in the initial phase of exploration. This is the main goal of the paper: to specify an advanced simulation framework that helps to validate the behavior of a migraine attack predictive system. Simulation means savings in terms of implementation costs, time, or human and material resources. Simulation is the natural step in a MBSE design prior to a physical implementation because it accelerates error debugging phase and it allows the verification of the methodology.

Concerning simulation frameworks for simulations of Cyber Physical Systems (CPSs) it can be found Ptolemy II [23], a discrete-event modeling environment focused on application to cyber-physical and embedded systems; or Simulink from MATLAB¹⁴, that is more oriented towards engineers and has hard semantics. Barhak *et al.* [16] present a software tool for chronic diseases. Despite of the software is presented as a tool for many different chronic diseases, models must be defined as states and transition probabilities of Markov transition models—which is a hard constraint. Another example is Archimedes [46], a commercial simulator for diabetes using an object-oriented approach. Both of them implement a Graphical User Interface (GUI) and are distributed as open source.

As stated before, this thesis pursues the simulation of a CPS to raise alarms

¹⁴MATLAB 2015. version 8.5.0.197613 (R2015a). Natick, Massachusetts, The MathWorks Inc.

for predictive modeling of symptomatic crises in chronic diseases, specifically, the migraine. As it will be shown, in this thesis it has been used a co-designed prototype for data collection and real time experimentation, however, this system does not incorporate an alarm interface. Before an actual device is implemented in hardware to do so, a hardware/software (HW/SW) co-simulation that includes hardware-in-the-loop (HIL) will be used. This will ensure that the system works in presence of actual hardware sensor failures and physical actuators, and triggers alarms accurately, as predicted by the simulation system. The specification of such a HW/SW co-simulation system is specified using the Discrete Event Systems (DEVS) formalism [189] that specifies unambiguous structure and behavior of any hybrid complex system. This is an incremental design with easy component substitution and rapid HW/SW swapping mechanism as previously shown through a DEVS-based transparent HW/SW modeling and simulation framework in [144]. A formal method for temporal predictive models for healthcare is proposed for depression by Breda *et al.* in [176]. In this thesis it has been implemented a DEVS-based model that will be the basis for the aforementioned HIL system. With this, we will be able to start clinical experiments to inform patients when to take medications in advance followed by a study of the benefits of prediction in terms of complete or partial pain relief.

To the best of our knowledge, this study is the first attempt to simulate a real device for the prediction of symptomatic crises. The advanced simulator presented in this manuscript simulates a robust system against sensor failures that performs error signal detection and signal recovery. In case that the sensors are not available, it executes a hierarchical methodology of predictive models selection if signal recovery is not possible.

1.2 The migraine disease

1 Introduction

1.1 Mobile Cloud Computing and health

1.2 The migraine disease

1.2.1 Migraine Prediction

1.3 Energy efficiency in Wireless Body Sensor Networks

1.4 Publications

This thesis focuses on the concrete use case of migraine prediction in a large-scale *eHealth* scenario. It will be used real data to develop predictive to warn

patients of the onset of pain.

Migraine is a chronic recurrent headache, lasting several hours, and affects around 10% worldwide. Migraine is considered one of the most disabling diseases with a great socioeconomic impact. In this scenario, migraine patients would benefit from a crisis prediction application based on WBSNs, and pharmacological prophylaxis would reveal effective, increasing the quality of life and decreasing recurring costs. However, scientific studies so far have not established the hemodynamic changes that occur before a migraine attack. Moreover, as far as the authors are concerned, the work presented here is an innovative approach in several areas of pervasive health monitoring for migraine patients, and for the first time: (i) it has been performed a continuous ambulatory monitoring of hemodynamic variables of migraineurs (ii) this research makes a multivariable study of hemodynamic changes before, during and after a migraine attack (iii) a study on the predictability of migraine attacks is presented.

In a real scenario of ambulatory monitoring of migraineurs, in this research we pose and answer the following questions: is it possible to predict a migraine? Can this prediction be launched in a specific time before the pain arrives? Does a migraine model exist? Is this model only per patient? How do these models respond under unreliable data acquisition? This research proposes to find multivariable patient-based models for the prediction of migraine attacks. These models should be robust and accurate in an ambulatory scenario, where the reliability of data acquisition cannot be ensured. This real scenario brings data collected from a WBSN with all of the real problems of noisy data, data loss and sensor disconnections.

Migraine is a primary chronic headache, with a significant hereditary component. It is characterized by recurrent headaches, unilateral or bilateral, throbbing, moderate to severe intensity and typically worsens with exercise. It may be accompanied by sensitivity to noise, light and/or odors. Sometimes, migraines are preceded or accompanied by transient neurological symptoms (visual, sensory or speech), and they are called migraines with aura. Migraine's current management and treatment is normally pharmacological and depends on the chronification level of the disease [40, 67, 77, 108, 154]. When the patients take their medication affects the efficacy of the treatment. This is usually too late to avoid the start of the headache, and the patient suffers the migraine disturbances equally.

Stovner *et al.* in [163] show that the prevalence of the migraine in European population is around the 15%. Others, such us Lipton *et al.* in [106], or the WHO [184], limit this value to the 10% worldwide. The economic consequences

of the migraine represent €1,222 per patient per year [105] in Europe, that means almost €125,000mill in the whole old continent. Most of the direct costs are due to absences from work or low job performance. The migraine is a social and economical problem.

A cascade of neurological processes precede a migraine followed by the pain for the next few hours or days. Some migraine sufferers experience symptoms that may occur from three days to hours before the pain starts [62]. These symptoms are called premonitory symptoms and they are subjective and unspecific: nausea, yawns, tearing, *etc.* Some patients also suffer from auras. Auras are objective and specific perceptual disturbances such as losing vision that occurs commonly within 30-15 minutes before the onset of pain. The most efficient way to stop this process and avoid the pain is to advance the intake of specific drugs. Therefore, the action mechanism of the medicine is able to block the symptoms before they appear. Thus, a prediction system becomes necessary.

1.2.1 Migraine Prediction

The Autonomous Nervous System (ANS) regulates body conditions through blood circulation (blood flow) at adequate rate. This lead to changes in the hemodynamic variables. The ANS regulates the heart and respiratory rate, sweating and vasomotor activity, among others. When a migraine occurs, changes appear in these variables. However, there are still many unknowns about how the dysautonomia is affected in a migraine patient and whether it is the cause or a consequence. In patients with aura or premonitory symptoms, the arrival of future pain can be mostly assured, but the prediction latency is unknown and pretty variable, as the symptoms can appear at any time from 48 to 6 h before the onset of the migraine. This prediction is almost useless for treatment purposes. As the current research shows a different dysautonomia in every patient, the study of this research will search primary a per-patient-based modeling technique.

There are no studies of sweating and oximetry that show any change between the baseline and symptomatic periods. With respect to the body temperature, there are contradictions between patients in the literature [123,136]. These findings support our search for per-patient models. Moreover, none of the previous studies included a systematic multivariable analysis of continuously acquired ambulatory variables. Finally, no studies have been found about the possibility to predict a migraine crisis using hemodynamic variables. The ANS in migraines regulates the aforementioned hemodynamic variables. For this reason, this study presents a way to predict migraines through these variables. No studies have been found about the

possibility to predict a migraine crisis; only Houle *et al.* in [80] have attempted to find some trigger (or absence thereof) to predict the cessation of a migraine attack to ultimately help with the prediction of the attack itself using only a dichotomized stress scale. Luciani *et al.* in [111] or Waelkens in [182] use premonitory symptoms, such as changes in character, appetite or sleep rhythm as predictors of migraines. However, these predictors are non-specific and are subjective markers that can appear in patients' daily lives for other reasons.

Pharmacokinetics defines the mechanisms of absorption and distribution of substances in an organism. Because of the pharmacokinetics of current drugs for treatment of migraine in the acute phase, premonitory symptoms and auras—some times—are not helpful to stop the pain—as it is difficult to estimate the onset of pain. Most migraine sufferers wait for the interval between period pain episodes to take the specific medication. The delayed intake reduces the effectiveness of the treatment. There are also some studies about the early intake of medication to abort the crisis. Goadsby *et al.* in [63] demonstrate that the earlier the intake, the more effective. In addition, Hu *et al.* in [83] demonstrate that specific migraine treatments, such as rizatriptan, can abort the migraine in 30 minutes. Other specific treatments, such as sumatriptan, reduce this time to as little as 10 minutes before the crisis starts.

The study we present in this paper focuses on four hemodynamic signals: heart rate (HR), electrodermal activity (EDA), skin temperature (TEMP) and peripheral capillary oxygen saturation (SpO₂). A multivariable analysis of these signals was used in an attempt to predict a migraine crisis.

Activity-related information is collected and used to distinguish between a migraine and other kinds of headaches. A robust patient model was developed in order to reduce the loss and the noise of data associated with ambulatory scenarios. According to Hu *et al.* in [83], this work is going to tackle the predictivity up to 30 minutes. Thus, prediction of the onset of a migraine attack will help the patient to stop the pain.

As stated at the beginning of this Chapter, the predictive modeling framework further presented, may need of data from external sources to improve prediction or extract relevant information. So, this has been considered in this thesis for the purpose of the migraine prediction. It has been used information from local and global environmental data and subjective symptoms such as premonitory symptoms. Some studies have indicated that anywhere from 7% to 61% of people suffering from migraines are susceptible to weather changes [79, 181]. With respect to the specific components, atmospheric pressure, humidity, wind and temperature have been implicated as potential factors for the appearance of headaches, but the relationship with them has not yet been

demonstrated. However, many of the patients often associate certain atmospheric conditions with headaches, in addition they describe changes in environmental factors that are triggering them, such as variations in pressure, luminosity, flashing lights, air quality, odors and noise [56]. In this thesis, these kind of variables are not going to be used to create fine-grained predictive models of the migraine, but they are going to be considered for an Expert Decision System [158] to improve the quality and robustness of the system.

As a chronic disease presenting symptomatic crisis, the migraine and the migraineurs are suitable for an ambulatory study. According to our estimations, the prediction of migraines for a target of 2% of population of European migraineurs would save more than € 1200 million. To achieve this, we propose the deployment of an unobtrusive ambulatory monitorization scheme of this population through the use of a Wireless Body Sensor Network (WBSN).

Using a WBSN monitoring the aforementioned hemodynamic variables, we pursue to create predictive models based on different techniques such as state-space models [177], or Grammatical Evolutionary algorithms, which presents several characteristics that makes it suitable for our purpose of predicting migraine crisis, such as automatic feature selection and creation of energy efficient models easily programmable in wireless monitoring devices.

1.3 Energy efficiency in Wireless Body Sensor Networks

1 Introduction

1.1 Mobile Cloud Computing and health

1.2 The migraine disease

1.3 Energy efficiency in Wireless Body Sensor Networks

1.3.1 Energy efficiency through computation in the node

1.3.2 Energy efficiency in the radio link

1.3.3 Holistic energy optimization

1.4 Publications

In the Internet of Things, dozens of commercial low power wireless monitoring devices allow making ambulatory monitoring easier. These WBSNs monitor biometric variables in a non intrusive way. The devices composing the WBSN are battery supplied and the capacity of their batteries is sufficient for short time operation. When an ambulatory monitorization is required, for

example in medical alarms, despite the usage of low power microcontroller architectures or more efficient wireless communication interfaces, improving the availability of the monitoring devices becomes a challenge.

The energy efficiency in WBANs has been studied in the literature to reduce the battery consumption and enlarge the availability of the monitoring devices. We can distinguish different levels to achieve energy efficiency in WBANs:

- Hardware: low power consumption electronic components
- Coding: efficient codes, bare metal programming, compiler optimization, *etc.*
- Wireless communication: efficient technologies, optimized spectrum utilization
- System: modular on/off control of electronic components
- Signal processing: computation on the device such as pre-processing, feature extraction, data removal, *etc.*
- Workload: high level decisions that can include some of the previous levels

The form factor of the battery limits the size of the monitoring device such that high density integration is still a challenge to deal with. Current commercial approaches use low power microcontroller architectures and high performance microcontrollers¹⁵, as well as more efficient wireless communication interfaces. Longer battery operation means also less disruptions in the monitored data. This is critical in the context of ambulatory monitorization for medical alarms. In this way, several research groups work to reduce the consumption to enlarge the battery life. Complete low power monitoring systems have appeared such as the one of the SmartCardia company [166]. Others, for example, study the consumption in medical monitoring devices at the system level, such as Tobola *et al.* do in [171] or Ghasemzadeh in [61]. Tobola *et al.* also study the dependence of the battery life with the sampling rate of the biomedical signals in [172] by reducing the sampling rate as much as possible, and studying how it affects to state-of-art signal processing algorithms in WBSN. Works like Braojos *et al.* [22] analyze the technique of compressive sensing in biosignals to reduce the energy consumption of such monitoring devices. One mechanism to reduce the energy consumption of the monitoring devices and increase the operation time of the battery is to reduce the complexity of the processing in the embedded microprocessor, and hence the number of clock cycles required to execute the code.

¹⁵Snapdragon: <https://www.qualcomm.com/products/snapdragon/> (accessed March 2018)

This research work targets three main energetic goals that interact and build a complex scenario. Based on a real application of monitorization for migraine prediction, this manuscripts studies the energetic impact of every element in a Mobile Cloud Computing scenario: (i) signal processing and data acquisition in the WBSN, proposing optimized modeling and prediction techniques with lower complexity than classical mechanisms, aiming a higher energy efficiency (ii) the radio link of the WBSN, and (iii) a holistic energy optimization considering workload balancing between all elements of the network (the WBSN, the coordinator node, and the Data Center).

1.3.1 Energy efficiency through computation in the node

Several works apply some data processing techniques in the node to achieve energy efficiency applied in the WBSN level. As an example, Mamaganian *et al.* [112] apply compress sensing techniques for low-complexity energy-efficient ECG compression, to reduce the amount of data to transmit and save energy. A low-power biosignal acquisition and classification system for ECG signal is proposed by Lee *et al.* [103].

In this research work there are developed and used energetically optimized predictive algorithms to run on monitoring nodes. We target the goal of increasing the operation time of the battery by the reduction of the energy consumption. This is achieved through optimization of two sources of energy consumption: (i) reducing the complexity of the processing in the embedded microprocessor of a monitoring device, that leads to reduce the number of clock cycles required to execute the code and, (ii) reducing the consumption of peripherals using the minimum number of sensors. To do this, we will focus on low performance microcontrollers to deal with the migraine prediction problem.

1.3.2 Energy efficiency in the radio link

Regarding the energy efficiency in the radio link there are two major concerns: (i) on body area communications between monitoring nodes, and (ii) communications from monitoring nodes to external devices (coordinator nodes or higher infrastructures acting as a back-end).

When there is more than one monitoring node, the wireless communication between them in a WBAN—typically with a star topology—is called an on-body channel communication. The blocking of the direct line of sight between the sensing nodes and the receiver due to the movement of users dramatically affects the link quality [39, 187]. Given the dynamic and complex nature of the on-body channels, the challenge is to maintain a good quality of the link between sensor

nodes while extending the network lifetime. This leads to the development of power transmission control policies to avoid data loss.

In the context of non-invasive WBSNs both, the human body and the node placement, affect the signal propagation, so that the radio network coverage can be reduced up to 1 meter. In the literature, different authors like Augustine [10], or Smith *et al.* [157] have shown that the quality of the on-body links in a non-invasive WBSN are highly dependent on the body posture, movement, the physical characteristics of the human body and the local environment. The movements of the user causes changes in the quality of the links, affecting their reliability [145, 168]. In this sense, some studies try to detect the body position to select the power transmission, such as in the work by Aulery *et al.* [11].

For this purpose, in this thesis it has been also considered this issue, and there are used the techniques developed by members of the research group in previous works [174] to minimize the radio link power without quality loss. These techniques are reactive policies for power level control based on algorithms using body posture detection. These techniques are trained with information of the Received Signal Strength Indicator (RSSI) level of the communication between the nodes and the coordinator, and the body posture. This allows to change automatically the power transmission to save battery.

In this work, we apply strategies to minimize the number of transmissions, managing the trade-offs between reducing the radio link power and increasing the processor power—due to the increase of signal processing.

Once data must be sent out of the body area network to the coordinator node—or the Data Center. Studies such as the presented by Casino [26] show the importance of the estimation of the wireless system operation in the radio link to reduce the energy consumption. There are several studies in the state-of-the-art that applies compression techniques and transformations to the raw data before wireless transmission. These can be compression methods—or transformations—with or without loose of information. As an example, the Fourier Transform and the Wavelet transformation are lossless transformations. In the transformed domain original data can be represented with less samples, and the original information can be retrieved in the back-end through inverse transformation, as done with ECG and Wavelet in [142].

There are other interesting techniques, such as Compressed Sensing (CS) to reduce the amount of data being transmitted. Compressed sensing allows us to create a representation of the original data in a transformed domain reducing the amount of data to transmit with an acceptable minimal information loss. Therefore, compressed sensing is more applicable when having a periodic signal such as ECG [30, 112], EEG [107], PPG [13] or motion data.

However, sometimes it is not necessary to keep all the information from biometric data, and just some features of them are relevant. These features may be extracted from a low-quality version of the monitored signal, such as Heart Rate (HR) from ECG. In this research work, it is going to be analyzed and optimized different compression techniques and transformations to substitute raw data transmission maintaining the accuracy of the predictive system and reducing the energetic consumption of the monitoring nodes. Furthermore, it is also going to be analyzed the energetic impact between the extremes of the radio link, *i.e.*, from the Wireless Body Area Network (WBAN) to the Data Center, as the decisions in one extreme affect to the other.

1.3.3 Holistic energy optimization

From a holistic point of view, the aim of this work is to reduce the overall energy consumption in the *eHealth* application. To this end, in this thesis we will focus simultaneously on: (i) minimizing the energy of the monitoring nodes via reducing the radio link power, (ii) developing strategies to minimize the impact of signal processing over the whole system, (iii) developing workload off-loading techniques to minimize energy consumption at the Data Center, and (iv) developing specific strategies to minimize energy consumption in the Data Center. We have already mentioned the problems of the two first in Section 1.3.1 and Section 1.3.2. This Section makes a review of the remaining elements—the Data Center—integrated in the network.

There are few contributions in the state-of-the-art considering a complete realistic scenario from a holistic approach. With respect to the work proposed by Rincon *et al.*, they provide a mechanism for the characterization and optimization of power consumption at the node level. However, they do not consider the relation of the node with the Cloud and the impact of their policies in the global energy consumption. In this research it has been incorporated their methodology at the node level but, on the contrary, this work targets the complete and realistic scenario of Mobile Cloud Computing, and performs a global optimization where the node, the servers and the network interact and trade off the savings.

Members of the research our group already tackled the problem of holistic optimization in an initial contribution by Zapater *et al.* in [188]. The work proposed by Zapater *et al.* targets a scenario similar to the proposed in this thesis, where all the agents of the Mobile Cloud Computing are considered. Their work can be considered as one of the initial contributions in this field. In this manuscript we will follow a similar methodology (development of orthogonal approaches at every abstraction level); however, this research upgrades their

previous work (i) as it has been considered a real workload in the monitoring nodes and servers (what requires the development of realistic policies, signal processing techniques and modeling approaches), (ii) the proposed approach is evaluated in the context of a clinical study with real patients and users, and (iii) there have evaluated the potential economic savings in a potential international market.

Data processing in the Cloud

Data obtained using the WBSN is communicated to an embedded processing element, *i.e.* a coordinator (usually a smartphone), that forwards the data to the Cloud. In order to detect, predict, create models *etc.*, huge data sets must be analyzed. To deal efficiently with such computationally intensive tasks, part of the processing and storage can be local to the coordinator, while another part can be communicated and processed in the Cloud, *i.e.* in Data Centers—the extreme end of the network. MCC problems targets problems that affect big populations, and the key challenge in MCC scenario is the definition of strategies to off-load and distribute efficiently the workload between the different elements of the system.

In this regard, one of the goals of this thesis is to balance efficiently the workload, reducing the computational burden in the Data Center and minimizing its energy consumption. Data Centers infrastructures consume a huge amount of power and generate a tremendous amount of heat. In year 2010, these facilities represented 1.3% of electricity use in the world [95]. In year 2012 alone, global Data Center power consumption increased to 38GW, and further rise of 17% to 43GW was estimated in 2013 [178]. A report for the US government estimated the country's data centers could need around 73GW of electricity a year by 2020 [153].

Data Center power budget is mainly devoted to the energy drawn by servers and the cooling needed to keep IT equipment under safe environmental conditions. In the last years, industry has devoted significant effort to decrease the cooling power, thus decreasing the Data Center Power Usage Effectiveness (PUE)—defined as the ratio between total facility power and IT power. According to a report by the Uptime Institute, average PUE improved from 2.5 in 2007 to 1.89 in 2012, reaching 1.65 in 2013 [116]. PUE values close to 1 are preferred. Moreover, academia has also focused on the development of Data Center optimization strategies to minimize energy from the computational and cooling perspective. However, in a MCC scenario, efficient computation off-loading, combined with tailored strategies to minimize the energy consumption at the Data Center, still represents an important challenge. This thesis devises new strategies to combine these two aspects: (i) workload

off-loading and (ii) local area networks, and (iii) data center energy efficiency techniques, to further reduce the energy consumption of eHealth applications. Further in the manuscript there will be presented and implemented a realistic case study for applied to the migraine prediction and it will be shown the overall impact of this strategies over the architecture of the MCC system.

1.4 Publications

1 Introduction

1.1 Mobile Cloud Computing and health

1.2 The migraine disease

1.3 Energy efficiency in Wireless Body Sensor Networks

1.4 Publications

1.4.1 Journal papers

1.4.2 Conference papers

1.4.3 Other publications

1.4.4 Intellectual property

The results of this thesis have lead to research publications in international conferences and journals, patents, grants and prizes as well. In the following lines these publications are shown with the detailed scores of each one. The contents of all the publications are distributed mostly in all the Chapters of this thesis.

1.4.1 Journal papers

This thesis has generated the following articles in international journals:

- PAGÁN, J., ZAPATER, M., AND AYALA, J. L. Power transmission and workload balancing policies in ehealth mobile cloud computing scenarios. *Future Generation Computer Systems* 78 (2018), 587–601
[JCR 2016=Q1, IF=3.997]
- PAGÁN, J., RISCO-MARTÍN, J. L., MOYA, J. M., AND AYALA, J. L. Modeling methodology for the accurate and prompt prediction of symptomatic events in chronic diseases. *Journal of biomedical informatics* 62 (2016), 136–147
[JCR 2016=Q1, IF=2.753]
- PAGÁN, J., ORBE, D., IRENE, M., GAGO, A., SOBRADO, M., RISCO-MARTÍN, J. L., MORA, J. V., MOYA, J. M., AND AYALA, J. L. Robust

and accurate modeling approaches for migraine per-patient prediction from ambulatory data. *Sensors* 15, 7 (2015), 15419–15442

[JCR 2015=Q1, IF=2.033]

1.4.2 Conference papers

This thesis has generated the following articles in international conferences:

- PAGÁN, J., MOYA, J. M., RISCO-MARTÍN, J. L., AND AYALA, J. L. Advanced migraine prediction simulation system. In *Proceedings of the Summer Simulation Multi-Conference* (San Diego, CA, USA, 2017), SummerSim '17, Society for Computer Simulation International, pp. 24:1–24:12
[CORE: B, MA: C]
Nominated for the best paper award
- PAGÁN, J., FALLAHZADEH, R., GHASEMZADEH, H., MOYA, J. M., RISCO-MARTÍN, J. L., AND AYALA, J. L. An optimal approach for low-power migraine prediction models in the state-of-the-art wireless monitoring devices. In *Proceedings of the Conference on Design, Automation & Test in Europe* (2017), European Design and Automation Association, pp. 1297–1302
[CORE: B, LiveSHINE: A++, MA: A, GGS Class: 2]
- PAGÁN, J., RISCO-MARTÍN, J. L., MOYA, J. M., AND AYALA, J. L. A real-time framework for a devs-based migraine prediction simulator system. *MAEB 2016* (2016)
- PAGÁN, J., RISCO-MARTÍN, J. L., MOYA, J. M., AND AYALA, J. L. Grammatical evolutionary techniques for prompt migraine prediction. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016* (2016), ACM, pp. 973–980
[CORE: A, LiveSHINE: A+, MA: A, GGS Class: 2]
- SANTAMBROGIO, M. D., AYALA, J. L., CAMPANONI, S., CATTANEO, R., DURELLI, G. C., FERRONI, M., NACCI, A., PAGÁN, J., ZAPATER, M., AND VALLEJO, M. Power-awareness and smart-resource management in embedded computing systems. In *Hardware/Software Codesign and System Synthesis (CODES+ ISSS), 2015 International Conference on* (2015), IEEE, pp. 94–103
[CORE: C, LiveSHINE: A, MAS: A, GGS Class: 3]

1.4.3 Other publications

The author has also contributed in the following articles in international and national conferences not specifically related to the contents of this thesis:

- FALLAHZADEH, R., ORTIZ, J. P., AND GHASEMZADEH, H. Adaptive compressed sensing at the fingertip of internet-of-things sensors: An ultra-low power activity recognition. In *2017 Design, Automation & Test in Europe Conference & Exhibition (DATE)* (2017), IEEE, pp. 996–1001

[CORE: B, LiveSHINE: A++, MA: A, GGS Class: 2]

Nominated for the best paper award

- GAGO-VEIGA, A. B., DE ORBE-IZQUIERDO, M. I., SOBRADO, M., PAGÁN, J., CARRERAS, M., AYALA, J. L., AND VIVANCOS, J. Monitorización ambulatoria no invasiva de variable biométricas en pacientes con migraña. ¿es posible predecir una crisis? In *XLVI Reunión Anual de la Sociedad Española de Neurología* (2014), Sociedad Española de Neurología, pp. 94–103

Stellar communication award

1.4.4 Intellectual property

The methodology presented in this thesis has led to the development of intellectual property:

- GAGO, A., SOBRADO, M., MORA, J. V., PAGÁN, J., ORBE, D., IRENE, M., AND AYALA, J. L. Method for determining the degree of activation of the trigeminovascular system, 01 2017

PCT. Number: WO/2017/149174. Status: Application. Currently studying the entrance to national phases

- GAGO, A., SOBRADO, M., MORA, J. V., PAGÁN, J., ORBE, D., IRENE, M., AND AYALA, J. L. Método para determinar el nivel de activación del sistema trigémino-vascular, 02 2016

Spanish National Patent. Number: ES201600158A. Status: Granted. Currently studying a patent license contract with the entities.

Chapter 2

Architecture of the robust predictive system of symptomatic crises

An eventual crisis is something that happens at the end of a period of time or a process. The objective of this thesis is the prediction of events in critical scenarios. In the case at hand, these events occur always after a process of changes in different variables that can be measured and correlated with the events. When the eventual crisis occurs in a biomedical scenario we refer them as symptomatic crisis.

As stated in previous sections, the framework that encompasses this thesis is framed into an IoT scenario. In these scenarios there might be multiple data sources, and they can be distributed and uncorrelated. In most of the cases, data come from low performance monitoring devices with limited resources such as low autonomy (short battery life), low data acquisition rate, low computation capabilities, low power transmission (in wireless connections), limited baud rate when sharing data, *etc.* All this leads mainly to unreliable data due to disruptions over acquisition time, which requires of a robust architecture against failures in data in order to ensure/ maintain predictability of critical events.

The architecture of the system and the prediction methodology proposed are orthogonal to the problem. Problems where this methodology can be applied must have temporal character in order to be able to correlate changes in variables with a critical event. This thesis has been developed in an IoT framework, in the field of *eHealth*. In the following sections firstly the architecture of the system is briefly presented as case study in three different paradigms of the IoT in Section 2.1. Then a top-down view of the architecture is shown Section 2.2, and finally, an overview of the energy problem of its implementation is shown in Section 2.3.

2.1 Robust predictive paradigms in the IoT

2 Architecture of the system

2.1 Robust predictive scenarios

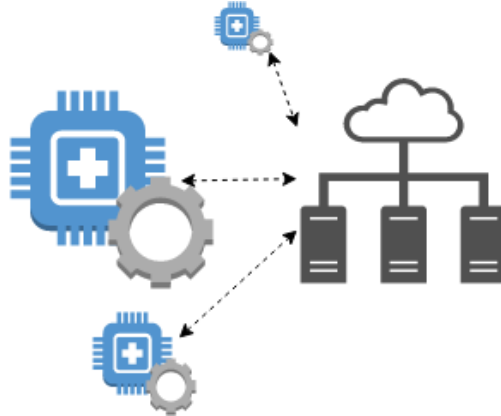
2.2 Model of the system

2.3 The energy efficiency problem

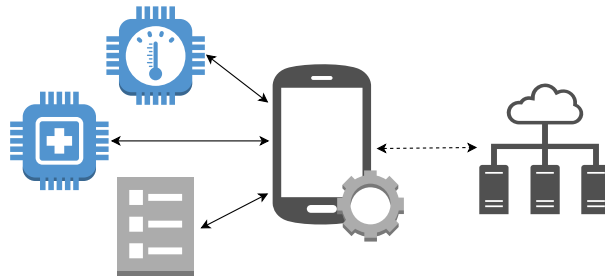
There are different elements that compose the IoT. Atzori *et al.* divide the IoT into three elements in [9]: (i) things oriented (data acquisition systems—sensors—and actuators), (ii) internet-oriented (middleware able to storage, transmit and perform some data analysis), and (iii) semantic-oriented (knowledge able to perform computationally expensive data analysis). In [66], many different healthcare uses—*eHealth* applications—are described in the IoT: triage, patient monitoring, or disease spread modeling and containment for real-time health status, or predictive information in pandemic scenarios. In this thesis we focus on robust predictive systems for detection of critical events using Wireless Sensor Networks (WSNs) in an IoT scenario.

Data acquisition systems are data sources that are generally distributed—like WSNs—and they can be separated from the corresponding prediction system that processes these data. Prediction systems represent the knowledge of the network, and they can be centralized in a high performance infrastructure as a Data Center (or servers in the Cloud). They can be distributed as well, linking sensors and actuators, which they provide the prediction support. A prediction system can be an hybrid solution as well, which is even better: processing distributed between the sensors and the data centers. It depends on how the whole system is implemented. There are multiple combinations and the solution is not unique, leading to different layouts. Some of the best-known system’s layouts are: (i) the distributed computing (DIST), (ii) the locally centralized Fog computing (FOG), and (iii) the centralized Mobile Cloud Computing (MCC).

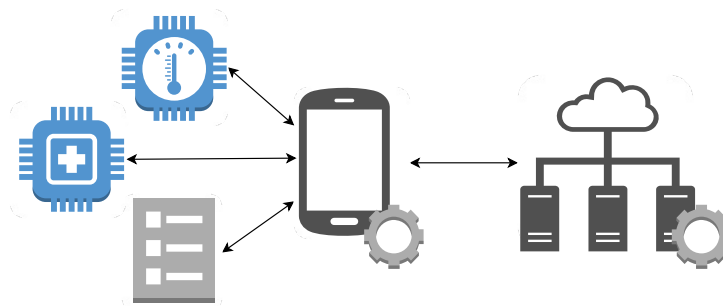
Figure 2.1a shows an example of a data acquisition system that belongs to a DIST network. Each one of the acquisition systems gathers and processes the data to perform predictions. Once in a while, the acquisition system sends status information to the Cloud. In this example the Cloud is used for the purpose of visualization and calculation of statistical parameters such as the number of critical events detected or and the characteristics of these events. These acquisition systems have the enough computation capabilities to process data and perform predictions, but limited wireless communication thus, raw data are never sent to the Cloud. Contrary to FOG, predictions and decisions in DIST are taken



(a) Distributed computing (DIST): the knowledge is distributed on each monitoring node, which performs predictions by their own. Communication with the Cloud is sporadic.



(b) Fog-computing (FOG): the knowledge runs on a coordinator. The communication is bidirectional through the nodes. Communication with the Cloud is sporadic.



(c) Mobile cloud computing (MCC): the knowledge runs either in a coordinator or in the Cloud. The communication is bidirectional point-to-point.

Figure 2.1: Three different perspectives a robust predictive systems for detection of critical events in the IoT.

autonomously without knowledge from other data sources.

In order to improve predictions, the FOG network in Figure 2.1b gathers data from several locally distributed data acquisition systems. Data is locally centralized in a coordinator—mobile devices—at the network edge that has the knowledge to run predictions from all data sources. Devices might also have light computing capabilities such as filtering and data preprocessing, in order to provide only useful data to the coordinator. This coordinator can manage the monitoring nodes, which are only connected to the coordinator. Same as DIST, the system only sends status data to the Cloud, but this data have more information thus this comes from several monitoring nodes. Depending on the amount of computation to be performed, FOG might be energetically inefficient, due to the low-medium performance of coordinator—typically a smartphone running many other applications. In order to mitigate this effect, the MCC shown in Figure 2.1c can balance workload computing through the network.

In the MCC network, mobile devices and cloud computing combine each other to create a new infrastructure where data processing and predictions can take place either the coordinator or the Cloud, or it can vary along the time. MCC can provide more computing power, storage and networking services (as visualization and statistical analysis) than FOG, but it leads to new energy efficiency problems in the whole system to deal with.

2.2 Model of the system architecture

2 Architecture of the system

2.1 Robust predictive scenarios

2.2 Model of the system

2.2.1 Data Acquisition System

2.2.2 Robust Prediction System

2.2.2.1 Data Driver

2.2.2.2 Sensor Status Detector

2.2.2.3 Prediction System

2.2.3 Decider

2.2.4 Expert Decision System

2.3 The energy efficiency problem

Figure 2.2 shows the overview of the architecture of the system proposed. This has been named as **Critical-Events Robust Prediction System (CERPS)**

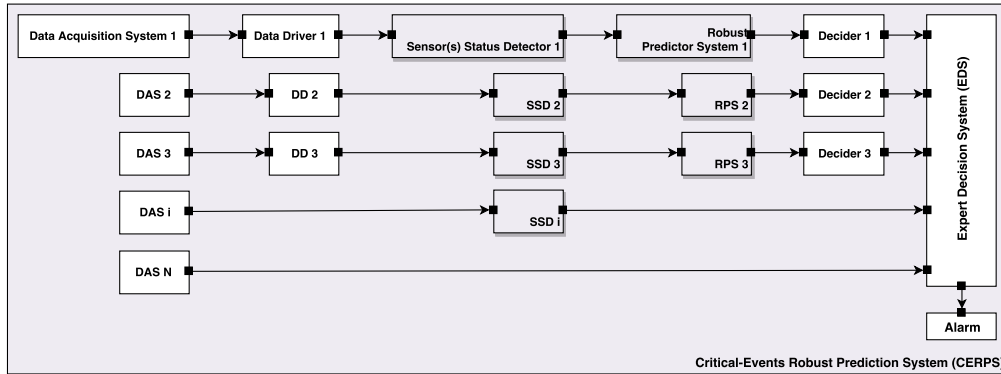


Figure 2.2: Model of the system architecture. This scheme represents a specific case study. The CERPS is scalable and there might be as many DAS and RPS as needed.

and it is composed of three type of subsystems to compute predictions: (i) **Data Acquisition Systems (DASs)**, (ii) **Robust Prediction Systems (RPSs)**, and (iii) **Expert Decision System (EDS)**. The CERPS performs predictions of critical events though the extraction of information of heterogeneous data sources and based in a expert decision system, the EDS. There might be as many data sources (DASs) and prediction systems (RPSs) as needed, but only one EDS is required.

Depending on the nature of data provided by a DAS, these might be an RPS or not; furthermore, raw data can directly be driven to the EDS as shown in Figure 2.2. The CERPS is easily scalable, and the addition or the removal of acquisition systems and prediction systems only leads to retrain the expert decision system. In Figure 2.2 five DAS are required, nevertheless this is an example. The architecture shown can be seen as an abstraction layer, a methodology, orthogonal to the actual implementation of the network.

2.2.1 Data Acquisition System

The DASs are data sources that are generally distributed, and they can be separated from the corresponding RPSs. The prediction systems can be centralized—in a high performance infrastructure as a Data Center (or servers in the Cloud), or they can be linked to the DAS to which they provide the prediction support. It depends on how the CERPS is implemented; there are multiple combinations and the solution is not unique, and some of the best-known system’s layouts are the DIST, FOG and MCC described in Section 2.1.

The data acquisition systems can be, for example, actual physical monitoring nodes—in most of the cases—or remote web services. Distributed DAS might lead to limitations with data correlation, and it constraints the system design; fortunately this problem is tackled by the Data Drivers. Some of them can even be hosted

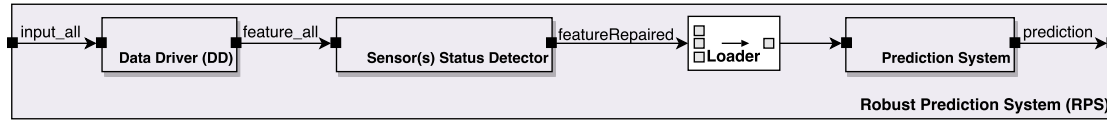


Figure 2.3: Robust Prediction System architecture. This is the most important subsystem and it can be an independent entity by itself out of the system.

in the same physical device, as a smartphone. What defines a DAS is the way its data need to be processed or used. Each DAS provides information to a different RPS, or none. DASs may provide information of different sensors/ services at the same time.

2.2.2 Robust Prediction System

The Robust Prediction Systems are the most important subsystems of the architecture (see Figure 2.3). Data are received from the DASs, and alarms are notified to the decision support system. This alarm is only generated from the knowledge extracted of one single data acquisition system, and it is not the final alarm that warns of the onset of a crisis.

Four subsystems shape the RPSs proposed. In the order in which the data is handled: (i) one Data Driver (DD) to pre-process the data (Section 2.2.2.1), (ii) one Sensor Status Detector (SSD) to repair signal errors (Section 2.2.2.2), (iii) one Prediction System (PS) where predictions are computed (Section 2.2.2.3), and (iv) finally one alarm (Decider) (Section 2.2.3). Among them all, the Prediction System (PS) represents the core of the whole architecture (the CERPS).

2.2.2.1 Data Driver

Data Drivers (DDs) are divided into modules (Figure 2.4); these modules are different depending on the data type: qualitative or quantitative data. Each one of the sensors/ services of a DAS may have different time sources, so different timestamps. The Data Driver can be (i) before the Sensor Status Detectors as the most common layout (shown in Figure 2.4), or (ii) after the Sensor Status Detectors.

If (i) the DD is before the SSD, the input of a DD is a matrix of data and timestamps from different sensors/ services of the same DAS; and the output is a new array of synchronized features or parsed data—that may have errors—labeled as *feature*. If (ii) the DD is after the SSD, the input is a matrix of error-free data and timestamps as gathered from the DASs; and the output is a new array of synchronized and features or parsed data.

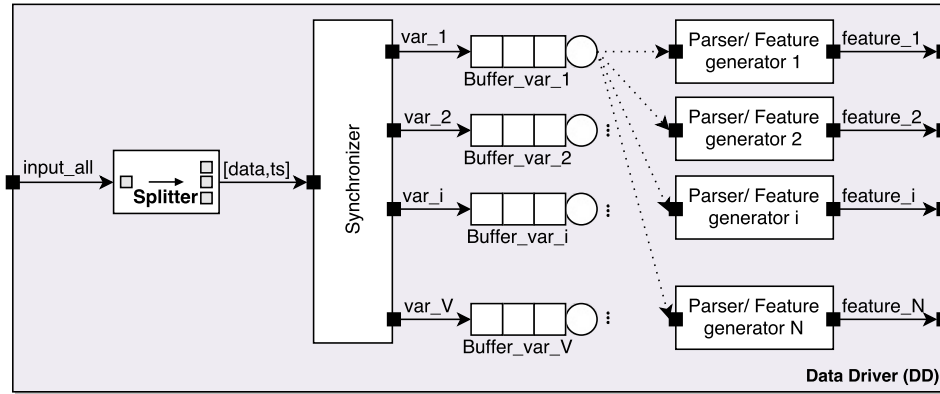


Figure 2.4: Data Driver. It is shown how one measured variable can lead to one or several features to be used in the subsequent modules.

Data coming from the **Data Acquisition Systems** are split into V different variables with their timestamps. For quantitative variables the sampling frequency must be known by the system in order to synchronize the data—using a common timestamp. Buffers are FIFO and size limited (size varies for each variable). When data are synchronized they are temporally buffered and after a while parsed to compute N features, if needed (the time that data is buffered depends on the features to compute). Qualitative data are also synchronized but they are not parsed, and features are not extracted either; in most of the cases they are considered features themselves.

Notice that the number of features and the number of variables is not the same and, N can be lower, equal or higher than V . Several features can be obtained through the information of different variables, and more than one feature can be obtained from one variable.

Data coming from the same **DAS** belong to the same data type. The type of the variables depends on the application of the system but commonly quantitative variables represent physical magnitudes varying along time and sampled at a certain rate. Besides, qualitative variables represent mostly non time-dependent information as events or actions happening once in a while.

2.2.2.2 Sensor Status Detector

The **Sensor Status Detector SSD** is the first block that gives the robustness quality to the **Robust Prediction System** and thus to the title of this thesis. The **SSD** (see Figure 2.5) monitors the status of sensors, generate alarms if something is wrong with them and repairs signals temporally.

Measurements in real scenarios are prone to data loss due to distributed data sources, with varying and unreliable channels and using sensors in

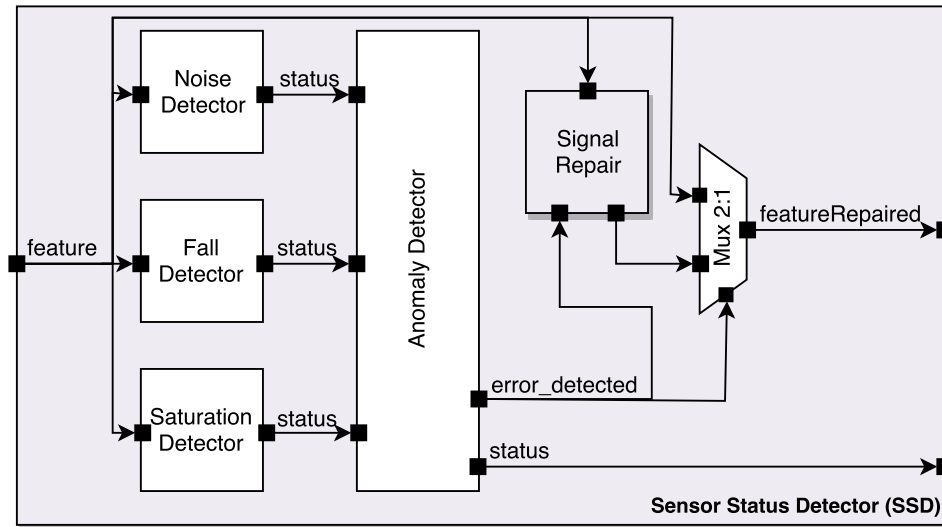


Figure 2.5: Sensor Status Detector. This module gives the robustness to this part of the system. It detects data errors and warns other modules to maintain the accuracy of the prediction.

non-static devices/ machines in noisy environments. The first step that the SSD performs is the detection of the problem. According to Kong *et al.* in [94], data loss in WSNs does not follow a random distribution and they divide data loss patterns into four different classes:

1. Element Random Loss (ERL): the simplest problem due mainly to noise.
2. Block Random Loss (BRL): adjacent time slots are dropped together. Data collision and noise are the main factors.
3. Element Frequent Loss in Row (EFLR): mainly due to interruption in transmission or bad sensor connections.
4. Successive Elements Loss in Row (SELR): time slots without data. This may occur when nodes are damaged or batteries run out.

In real biophysical scenarios using WSN, data loss is a combination of the aforementioned loss patterns. To detect them, three blocks have been included in the SSD to detect the cause of error. Causes of error have been classified into three: noise, fall (disconnection), and saturation (due to battery problems mainly). The blocks implemented are (see Figure 2.5): the **Noise Detector**, the **Fall Detector**, and the **Saturation Detector**. Each one of these blocks generates an alarm. The decision to raise an alarm can be as simple as a threshold or something more sophisticated as a fuzzy logic algorithm (see Section 3.2.1).

Whenever the data processing takes place, in a server of a distributed network, or in the same device that makes measurements in a MCC scenario, this is a problem that must be tackled before predictions can be done, otherwise, data loss might lead low reliability erroneous solutions and alarm failures. The solution is the substitution of the sensor if it is damaged, or warn to avoid the situation that causes data loss: noisy environment, movements, *etc.*

The SSD performs data recovery for a while if none of these solutions is taken. Data recovery takes place in the **Signal Repair** module shown in Figure 2.6. This module is enabled when the **Anomaly detector** activates it. This happens when one or more errors are detected. When an error occurs, the **Anomaly detector** warns that the sensor has problems, and immediately the Signal Repair module is activated. If the anomaly takes too much time to be removed, the **Core algorithm** would not have enough recent data to estimate new values. In this case, when a predefined time is exceeded, the **Anomaly detector** activates an alarm signal, and the data from the damaged sensor is not used for prediction until the problem is solved. An external indication might indicate that the sensor has been restored before the error detection modules realizes. Then all alarms signals are disabled and SSD relays the data from the sensor without errors.

Signal Repair

During normal operation, the **enable** signal equals 0, and multiplexers drive the data value **feature_i** to output.

In these modules, it is supposed that we work with temporal data and there exists a time relation between samples of data¹, so that when this module is activated, data are recovered or repaired based on recent past samples of the damaged sensor. The amount of past data to be used to recover the current value must be optimized for each feature and the **Core algorithm** implemented. Different algorithms can be implemented to calculate the current missing value (some times named as prediction in the literature). If the algorithm runs on a monitoring device with constrained computing capabilities, time-series algorithms can be suitable to perform the recovery, on the contrary, if the algorithm runs on a platform with high computing resources, more complex methods can be applied, such as Gaussian Process Machine Learning [138] which infers statistical outputs with a confidence interval instead of a single value. For a better comprehension see Section 3.2.2.1.

¹Spacial relation between data monitoring devices is not taken into account in the study cases of this thesis, but it might be useful in other IoT scenarios such as traffic monitoring in smart cities.

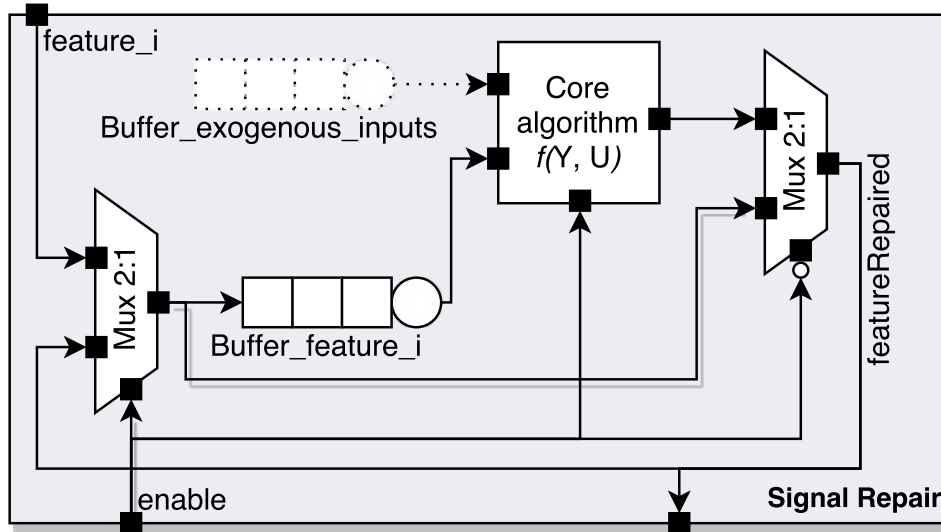


Figure 2.6: Signal Repair. Based on a core algorithm, this module repairs data loss temporally. After a while, if the error persists, other robust techniques further in the system must be applied.

One feature can depend on its own past data, but also it might depend on the history and current values of other signals. As an example, in a biomedical application, temperature and electrodermal activity can be correlated, such that data from one signal can help to recover the other if its damaged or lost. To do this, the **Signal Repair** module stores data into a buffer for exogenous (external) inputs. Both buffers (**Buffer_feature_i** and **Buffer_exogenous_inputs**) are FIFO and size limited. The **Core algorithm** should be trained to support both data inputs and eventually it provides a estimated calculation (**featureRepaired**) of the damaged signal data.

2.2.2.3 Prediction System

When data from all SSDs are available, they are merged into a **Loader** of the **Robust Prediction System**, prior to perform the prediction. This loader buffers all data into a synchronized data array.

There are three major blocks in the **Prediction System (PS)**: (i) the **Sensor Dependent Model Selection System (SDMS2)**, (ii) fc different types of **Predictors**—each one depending on a different set of features—and, (iii) the **Linear Combiner**. The behavior of each one of these modules are described in the following lines.

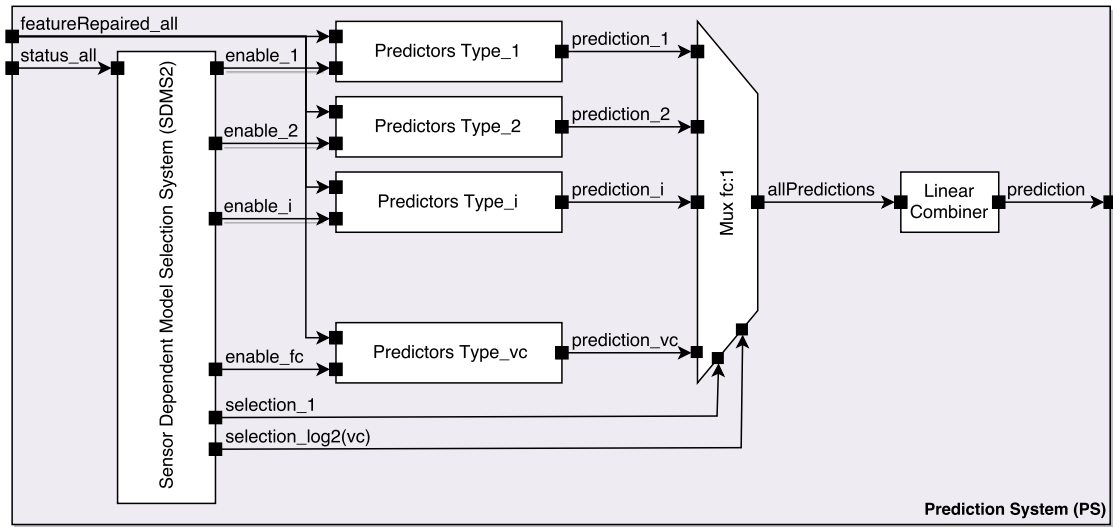


Figure 2.7: Prediction system. The SDMS2 is the core of this module that allows the adaptation of predictive models when data errors last in time.

Sensor Dependent Model Selection System

At this point of the thesis it must be clear that in real scenarios, sensor failures are a big problem to deal with, so that, the lack of information from one of the multiple data sources might not be condition to stop running the system up to the solution of the problem in real system for the prediction of critical events. In the proposed modeling methodology the robustness against sensor failures is always considered and it affects the design of all modules. Previously, it was described the **Signal Repair** module to provide error-free data to the predictors as a temporal solution; in this case, the **SDMS2** is in charge of the selection of the adequate predictors when data from an affected sensor is not useful anymore.

For a better comprehension of the **SDMS2**, we must understand it as a methodology that chooses the best set of predictive models based on the statuses of sensors in real time. This methodology is the second point that gives the robustness quality to the **Robust Prediction System**—as the **SSD** was the first one. Despite this system is shown in Figure 2.7 as a switch that enables or disables the **Predictors** modules, the **SDMS2** is a methodology sorts predictive models hierarchically depending on the availability of sensors, and it enables the predictive module that does not use damaged sensor information. With this the system does not stop and tries to keep accuracy in the prediction. In Section 3.2.3.1 it is explained how the **SDMS2** is built.

Predictors

To be adapted to the worst case, when all V variables but one are damaged, there would be $VC = \sum_{i=1}^V \binom{V}{i}$ possible variable combinations to create predictive models. In the practice—but depending on the application—predictive models based on couples or single features do not provide accurate predictions and a subset $vc \subset VC$ is used only, thus vc types of **Predictor** modules.

Predictors execute predictive algorithms to calculate the desired output h steps ahead (seconds, minutes...). h represents the prediction horizon, *i.e.* the time between declaration of an hypothetical event and the event itself. Algorithms create models to express mathematically a function that represents the system's output—in our case a critical event—according to information from inputs. Each **Predictor** module executes one or more of these models to provide one or more predictions respectively. It has been proved that a linear combination of predictions—from different models—provide more accurate results and larger prediction horizons.

Many different predictive algorithms are suitable to run over time series data to predict events, such as state-space algorithms, time series analysis, artificial neural networks, or Grammatical Evolutionary (GE) algorithms among others. These models use past data to calculate a single future value, so they have internal buffers from past input data. Autoregressive models are those that use information from past known system's outputs and predictions too. **Predictors** that run autoregressive models buffer predictions too (as done in Figure 2.6). Eventually, when all modules have computed predictions these are available as an array in signal `allPredictions` (Figure 2.7). According to the relevance of each model of the **Predictor** module, the **Linear Combiner** weights predictions and gets as a result a sole prediction that feeds the **Decider**.

2.2.3 Decider

As shown in the system's architecture Figure 2.2, a **Robust Prediction System** provides predictions to the corresponding **Decider** to raise individual alarms. **Deciders** can take decisions based on the current prediction value, or based on past predictions too—where past data are temporary stored in a local buffer. **Deciders** evaluate a core function to raise alarms. As shown in Figure 2.8, this is not the final alarm of the **Critical-Events Robust Prediction System** but a `local_alarm`.

The core function (in Figure 2.8) that raises alarms can be simple or complicated mathematical decision functions. Three different examples of

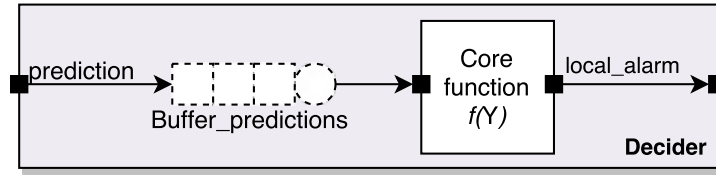


Figure 2.8: Decider. Based on a core function, this module generates a local alarm for the Expert Decision System.

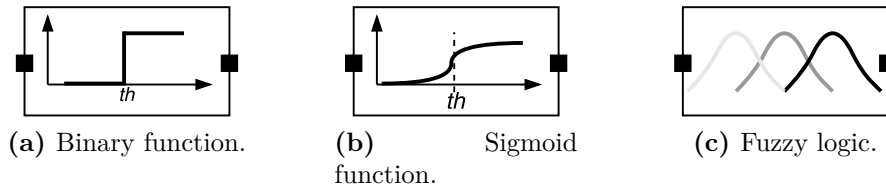


Figure 2.9: Three examples of core functions for the Decider module.

possible core functions are drawn in Figure 2.9: (i) a binary threshold decider (Figure 2.9a) where the local alarm $a_l \in \mathfrak{B}$ raises when the current prediction or a weighted averaged value of buffered predictions p_a exceeds a threshold th , (ii) a general case of sigmoid function (Figure 2.9b) where the alarm $a_l \in \mathfrak{R}$ is a softer version of the binary one, and (iii) a fuzzy logic function that represents the alarm $a_l \in \mathfrak{R}$ as a result of the *fuzzyfication* of the current prediction or individual past predictions.

In a predictive system with only one DAS and RPS, the final alarm over the critical event would be the local alarm generated by the Decider module, and the Expert Decision System is not necessary. Otherwise, these local alarms fed the DAS, the last module that provides the final decision.

2.2.4 Expert Decision System: triggering alarms

The Expert Decision System module is a computer-based system to aid triggering the alarms of the Critical-Events Robust Prediction System. In a real scenario with multiple data sources that can suffer from temporal data loss, the automatic generation of decision algorithms is not suitable. To deal with unlabeled data, or data missing values, this module has been constructed using Active Learning (AL) algorithms. Despite it requires of human interaction during the training process, unlike pure Decision Support Systems (DSS), the EDS works in autonomous way. The EDS receives all the information from the Deciders and other modules as shown in Figure 2.2 and based on a definitive prediction model, seeks to improve the result of individual alarms of the deciders.

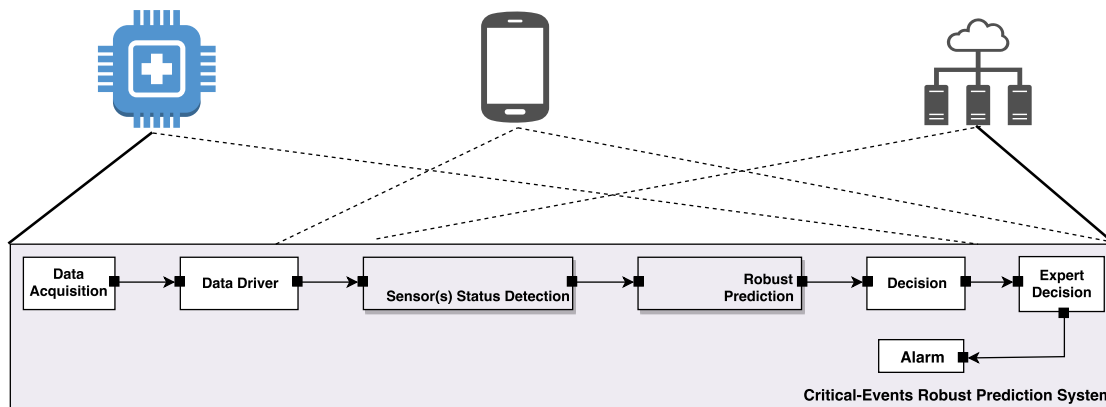


Figure 2.10: Energy consumption and workload distribution. Tasks distribution for the three network elements. Dotted lines represent that the tasks under the network element might be carried out by any of them. Solid lines force the task execution in the element above.

Active Learning bases on machine learning techniques to create a decision model or classifier. AL is a semi-supervised machine learning that communicates with the user to ask for the output for unlabeled input data. This basis function can be performed using Support Vector Machine (SVM) algorithms, decision-tree based algorithms, Adaptive Neuro-Fuzzy Inference System (ANFIS), or any other. In our case, the decision model determines the occurrence, or not, of a critical event.

2.3 The energy efficiency problem

2 Architecture of the system

2.1 Robust predictive scenarios

2.2 Model of the system

2.3 The energy efficiency problem

Independently of the actual implementation of system's architecture, the energy efficiency is a problem to deal with in IoT scenarios. These scenarios, characterized by its geographical distribution and the wide range of computation systems that imply, require an intelligent use of resources to optimize the energy consumption of the whole system.

Large scale population monitoring systems in MCC scenarios are starting to become a reality. In a general way, but with the *eHealth* field in mind, it has

been depicted the tasks assignment in Figure 2.10. In this figure, each element of the network (monitoring devices, coordinator nodes and cloud servers in a data center) encompasses loosely a set of task of the defined architecture that may or may not perform. Solid lines beneath the monitoring node and the Cloud elements represent that the extreme tasks **Data Acquisition** and **Expert Decision** must be performed by the extreme elements respectively. On the other hand, dotted lines represent that the tasks under the network element might be carried out by any of them. As an example, we will refer to the distributed architecture in Figure 2.1a when the coordinator does not perform any action but the **Data Acquisition**.

In this thesis it is shown that it is possible to optimize energy consumption with economical constraints and taking into account the computing capabilities of each element of the network in an *eHealth* application. So, in addition to be robust and data independent, the aim of the proposed methodology is also to reduce the overall energy consumption in a real world application, from a holistic perspective. To this end, in Chapter 3 focus on: i) minimizing the energy of the monitoring nodes via reducing the radio link power, ii) developing strategies to minimize the impact of signal processing over the whole system, iii) developing workload off-loading techniques to minimize energy consumption at the Data Center, and iv) developing specific strategies to minimize energy consumption in the Data Center.

As aforementioned, all the different perspectives presented can be applied in many different fields. It is only a matter on the how the optimization and tasks distribution are made what differ from one application problem to another. The implementation of the methodology, the architecture, predictive model algorithms, data processing and other techniques, such as optimization techniques implemented are shown in Chapter 3. And the results obtained are presented in Chapter 4 to satisfy the methodology framework drawn as research objective in Chapter 1.

Chapter 3

Implementation of the system

This thesis has had a marked practical character from the early beginning of its conception. The architecture shown in Chapter 2 would not have sense in a purely theoretical framework and the implementation of the system become reality from the first steps. Because the architecture was developed by the specification of a Model-Based Systems Engineering (MBSE) system [48] following the principles and best practices of the Model-Based Engineering paradigm [18] (MBE)—also known as Model-Driven Engineering (MDE)—leading, as a consequence, to the simulation (Modeling and Simulation-based Systems Engineering, M&SBSE) and implementation of the system described in this Chapter.

The implemented system follows the architectural structure of the Critical-Events Robust Prediction System, CERPS, shown in Figure 2.2. The implementation of the methodological framework stated in this thesis has focused, as aforementioned, on the prediction of symptomatic crises of the migraine disease, due to its relevance and novelty in and for the state of the art. To serve as a guide, Figure 3.1 represents a mind-map. This is an equivalence graph between the formal architecture elements described in the previous Section, and the actual implementation of each module. At the date of writing of this volume, the CERPS has been divided and implemented as shown in Figure 3.2.

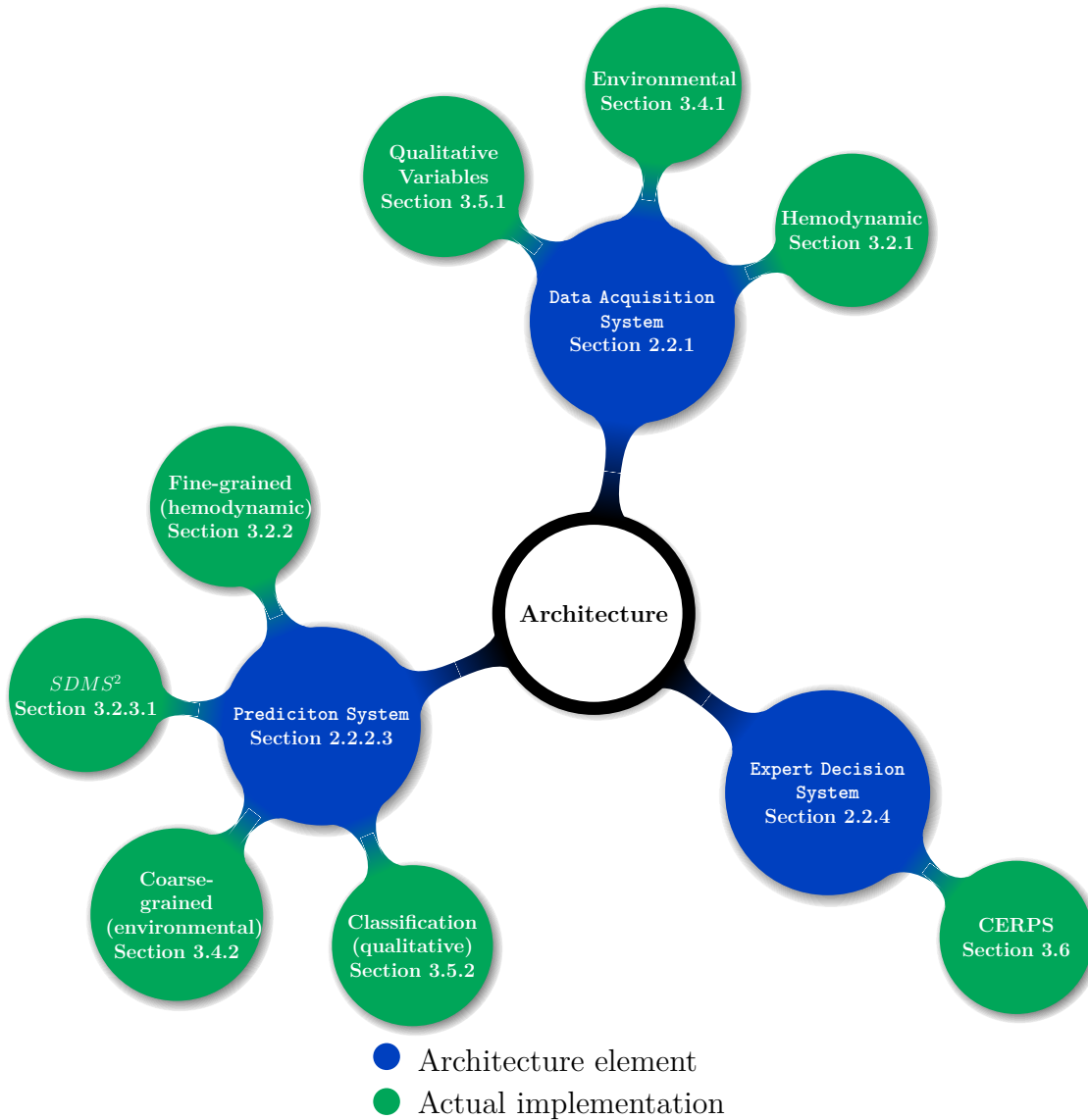


Figure 3.1: Equivalence graph between the architecture described and its actual implementation.

On a conceptual level, there are three well differentiated parts in this Chapter: (i) fine-grained prediction (Section 3.5 and Section 3.3), (ii) coarse-grained prediction (Section 3.4, Section 3.5, and Section 3.6), and (iii) energy efficiency frameworks for different MCC implementations.

The CERPS depicted in Figure 3.2 represents the actual online implementation of the system. It can be shown that there are three major predictive sub-systems (each one divided, at the same time, into smaller subsystems):

1. Sub-system I. It is the main and most important subsystem and it is the one

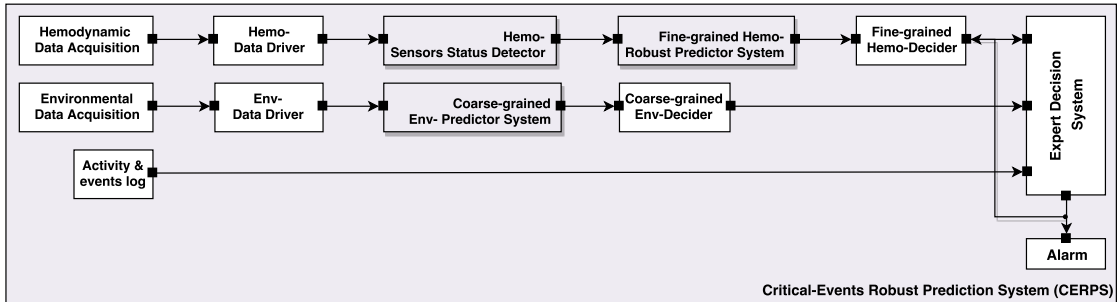


Figure 3.2: Overview of the implementation of the whole system architecture. There are three types of data sources: two of them perform fine and coarse predictions, while the third one serves as prediction support to the Expert Decision System.

that has more relevance and interest. This provides a fine-grained migraine prediction based on the measurement of hemodynamic signals. It has been the first time in the literature that this issue has been tackled, and it has attracted the interest of media, health insurance companies and the pharma sector, giving rise to national and international prizes. This sub-system has been fully studied and, despite there is still a lot of research to do on it, it has been modeled, simulated and implemented. Due to this, the research in this thesis has focused in Sub-system I, so this constitute the bulk of this Chapter.

2. Sub-system II. A coarse-grained migraine prediction based on environmental data is provided by the second subsystem. This system has a lower relevance, and helps the Expert Decision System, EDS, in the decision making process.
3. Sub-system III. A classifier based on qualitative data—occurrences or events that surround the prediction problem—is the third subsystem. Events are classified into categories; the probability of belonging to one or another category acts as information that helps the Expert Decision System as well.

At the time of writing this manuscript, the implementations of Sub-systems II and III have a marginal contribution and they have been merely conceived as functionally tests by the moment; nevertheless they are carefully described and commented further below.

Along this Chapter, the implementation is going to be distinguished into *offline* and *online* implementation. The offline implementation encompasses the training of predictive models and the search of parameters of other modules of the system. This training phase requires of some steps that the online implementation do not. This leads to the final models and parameters implemented in the online

phase. Section 3.5 and Section 3.6 go deeper each subsystem, describing both offline and subsequent online implementations. The framework underlying the energy efficiency aspects of the implementation of Sub-system I is finally shown in Section 4.4.

Implementation, and research studies collaborations

This thesis has been developed within a research framework which involves several research groups. This is an on going research and this manuscript settles the technical foundations for further studies. In parallel to this thesis it has been developed a clinical study which has lead to clinical conclusions, and has established new criteria for further clinical researches as well.

The clinical study associated to this research started in August 2014 (see Appendix E), led by the Head of the Neurology Service of the Hospital Universitario de La Princesa of Madrid, MD PhD José Vivancos. The clinicians developed a clinical protocol including ethical, technical and clinical aspects of the study. They conducted the clinical part of the study whose main aspects are (i) recruitment of patients according to inclusion criteria previously stated, (ii) explanation and training of patients participating in the experiments, (iii) follow up and feedback of clinical mishaps taken during the experiments, such as urticaria due to the continuous contact to the skin sensors.. The inclusion criteria are:

1. The diagnosis of migraine must have been made by a headache specialist neurologist.
2. Patients between 15 and 69 years old, who have at least one crisis per week.
3. Patients with a basic knowledge of the use of a smartphone and sign a commitment to collaborate during the collaboration period.
4. Patients with migraine with and without aura may participate.
5. The patient may continue taking their usual medication, although patients who do not take any medication prescribed daily for pain are preferred.
6. They should not have serious cardiac or pulmonary pathologies, Sjörgren's syndrome or poorly controlled thyroid changes. Not being pregnant or breastfeeding.
7. The study takes between 4 and 6 months.

During the first year, there were recruited thirteen patients. Most of them reported few migraines or events with low signal quality during monitoring. To develop the technical methodology proposed in this thesis, there were used two patients. To conclude with statistical clinical results, this sample size is not enough and, at the time of writing of this manuscript, the study continues incorporating more patients. These will be analyzed in a near future to lead to clinical conclusions.

On the other hand, the author of this manuscript as well as his advisors are members of the Research Center for Computational Simulation-CCS. This center hosts many researchers of different groups belonging to four public universities in Madrid. This has lead to collaborations with students of the GreenLSI group in the Technical University of Madrid, and the GreenDISC group in the Universidad Complutense de Madrid. The author as also collaborated with the GDAF group of the Universidad Carlos III de Madrid.

The content of this thesis is completely novel. From the acquisition, up to the data processing and modeling, everything presented in this volume are original contributions. In parallel, and thanks to the research collaborations, there have been developed tools mainly focused on (i) environmental data acquisition experimental set-up, (ii) energy consumption and measurement in wearable devices (radio and processing), (iii) prototyping of new sensors (not shown in this thesis), and (iv) testing of data integration and modeling techniques. As a result, these have led to dissertations of several bachelor and master theses, and papers which are referenced further in following chapters.

3.1 Overview

3 Implementation

3.1 Overview

- 3.2 Sub-system I. Fine-grained modeling
- 3.3 DEVS-based migraine predictor simulator
- 3.4 Sub-system II. Prediction support
- 3.5 Sub-system III. Prediction support
- 3.6 System implementation: CERPS
- 3.7 Energy aware prediction system

Along the sections in this Chapter, it is used a wide spectrum of algorithms and techniques for predictive modeling, classification, signal repair, or simulation among others. The use and the experimental set up—the actual implementation—are explained here, but the theoretical background is provided in Appendix A.

The objective of this Appendix is to explain the fundamentals of modeling and simulation techniques used in this thesis to predict symptomatic crises of a migraine sufferer using an MCC layout in an *eHealth* scenario for a better comprehension of modeling and simulation aspects. These techniques are implemented offline, before the system works autonomously. Predictive models are trained and then validated, both with real data. Models represent the behavior of the symptomatic crises based on input data: hemodynamic variables, environmental information, or events related to the disease. Different models are obtained in the training and validation steps. The bests of these models are tested (test step) and then used in the final stage (real time simulation and implementation).

Prior to modeling, it must be defined (i) what is going to be modeled, *i.e.* the output of the system which is, in our case, the objectification of the subjective pain of the migraine, and (ii) the metric or reference to select the best models. The output to be modeled is going to be presented in detail in Section 3.2.1.3; and we use the *fit* as quality criterion (defined in Section 3.2.2.1).

The theoretical background in Appendix A explains briefly the algorithms and maths behind the implemented models and simulations: (i) Three different family of algorithms have been implemented to create fine- and coarse-grained models of migraine crises: state-space algorithms, time series algorithms, and Grammatical Evolutionary algorithms (as an example of a heuristic approach). These models are mainly used in the RPSs. (ii) The most important subsystem (migraine prediction based on hemodynamic signals) is simulated using the Discrete Event Systems (DEVS) formalism. (iii) Fundamentals of decision making algorithms based on fuzzy logic are shown too. These algorithms are used in the Deciders as well as in the EDS. (iv) Due to the use of real data with data loss problems, active learning has been used to train the EDS too. (v) Finally, the machine learning classification algorithms used for prediction support based on qualitative data are described.

Before moving forward, to serve as a reminder, it worth to clarify briefly two concepts that will appear constantly in this and the next chapters. (i) First. A model is a representation that describes the behavior of a system—in our case, mathematical representations that describe a symptomatic crisis as an output of the system with respect to changes in some input variables. When it is said *predictive model*, it means that the model represents the system behavior time in advance, *i.e.* by the knowledge of current inputs, the systems explains future outputs. This lapse of time it is known as the *prediction horizon*, as stated in Section 2.2.2.3. (ii) Second. Model training is the search of the parameters of the models. This search can be supervised, unsupervised or semi-supervised. On the one hand, it is said that model training is supervised when parameters are

obtained iterative based on known examples and improving the results according to a metric with respect to the known (correct) solution. Eventually, it can be obtained the learning curve of the algorithm. On the other hand, an unsupervised model does not compute parameters based on iteration over a metric, but based on data structure or data distribution. The correct solution is unknown, and at the end, only one metric value is obtained. Semi-supervised learning is a kind of supervised learning that allows missing labels (values of the correct solution), modeling the way humans learn.

In the following sections the three subsystems in the CERPS are explained following the data flow starting from the data acquisition. For the specific case study of the migraine prediction system, there have been used three quantitative data types: biometric, atmospheric, and ambient variables (DAS 1, DAS 2 and DAS 3. Quantitative variables—as activity information or subjective symptoms—are represented by DAS 4.

3.2 Sub-system I. Fine-grained predictive modeling from hemodynamic variables

3 Implementation

3.1 Overview

3.2 Sub-system I. Fine-grained modeling

3.2.1 Hemodynamic data acquisition

3.2.1.1 Overview of the monitoring experiments

3.2.1.2 Conditioning circuits of hemodynamic signals

3.2.1.3 Pain objectification

3.2.2 Migraine predictive modeling

3.2.2.1 The training and validation blocks

3.2.2.2 Set of models

3.2.3 Improvement of the prediction and decider

3.2.3.1 Sensor-Dependent Model Selection System

3.3 DEVS-based migraine predictor simulator

3.4 Sub-system II. Prediction support

3.5 Sub-system III. Prediction support

3.6 System implementation: CERPS

3.7 Energy aware prediction system

Of the three sub-systems represented in Figure 3.2, this is the most important one. This sub-system (Figure 3.3) provides the most accurate prediction of a migraine event based on the information of hemodynamic data. In this Section, the different modules of the architecture described in Chapter 2 are implemented. This sub-system has been modeled, simulated and implemented. In addition, an energy efficiency study has been carried out. First, the **Data Acquisition System (DAS)** used for the acquisition of hemodynamic data is shown in Section 3.2.1. In Section 3.2.1.3 it is defined the objectification of the migraine pain, prior to its predictive modeling, which leads to the **Robust Predictive System** and is explained carefully in Section 3.2.2. Finally, in Section 3.2.3 it shown the **Decider** and methodologies to improve the system's accuracy.

3.2.1 Hemodynamic data acquisition

The relation between changes in hemodynamic variables regulated by autonomous nervous system and the migraine has been already discussed in the clinic literature. For instance, Hassinger *et al.* relate the cardiovascular response to the migraine [73]

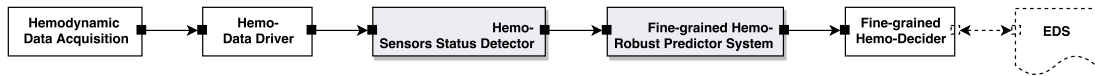


Figure 3.3: Architecture of the implementation of the Sub-system I for fine-grained predictive modeling based on hemodynamic variables. The decider can receive and use information from the expert system to improve its result, however this has not been implemented yet and that is why it appears dotted.

and Vollono *et al.* do the same with the heart rate variability during the sleep [180]. Kewman *et al.*, for example link changes in the skin temperature with migraines, as other authors do [90]. Passchier shows also changes in the electrodermal activity in migraine sufferers in [132]. Regarding the SpO₂, Lovati shows in [109] how blood oxygenation during sleep was significantly higher among headache patients with respect to controls. In this thesis it has been demonstrated that the hemodynamic variables are good predictors of the migraine, clearing out some doubts about the timing of changes in hemodynamic variables—migraine pain prediction observing the dynamics of these variables demonstrates that changes occur before the pain starts.

The aforementioned hemodynamic variables have been monitored in migraineurs during, ideally, 24 hours per day: heart rate (HR), electrodermal activity (EDA), skin temperature (TEMP) and peripheral capillary oxygen saturation (SpO₂). In addition to the hemodynamic variables, the subjective pain has been manually registered by patients to correlate the real pain with the biometric signals and to train the predictive models.

Along the development of this thesis, due to the non-availability in the market of devices with the required sensors, they have been used different devices to acquire the data. There were used three commercial devices and a prototype developed for ourselves; they are shown in Figure 3.4.

The Shimmer2r¹ monitoring platform was the first used. These platforms allows programming based on the operative system for embedded devices TinyOS² whose codes are written in nesC (a variation of the C programming language). Figure 3.4a shows the assembly for the monitoring of ECG. Shimmer devices provide an external input to add sensors; the skin temperature and the electrodermal activity where gathered in this way. To monitor the oxygen saturation it was used the OnyxII³ wireless device (Figure 3.4c): a finger pulse

¹Shimmer: <http://shimmersensing.com> (accessed January 2018)

²TinyOS: <https://web.archive.org/web/20050613075906/http://www.tinyos.net/> (accessed January 2018)

³Onyx II: <http://www.nonin.com/Onyx9560> (accessed January 2018)



(a) Shimmer 2r platform.

(b) BioPlux platform.

(c) OnyxII wireless device.

(d) Own prototype platform.

Figure 3.4: Different monitoring devices used along the implementation of the methodology to gather hemodynamic data and run fine-grained predictions.

oxymeter that transmits data via Bluetooth. This device bases on the OEM-III⁴ platform, both belonging to NONIN ®. Its small battery (around 400mA) and the complex assembly of different sensors made it not suitable for ambulatory monitoring. It was then when the PLUX-Wireless Biosignals⁵ was used. This device was used for a long time; it was not developed for continuous ambulatory monitoring lasting several weeks, and they suffered permanently from cable breaks. The first decision to tackle this problem was to use the OnyxII to monitor SpO₂ instead of the wired finger pulse oxymeter they provided. It is a closed platform and it cannot be programmed.

Both, Shimmer and Plux, devices transmit data via Bluetooth to a smartphone, and this was one of the major concerns of these platforms. Bluetooth suffers from continuous disconnections and has a short transmission

⁴OEM-III: <http://www.nonin.com/OEM-III-Module> (accessed January 2018)

⁵Plux: <http://www.biosignalsplux.com/index.php/en/> (accessed in January 2018)



Figure 3.5: Patient wearing the monitoring system.

field, so that, communication breakdowns also occurred when patients separated momentarily of the smartphone during their daily activities. This was why eventually it was decided to design our own prototype (called BrainGuard, as the clinical project was called). The multi-variable monitoring prototype in Figure 3.4d was developed jointly with the company M2C⁶. This device sends data via 3G to a Cloud server. It holds the OEM-III for backward compatibility. It is fully programmable and has a long battery life that allows long lasting monitoring sessions. As can be seen its size is a problem to highlight—compared to a 2 euro coin—; despite it is suitable for ambulatory monitoring, patients complain of its ergonomics.

3.2.1.1 Overview of the monitoring experiments

The migraine prediction study starts in the hospital with the patient, the doctors, and the engineers of the research group. Once patients sign the informed consent (the protocol for the clinical study that was approved by the Local Ethics Committee of the hospital), the monitoring phase begins. The medical staff of the hospital selects the body locations where the sensors are placed. The study comprises two phases: (i) data acquisition and training (modeling), also known as offline phase, and (ii) the real-time prediction, or online phase (see Figure 3.6).

⁶M2C: <http://m2c-solutions.es/> (accessed January 2018)

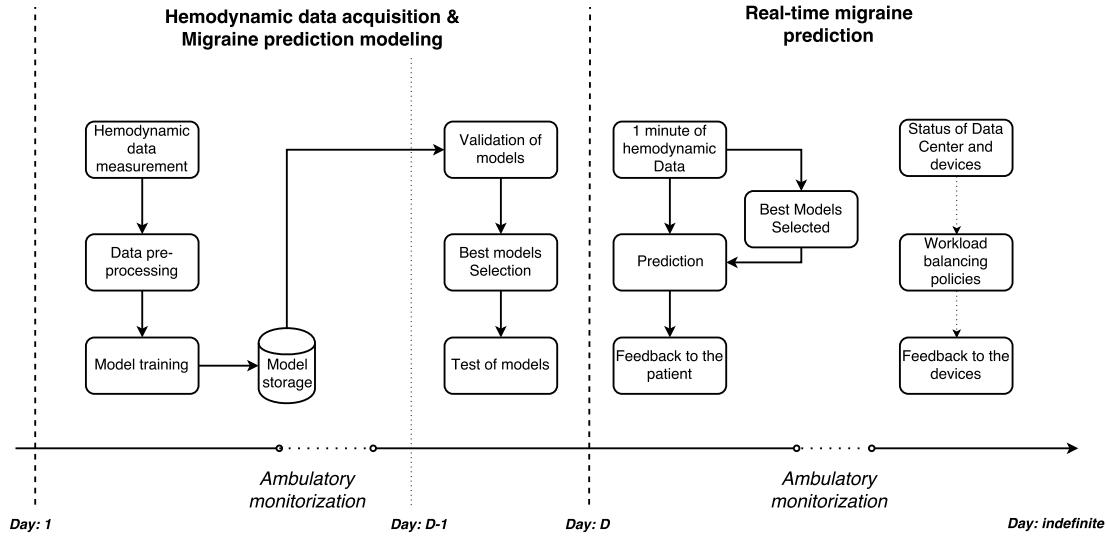


Figure 3.6: Scheme of the monitoring study. First, at the beginning of the study data are acquired, later the migraine prediction models are calculated. Data are gathered by the monitoring node and they might be transmitted from the node to the coordinator and then to the Data Center, or directly to latter.

The offline phase of the study lasts from 4 to 6 weeks. During this time the patient’s migraines are recorded and the prediction models are developed. The patients try to avoid any drug ingestion that can alter the monitored signals during the monitoring process. This is crucial in order to register the real body response before and during a migraine crisis. The patient will be allowed to take some medication if a strong migraine occurs. In these cases, the hemodynamic variables and the pain evolution are also collected; nevertheless, how the real pain would end without the effect of medication will be unknown, due to the effect that medications will accelerate the end of pain.

In spite of the different monitoring devices used, the data acquisition parameters remained the same for all of them. Skin temperature, EDA and ECG signals were all analog signals acquired as it. The SpO2 was digitally retrieved from the NONIN Onyx II and the OEM-III modules. Table 3.1 summarizes the placement of sensors, their data acquisition rate, accuracy, and the amount of data gathered during 24 hours of monitoring. Despite the HR is used for modeling, this is calculated offline from the ECG signal; this fact reduces the amount of data to process from 31.4 MB to 0.51 MB per day.

Patients indicate through an electronic form in an Android smartphone app developed in the research group the evolution of their pain and many other events related to the migraine crisis, such as premonitory symptoms, auras, triggers, concomitant symptoms and postdromic symptoms. All these data are used later for the implementation of the **Expert Decision System**. Figure 3.7 shows a snapshot

Table 3.1: Data acquisition parameters.

	Placement	Sampling (Hz)	Precision	Data-24h (kB)
TEMP	Armpit	1	0.0223 °C	126.6
EDA	Arm	1	0.0062 μS	126.6
ECG (HR)	Breast	250 (0.1)	4 ms (1 bpm)	31640.6 (12.7)
SpO2	Finger	3	1 %	253.1
Total (MB)				31.4 (0.51)

of the app of a migraine example to monitor the evolution of pain.

3.2.1.2 Conditioning circuits of hemodynamic signals

In this subsection it is going to be explained the algorithm developed for the extraction of heart rate from ECG data.

Data from analog sensors is provided as the tension measured in its terminals V_{meas} . Temperature and EDA sensors can be mounted in a Wheatstone bridge configuration. The temperature, in our case, is parsed using the standard equation of an NTC thermistor (Eq. 3.1). Making an abstraction of the actual electronic implementation, it can be said that:

$$\frac{1}{TEMP} = 273.15 - (a + b * \ln(R_{temp}) + c * (\ln(R_{temp}))^3) \left[\frac{1}{^{\circ}C} \right]$$

$$R_{temp} = R_{25^{\circ}C} * \frac{V_{meas,temp}}{V_{ref} - V_{meas,temp}} \quad (3.1)$$

For our WBSN, the NTC used is the model SA2F-TH-44031-40 from Omega⁷. R_{temp} is the impedance of the NTC thermistor measured at temperature $temp$. The constants for this sensor are:

$$a = 0.00102916, \quad b = 0.00023913, \quad c = 1.566 * 10^{-7}, \quad R_{25^{\circ}C} = 10k\Omega, \quad V_{ref} = 3.3V$$

The EDA represents the conductivity measured in *Siemens*. What it is measured is the voltage difference among two close measurement points of our skin, among which there exists an impedance R_{eda} :

⁷Omega: <https://www.omega.com/> (accessed January 2018)

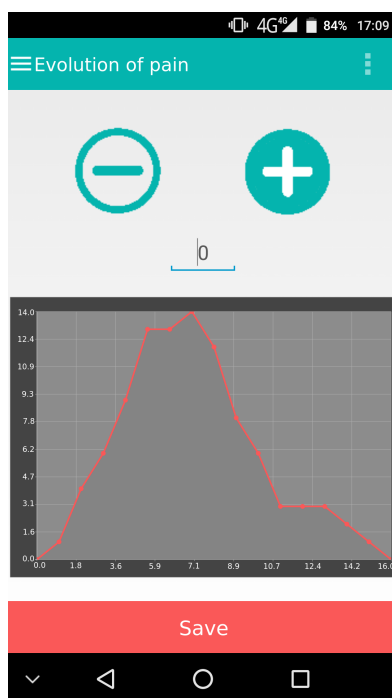


Figure 3.7: Evolution of the pain marked in the smartphone app. Notice that the ordinate indicates the subjective pain evolution, and the scale has no limits. The abscissa indicates the number of points marked, and in the sake of simplicity, they are not equally spaced.

$$EDA = \frac{1}{R_{eda}} [S] \quad (3.2)$$

$$R_{eda} = R_{refbridge} * \frac{V_{meas,eda}}{V_{ref} - V_{meas,eda}}$$

For our WBSN, they have been used two disposable medical electrodes and medical leads following the standard DIN 42-802. The constants for the EDA circuit are:

$$R_{refbridge} = 1.5M\Omega; V_{ref} = 3.3V$$

The electrocardiogram signal is measured using a single-lead local differential bipolar sensor, placed in the chest as shown in Figure 3.5. Prior to the HR calculation, the ECG is band-pass filtered with a 4th order Butterworth filter with $f_{c1} = 3Hz$, and $f_{c2} = 20Hz$. This, according to the power spectra of the ECG signal shown in Figure 3.8b [169], allows us to remove the DC component and retrieve a big amount of energy of the QRS complex—the signal of interest to calculate the heart rate—removing a lot of electromagnetic noise

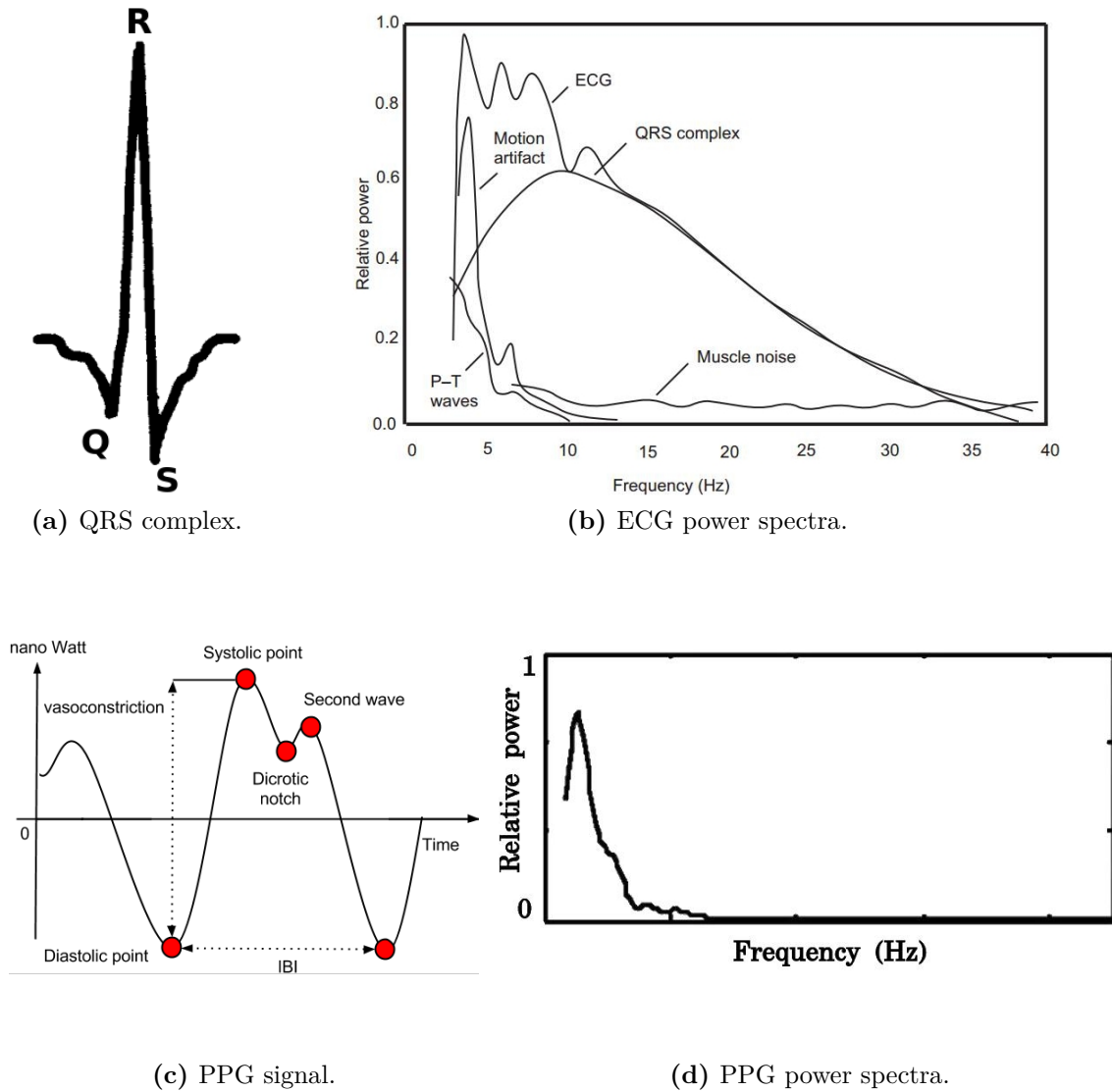


Figure 3.8: (a) QRS complex, the signal of interest to calculate the heart rate; (b) ECG power spectra. Notice the bandwidth of the QRS complex; (c) Systolic and diastolic points, and inter-beat-interval; (d) PPG power spectra. Notice the bandwidth of the QRS complex.

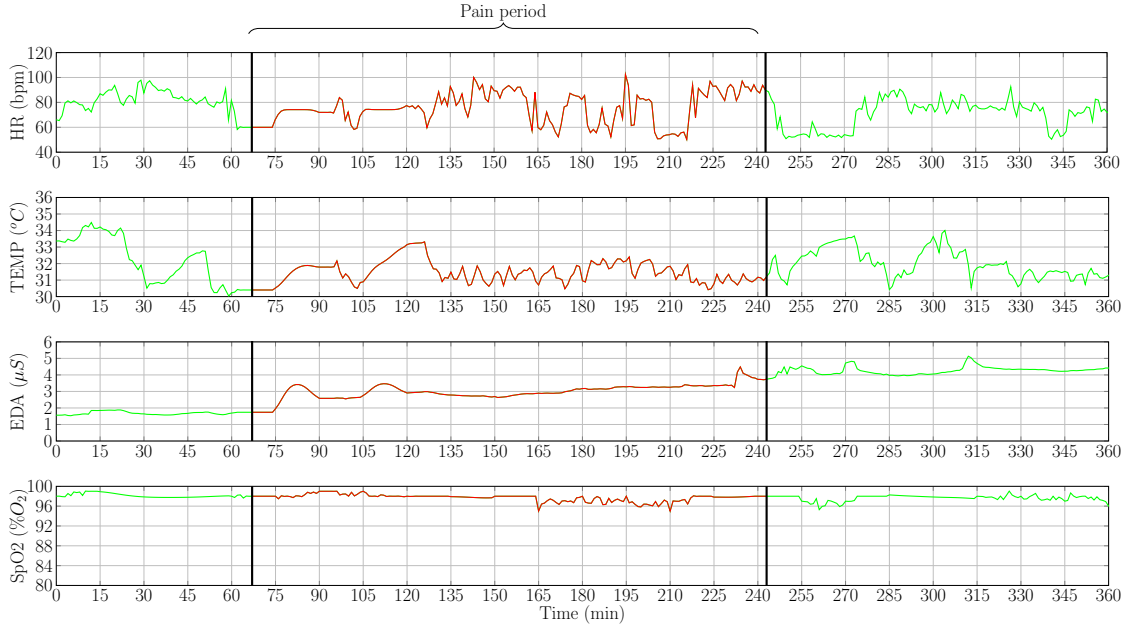


Figure 3.9: Hemodynamic variables after synchronization and preprocessing during a migraine episode (red curve between vertical bars).

coupled to the signal.

The HR is calculated counting *beats per minute* in the ECG signal. *Beats* are the *R* peaks in the QRS complex. The HR for one minute is the average value of the peaks counted in five windows of 20-second length each with 10 seconds overlap. The HR is smoothed to remove spurious samples according to the age of the patient using the regression Eq. 3.3 proposed by Tanaka *et al.* in [167]:

$$HR_{max} = 208 - (0.7 * age) \quad (3.3)$$

Peak miss-detection as well as over-detection can be corrected by mean of the Inter-Pulse-Interval (IPI) measurement as proposed in [152]. This improvement should be taken into account in further implementations.

The SpO2 signal is digitally acquired from the OEM-III. It can provide different time and level resolutions. This module provides data once a second through a serial communication. Photoplethysmography (PPG) is provided too if needed (Figures 3.8c and 3.8d). It is acquired at a sampling rate of 75 Hz. Synchronization of PPG and ECG can lead to the extraction of blood pressure information, but signal has not been used yet in the implementation proposed.

All variables have been calibrated using commercial medical devices. An example of a 6 hours monitoring is shown in Figure 3.9. Data are sub-sampled to

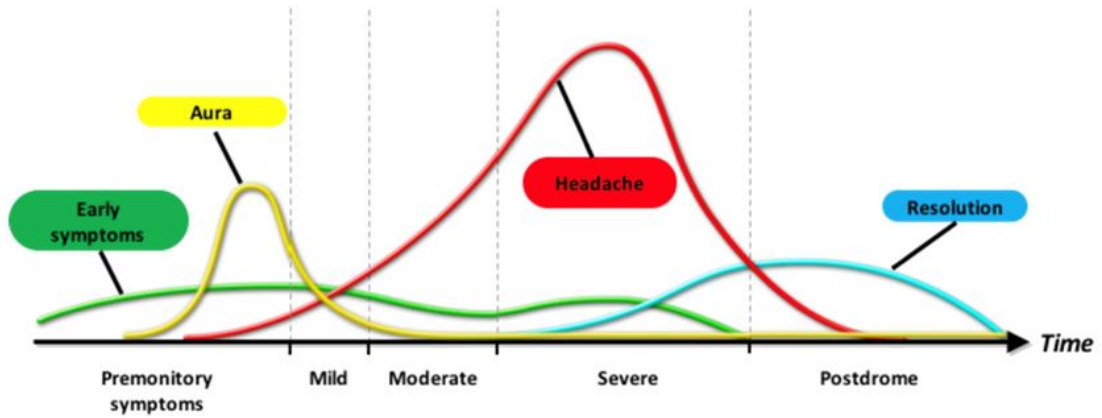


Figure 3.10: Migraine as a cascade of neurological phenomena. The red curve, the headache, is the pain to be modeled. Credits: Copyright © 2010 National University Hospital of Singapore.

1 minute period to work with them in the following steps. The figure marks in red, between the vertical bars, the pain period of a migraine episode. Before the pain period, it can be intuited differential changes in some of the hemodynamic signals such as heart rate and skin temperature. However—looking at different migraine recordings—it is not evident to say whether these changes happening prior to the onset of the pain must be increases or decreases of the variables. It is not evident neither when compared with the patient’s basal status. That is why there were contradictions in the clinical literature [123, 136], and thus, the proposed methodology contemplates the use of powerful techniques (such as state-space algorithms or metaheuristics), in contrast to static thresholding methods applied over absolute values of the signals.

3.2.1.3 Pain objectification

In this subsection it is going to be explained the objectification of the migraine subjective pain. In health, a symptomatic crisis means pain. Pain is subjective and, in order to objectify the patients’ pain it has been designed an objective function.

A migraine is a cascade of neurological phenomena [25]: (i) premonitory symptoms, (ii) aura phase, (iii) pain, (iv) finally a postdromic stage. The pain is just one of the latest ones, but the most noticeable, as it is shown in Figure 3.10. Evaluate a pain event after it occurs with a single-number is not useful if it is desired to relate changes with time-varying signals—such as hemodynamic signals. It makes necessary to continuously evaluate the pain of the patient to model the migraine with changes in hemodynamic variables. Regarding to this,

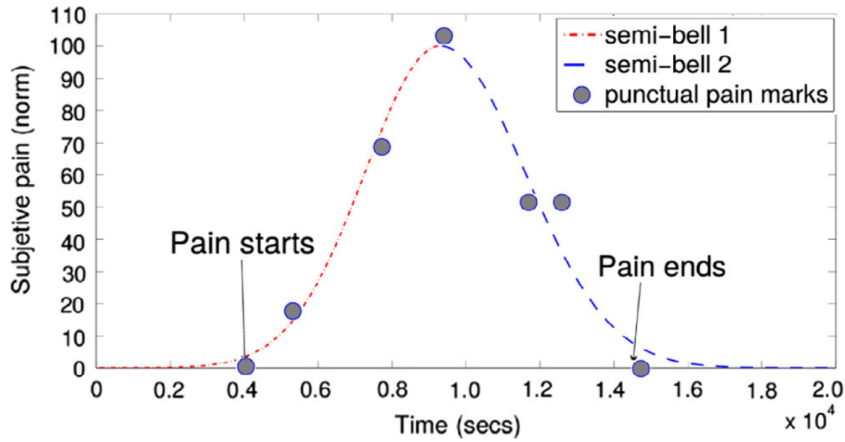


Figure 3.11: Modeling of subjective pain evolution curve using real data. The pain has been described as two semi-Gaussian curves.

in the experimental design of this problem, it was asked to patients for the preferred way to evaluate their pain during a migraine crisis. It was concluded that: (i) each patient has its own scale, and graphical and traditional numerical scales are not always useful for them. (ii) Single values are not practical and, to predict migraines, it is necessary to create a continuous pain curve. (iii) Upper bounds limit the option to indicate higher values once the highest has been reached. This happens because the maximum it is never known till the end of the event.

Each patient evaluates his or her pain in two ways: a global index of pain for the total migraine period, and punctual pain levels continuously marked during the migraine attack. For modeling we are going to focus on the latest. The punctual pain levels along the migraine attack have been chosen in an unlimited numbered scale where 0 is no pain, and in the evolution, the marked points grow and fall according to the perception of the patient. If the pain increases, the patient marks a positive number, negative if it decreases and 0 if it remains equal. Thus, a curve of subjective symptom evolution can be drawn. The maximum represents the highest pain, and it will be different for each migraine. The unlimited range responds to the ignorance of the patient who does not know if the maximum pain has been reached or not. With a limited scale, there could be several saturated points to the maximum of the scale along a period of time. With this method, consequently, the curve of its evolution must be normalized in amplitude. The intensity varies along a migraine episode, usually starting with low levels, but depending on the patient, it can reach its maximum intensity in a variable way; the same occurs with its disappearance. Therefore, the pain curves must be individualized for each patient.

To predict the symptomatic crisis, the first step is to generate a model of the migraine pain. To do this, an adjustment process of the registered subjective pain curve was carried out during the experiments. The main thoughts about how the curve could be pointed out to the Central Limit Theorem (CLT), where the addition of several random independent variables tends towards a Gaussian distribution. The migraine pain, seen as a combination of independent (*pseudo-*) random factors—most of them unknown—could be described as a Gaussian curve. It is known that the pain raises faster than it recesses, so, the symptomatic curve has been modeled as two semi-Gaussian curves, as they fit the patient’s subjective response. In addition to the punctual point of the pain evolution, patients also indicate two timestamps during the migraine attack. The first timestamp indicates the onset of the pain when detected, and the second one indicates the end of pain. With these two points and the punctual points of the pain evolution, two semi-Gaussian curves can be generated, as can be shown in the example in Figure 3.11. $\{(\mu_1, \sigma_1), (\mu_2, \sigma_2)\}$ are the two semi-Gaussian’s parameters necessary to define a symptomatic curve. The symptomatic curve includes the pain period, as it reflects some changes in the migraine process. An example of the resulting function is shown in Figure 3.11 using real data.

3.2.2 Migraine predictive modeling

Now that it has been described how the system’s objective function is, it is necessary to train the system to identify relations between the inputs signals and the output. This is called modeling, and because our system tries to anticipate future statuses based on current and past information, it is going to be referred as predictive modeling.

In this Section lets describe the creation of the Prediction System (PS). The results of the following discussions will lead to the architecture seen in Figure 2.7, and included in the block Hemo-PS of Figure 3.2. This section implements the Predictors and the Sensor Dependent Model Selection System in the Prediction System (PS) of Figure 2.7, hold in the Hemodynamic-RPS module of Figure 3.2. The Predictors that were seen as a black box that makes predictions, are going to be described based on some of the algorithms explained in Appendix A.

The objective of this section is to explain the modeling system created to predict the symptomatic crisis of a migraine sufferer. It is shown the offline implementation that leads to models and applications to be implemented online. Different blocks compose the modeling system in the training and validation stage. This stage provides a set of models to work with. The test stage will be

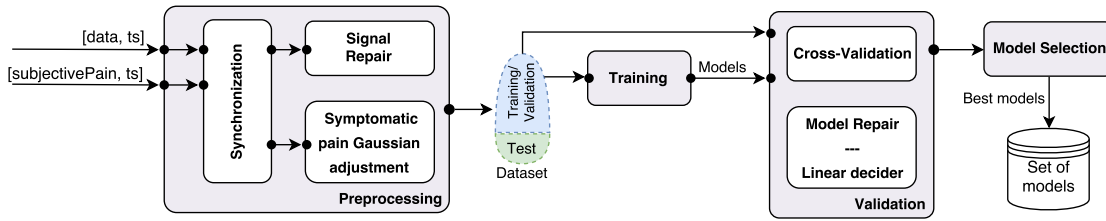


Figure 3.12: Training and validation diagram. As a result, a set of personalized migraine predictive models based on hemodynamic data are obtained.

presented in Chapter 4, as well as the results from the implementation of the final stage (real time simulation)—explained later in this Chapter.

The training and validation stage, shown in Figure 3.12, is the first step to find the model or models that describe the symptomatic pain curve of a migraine sufferer better. This stage runs offline and has as inputs the data recorded in the monitoring: hemodynamic variables and subjective pain marks for every migraine attack. The most important modules in this stage are: the preprocessing, the training and validation dataset, the training and validation modules and the batch of models achieved.

The preprocessing block parses data from sensors, detects errors and repairs them, and synchronizes data prior modeling. The preprocessing block in Figure 3.12 hosts different modules of the architecture. In this block, it can be found the **Data Driver** of the hemodynamic variables monitoring device, and the **Data Driver** of the smartphone app that collects, among other variables, the punctual points of the pain evolution (see Figure 3.13). As the WBSN can suffer failure of sensors or data can be lost due to Bluetooth disconnection or battery discharge, this block also hosts an adaptation of the **Sensor Status Detector** that detects and repairs broken data as stated in Section 2.2.2.2.

Data Drivers

Following the architecture shown in Figure 3.2, after the acquisition of the hemodynamic data, the feature extraction is made by the **Data Driver (DD)**. The WBSNs used provide data packets that encapsulate several variables with common timestamps, so the feature extraction is made first (on the contrary that the example shown in the architecture diagram in Figure 2.4).

Due the prediction of migraines is a novel field that has not been tackled before, it has been used the raw information without performing an exhaustive study of features—just the HR extracted from ECG—and it has been covered briefly using GE models. This problem will be addressed in a near future in this research. There have been used four variables: skin temperature (TEMP),

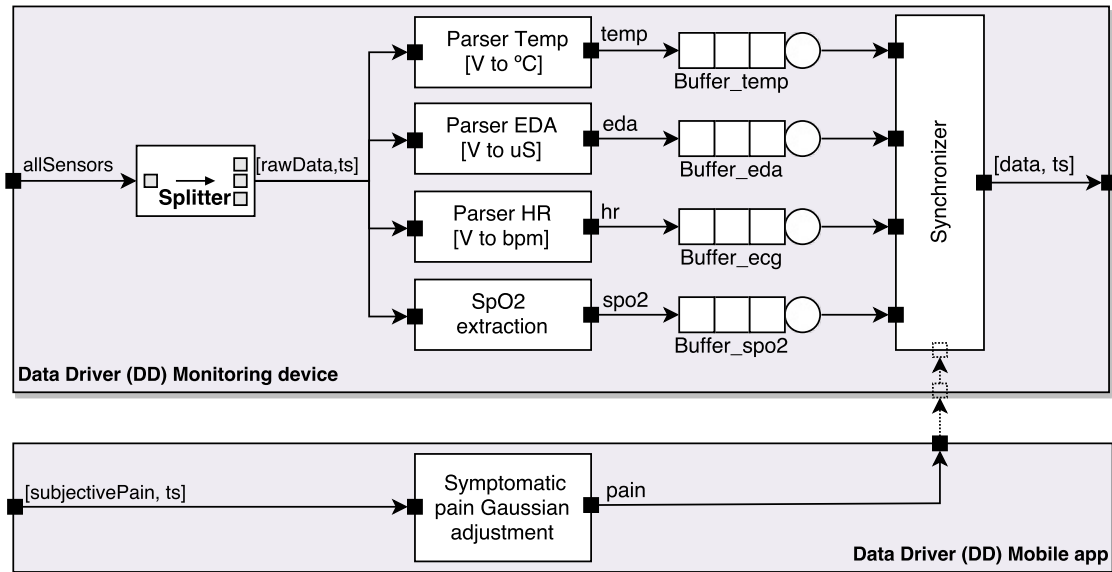


Figure 3.13: Data Driver implementation for the monitoring of hemodynamic variables and symptomatic pain. Notice that dotted arrows from the **Data Driver Mobile app** towards **Data Driver Monitoring device** implies that this connection takes place only during the training phase (offline); it will never be possible during the real time because the system is the one which provides the prediction of the pain.

electrodermal activity (EDA), heart rate (HR) and oxygen saturation in blood (SPO2). Data are parsed following the equations and algorithms aforementioned in Section 3.2.1.

During the experiment, when a migraine starts, patients are required to indicate it, and they are asked to indicate the punctual subjective pain till the pain ends. The subjective pain is gathered using a smartphone app developed in the research group. The value and the timestamps are saved and the symptomatic pain Gaussian adjustment is made leading to curves as the one presented in Figure 3.11.

It is now, once features are parsed and computed, and the pain curve—the objective function to be modeled—defined, when these are synchronized. This subprocess sets a common time interval between samples using a high-order FIR filter decimator. The time between samples is 1 minute for all signals and prediction horizons. This curve is then synchronized with the hemodynamic data.

Signal Repair

After the synchronization, data from hemodynamic signals are repaired. The system is identical to the one presented in the architecture for the **Sensor Status**

Detector (Section 2.2.2.2), however, this offline implementation does not raise alarms.

Each one of the three detectors represented in Figure 2.5 generates an alarm $a_i \in A$ when a condition occurs. $A = \{a_{noise}, a_{fall}, a_{sat}\} \in \mathfrak{B}$ represents the set of possible triggered alarms. \mathfrak{B} represents the Boolean space.

For the sake of simplicity, let's assume that decision makers are based on threshold detection. Each one of the detectors averages data during a period of time $T_i \in T = \{T_{noise}, T_{fall}, T_{sat}\}$, prior to the notification of an error to the **Anomaly Detector** block, avoiding, therefore, rebounds in alarms.

Raising an alarm a_{noise} to detect noise requires that the average energy exceeds a threshold τ_{ε_f} , as shown in Eq. 3.4 for a variable or feature f :

$$\varepsilon_f = \frac{1}{T_{noise}} \sum_{t=0}^{T_{noise}} f(t)^2 \geq \tau_{\varepsilon_f} \Leftrightarrow a_{noise} \quad (3.4)$$

Similarly, alarms to detect falls and saturation are raised when are below or exceed a threshold τ_{fall_f} or τ_{sat_f} respectively in different time windows T_{fall} and T_{sat} . Consequently, these conditions for a variable or feature f are described by Equations 3.5 and 3.6:

$$\bar{f} = \frac{1}{T_{fall}} \sum_{t=0}^{T_{fall}} f(t) \leq \tau_{fall_f} \Leftrightarrow a_{fall} \quad (3.5)$$

$$\bar{f} = \frac{1}{T_{sat}} \sum_{t=0}^{T_{sat}} f(t) \geq \tau_{sat_f} \Leftrightarrow a_{sat} \quad (3.6)$$

It is important to clarify here that, at this point of the system, we refer variables as features and *vice versa* indistinctly. During the implementation there is not a best option according to (i) detect sensor status, repair data loss if needed first, and then pre-process them (compute features) or, (ii) pre-process data and, after that, repair the data if needed. With the first option we can save energy and time processing if data are definitively lost; otherwise we would have preprocessed data that were damaged. The solution to this problem is a trade-off that has to be decided for each data source independently. As an example: noise detection is a hard task in variable ECG using simple detectors as the one described in Eq. 3.4; in addition consecutive signal repair of ECG is impossible for periods of time larger than few seconds. On the contrary, noise is easily detectable in heart rate.

Two different core algorithms have been implemented in the **Signal Repair** module. Each one of them with its advantages and inconveniences, and each one is suitable for different contexts.

- i) A Gaussian process machine learning (GPML) algorithm has been used in

those implementations where the `Signal Repair` module runs in moderate/high power computing infrastructures (as a server in a Data Center). This algorithm has been developed by Rasmussen [138], and the code is released under FreeBSD license⁸.

Gaussian processes provide a probabilistic approach to learning in kernel machines to detect patterns. Gaussian processes can be used for regression and classification problems. Gaussian processes represent distributions over functions instead of the traditional probabilistic methods that define distributions over individual data values. This is interesting to the analysis of time series of hemodynamic data, in which it is performed inference over functions—as Clifton *et al.* assets in [32].

Gaussian processes' hyperparameters have been personalized for each patient, and obtained (trained) with several hours of hemodynamic data for each one of them. GP performs interpolation over all the data, filling those gaps where data are not available because of noise, disconnection or sensor saturation. It is worth to notice that, the larger the gap, the lower the accuracy of the interpolation, tending it towards a constant value in some variables. This might be due to over-training or, on the contrary, due to a weak training. Whatever it is, it would require of a more precise study to draw a conclusion.

The training of hyperparameters as well as the interpolation of new data are processes that take computation time. Because of this, and despite GPs provide an accurate result with a probabilistic output, it was necessary to look for other methods to be run in light computing units such as a monitoring device.

- ii) Hemodynamic variables are controlled by the autonomous nervous system, and they are somehow interrelated, so that, in the absence data of one of them, the damaged data could be repaired though the information of the others. With this idea, the Autocorrelation Function (ACF) and the Partial Autocorrelation Function (PACF) were studied. Using the software Statgraphics Centurion XVII [160], it was seen rapidly that hemodynamic variables can be described by ARIMAX models—in its most general form.

Time series models are polynomial functions based on past data of the series. Time series models can be used to predict data by recursive iteration over the past data. Thus, it was decided to create time series models with exogenous inputs, so that, data loss in one variable could be predicted

⁸The Gaussian Processes Website. Available online: <http://gaussianprocess.org/gpml/code/matlab/doc/> (accessed January 2018)

temporary by an ARIMAX model where the exogenous inputs are the remaining hemodynamic variables (see specifications to the implementation in Section A.3). ARIMAX models are polynomial equations that can be executed in the most basic microcontroller embedded in a monitoring device. This makes this option more suitable for those MCC architectures when the `Signal Repair` runs on the monitoring device.

3.2.2.1 The training and validation blocks

After preprocessing the gathered data, a set of data is available for model training. This dataset is randomly divided into two groups: around 75% of the monitored migraine episodes (T) are chosen for training and validation of the models, and the remaining 25% are used to test the models. Each event has as much hours of data without pain before and after this as possible.

The algorithms chosen in the training and validation blocks for the predictive modeling of the migraine were the state-space based N4SID and the holistic Grammatical Evolutionary (GE) approach. In Section A.1 and Section A.2 the modeling backgrounds are explained. In addition, the specifications of implementation for our problem are explained there. In our case, there are the four hemodynamic inputs to create models over one output, the symptomatic pain. The metric used to evaluate the accuracy of the models is the aforementioned fit:

$$fit = 100 \times \left(1 - \frac{\|y - \hat{y}\|}{\|y - \text{mean}(y)\|} \right), \quad (3.7)$$

where y is the real symptomatic curve understood as shown in Section 3.2.1.3, and \hat{y} is the modeled one. $fit \in (-\infty, 1]$; the higher fit the better the model represents the behavior of the problem. Henceforth, this metric is used when we refer to the accuracy of the models/system, unless otherwise specified. There are several interesting metrics, but the fit has been chosen because it bases in the well-known Normalized Root Mean Square Error, whose normalization allows direct comparison between datasets and models with different nature.

In the training process, an optimization loop finds the optimal model parameters to maximize the fit, as shown in Algorithms 2 and 3. State-space models define immeasurable states to describe difference equations that calculate the current and future outputs from past and current inputs. N4SID is defined as a combined deterministic-stochastic model. Within the N4SID optimization loop, a study for feature selection has been performed. Algorithm 2 has been applied

for the four hemodynamic inputs, but also with the combinations in triads of them. In this thesis, the N4SID algorithm has been selected to solve this problem. The algorithm has been computed using the System Identification Toolbox of the MATLAB software [86].

This search of the optimum solution in a bounded problem might be tackled by brute force, as done using N4SID. However, this issue does not escalate linearly with the number of parameters to optimize. Furthermore, and this approach is not affordable more over when the number of parameters increases considerably. In these cases, heuristic approaches, such as Genetic Algorithms, make necessary to reduce the set of candidate solutions to evaluate. GE serves as feature creator as well, and it generates predictive models depicted by a mathematical expressions for the targeted model. All the details are explained carefully in Section A.2. In this thesis it has been used the HERO library published in [76] under GPL license. The HERO library compiles a Java code with the individuals (solutions) of each generation to evaluate them quickly. To reduce the number of compilations, the size of the population will be high but still allowing diversity of solutions in the next generations. The probability of mutation is the inverse of the number of rules. Wrapping is not allowed, *i.e.* if a solution is not decoded it will return a non-valuable mathematical function. The length of the chromosomes have to be sufficient to avoid this situation.

Training

For each migraine and algorithm, it has been trained a model. On the contrary to what traditional modeling studies do, each model is not calculated as the result of a multi-experimental training. In multi-experimental training a single model is created from several events. In our case, for each algorithm, the training block provides a batch of ($M = T$) models (one per migraine attack) to the validation block. It will be shown in Chapter 4 that a multi-experimental training does not provide statistically better results.

The issue stated above is a very important point of the methodology proposed. The pros of this methodology against a multi-experimental data are two: (i) for each model it can be optimized the prediction horizon individually. The quality of the data of a migraine event affects to the accuracy of the trained model for a given prediction horizon. So, events with good data suffer from the presence of bad data. These good events could lead individually to better models, but with the union of data, the multi-experimental model accuracy and potential prediction are diminished. (ii) Having several models trained with different prediction horizons, allows those models with higher predictive capacity support the others. For a

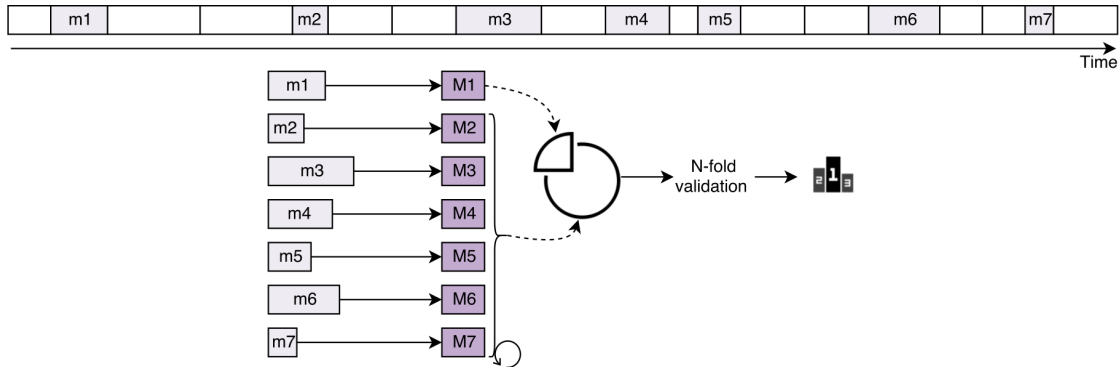


Figure 3.14: Cross-validation. For each patient, each migraine m_i lasts a different time. Each migraine leads to a new model M_i , and each model is validated against the remaining migraines. Eventually models are ranked and the best are selected to model the migraines of that patient.

desired prediction horizon, the accuracy of a model trained for a larger prediction horizon is higher than the accuracy of another one trained for a shorter horizon. The system takes advantage of this opportunity in the **Linear Combiner**.

Validation

Not all of these M models are going to be used in the final prediction stage; just the best ones are selected. The validation block looks for the best models to predict migraines using the a cross-validation criteria.

The nature of prediction problems in ambulatory *eHealth* MCC studies makes a difficult task to get long lasting continuous data records. This leads to separate migraines and basal data individually. This is possible because there exists, (i) inter-migraine independence: events are widely separated in time, and their basal conditions—periods without pain—may differ considerably; and (ii) intra-migraine similarity: for the same patient events resemble each other, because the autonomous nervous system is the same.

It is worth to remember that a migraine event, for training, includes as much hours of data without pain before and after this as possible, in order to capture the complete dynamic of the hemodynamic variables. With this, the cross-validation obtains each model M_i , $i = 1, 2, \dots, M$, from the i -th migraine, and the model is validated against the other j -th migraines, with $i \neq j$, as shown in Figure 3.14. The validations are made for different horizons, drawing a prediction curve. Each M_i model performs $M - 1$ prediction curves and then they are repaired. The models are ranked and the best are selected.

3.2.2.2 Set of models

The aim of the model selection is to make more robust predictions. It is supposed that just one model to predict all the migraines can underestimate or overestimate the input data. With this idea, a set of models is selected from the previously trained and validated batch. In the sake of simplicity, and to improve the readability, the block of *Improvement of the results* is explained in Section 3.2.3. Let us assume that this block does not change the validation result.

In the simple and traditional form to rank models, the average fit value is prioritized, setting a fixed value of \overline{fit}_{min} and observing the achieved prediction horizon. In the methodology proposed, there have been defined two different strategies to select the models: regarding to the fit achieved, or regarding the prediction horizon. This leads to a strategy for the selection of models.

(i) Given a prediction horizon: models are trained for a single prediction horizon and then these are selected by the average fit \overline{fit}_{min} achieved in the validation stage. (ii) On the other hand, given a requirement of minimum fit of quality, models that are trained for different horizons are selected according the level of compliance with the fit requirement achieved. A minimum average fit (quality) \overline{fit}_{min} is required. To rank the models these are sorted in relation to the average prediction horizon achieved in their validation for a similarity level of \overline{fit}_{min} . The best models are chosen.

The goodness of the fit and the prediction horizon can be used as criteria to select the models. Selecting one criterion automatically sets the other. Setting the fit, we can follow a more conservative approach that reduces the prediction horizon but improves the confidence in the models. However, setting the prediction horizon can achieve farthest predictions by loosing accuracy.

According to our experimental set, it can be considered that the selection one third of the models ($M_{best} = M/3$) works well. If it is not possible, a minimum $M_{bestmin} = 3$ models is considered good too. To avoid overfitting, it has been also considered that these models must validate at least other 3 migraines with a fit_{min} .

Linear combiner

Each one of the M_{best} models will generate a prediction; the prediction result will be calculated as a linear combination of all of the M_{best} predictions performed in the **Linear Combiner**.

In our implementation all models are equally considered and the linear combination is, actually, an average of the predicted values. It is a matter of future work to define a smarter way to weight models such that it can smooth high differences among predictions, for example.

3.2.3 Robust methodologies for improvement of the prediction and decider

Now that the reader knows how models are trained, and the details of the implementation for each algorithm, it is going to be presented the block of improvement of the results of Figure 3.12. It is worth to mention that most of these techniques are applied offline in order to improve the behavior of the system. When they are applied offline, all the data (past, present and future) are known. Nevertheless, its implementation in the real time has the limitation of the knowledge of future inputs. This Section leads to the **Decider** module shown in Figure 3.2.

This block, from the scheme shown in Figure 3.12, implements a sequence of processes to improve the prediction. This block must detect false positives and correct them. Hence, the aim is to detect and remove false positives in the predicted symptomatic curve during the validation.

The predictions obtained by the models have difficulties maintaining a constant value, and they tend to oscillate around the zero value when no symptomatic crisis is detected. This fluctuation causes an artificial reduction in the fit. As an example, the blue curve in Figure 3.16a represents a prediction with fluctuations (the original symptomatic curve is the black one). These oscillations can be easily detected and removed. To do this, two methods are evaluated: (i) reparation of the prediction, and (ii) Gaussian fitting as the original symptomatic crisis was modeled. These methods are applied to the our predictive modeling scheme. All the possibilities studied are shown in Figure 3.15. Each one of the four branches represents a scheme to improve the predictions.

Reparation of the prediction

The false positive correction works as follows in an iterative process as Algorithm 1 describes. In order to illustrate these processes, Figures 3.16a and 3.16b show how to repair a prediction.

False positives are detected using a level and a time threshold (th_{level} , th_{time}). Firstly, a time window of the predicted curve is selected. The length of the prediction horizon ph used for training the model. Then, those values out of limits (below zero and above the maximum) are marked with red x , as shown in Figure 3.16a. Then, negative values are set to zero, and the rest of outliers are set to the maximum. After that, the level threshold is applied. This process marks as candidates those values above 50% of the probability of occurrence,

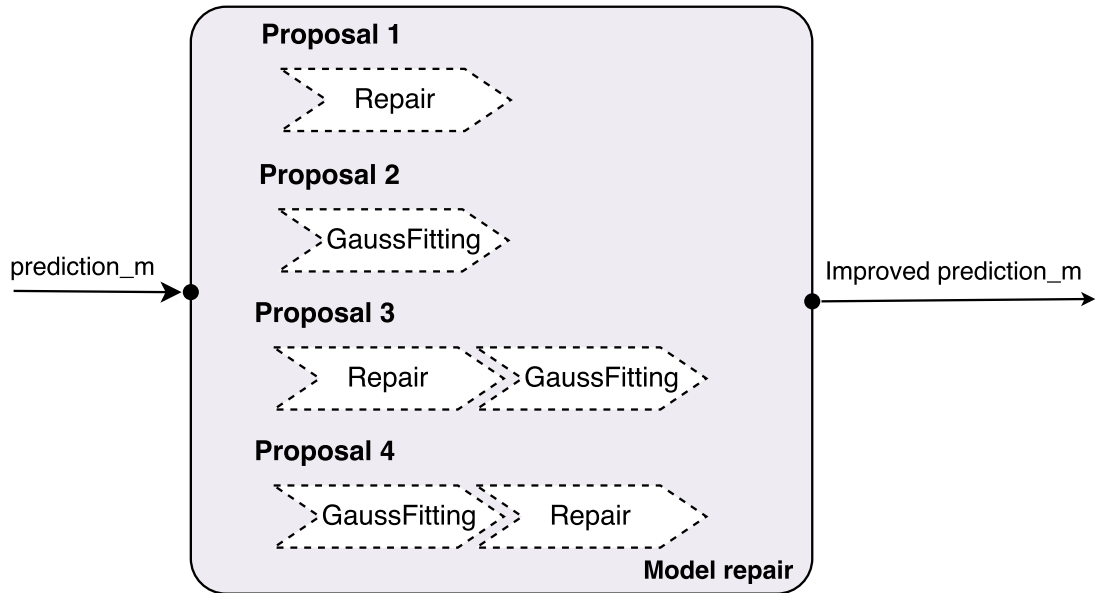


Figure 3.15: Schemes proposed to be used in the methodology to improve migraine predictions for each m -th model based on hemodynamic variables.

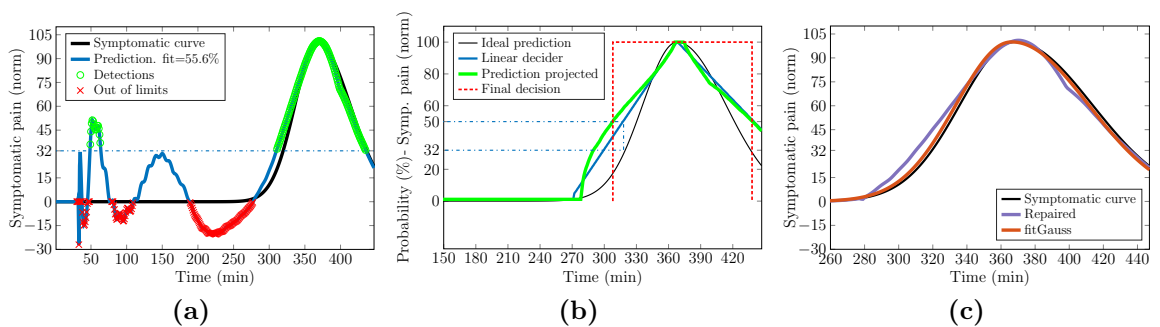


Figure 3.16: Improving the predictions. (a) Prediction over real symptomatic curve. Detection of events, removal of oscillations and false positive detected; (b) Probability curve of pain occurrence from the repaired prediction over the ideal probability curve. Final detection time limits; (c) Result after repairing the prediction and fitting the prediction to two semi-Gaussian curves.

$P_{50\%}$, using the linear decider explained below (green circles in Figure 3.16a). The 50% of pain probability is projected to a level of 32 over the ideal prediction (same as the original symptomatic curve). The blue dotted line represents this in Figure 3.16b and extends through Figure 3.16a.

Data: $\{O = \text{Predicted output}\}$
Variables: $\{th_{level}, th_{time}, ph\}$
Result: singleEvent = Prediction improved

Init: MarkA = 0; MarkB = ph;

Improvement of the prediction:

for \forall time windows **do**

Select time window: data = $O|_{MarkA}^{MarkB}$;
Limit the range:
 data(data < 0) = 0;
 data(data > 100) = 100;
Linear decider: candidates = data > th_{level} ;
Extend curve both sides towards 0:
 MarkA = min{O[MarkA:candidate[0]]};
 MarkB = min{O[candidate[end]:MarkB]};
 data = $O|_{MarkA}^{MarkB}$;
Detect events: events;
Time threshold: singleEvent;

end

Algorithm 1: Reparation of the prediction with false positive correction.

Finally, the time threshold is applied. Doctors consider that close events are rebounds of the same migraine, and must be considered together. If the distance between the farthest points is lower than 60 minutes (enough to detect if a migraine attack occurs or not), it is considered as a false positive. These points are removed. In the real time, to avoid multiple alarms, after a migraine detection, the system stops raising alarms for the time the threshold lasts. This action avoids make the patients tend to intersection. In Figure 3.16a the left detection is removed.

As a result, the repaired prediction is represented in Figure 3.16c (purple curve). It is worth noting that the fluctuation that appears in the middle of the curve was not detected by the threshold level.

Linear decider

At this point, the criteria for considering a pain intensity as migraine crisis must be defined; this decision is performed during the model repair sub-process.

The linear decider sub-process is such a decision sub-process, which will detect a migraine event when the probability of occurrence of a migraine episode exceeds a threshold.

The linear decider will detect a migraine event when the probability of occurrence of a detection exceeds the 50% of probability. This a worst case study; it is the latest probabilistic time the system can raise the alarm. This linear decider (blue triangle in Figure 3.16b) ranges from 0% (minimum pain intensity in the normalized symptomatic Gaussian curve) to 100% (maximum pain intensity in the normalized symptomatic Gaussian curve) of probability. Therefore, the linear decider projects the repaired prediction (blue signal in Figure 3.16a) to a probability of occurrence curve (green curve in Figure 3.16b). The linear decider uses a linear function as the projection function. As a result, the migraine detected (all those values higher than the 50% of probability of occurrence) is bounded by the red dotted line in Figure 3.16b.

This method improves the prediction horizon in several minutes for the best cases. In the worst case, the fit does not change. If this repair sub-process was not performed, all of the false positives would be included in the validation; the fitness may be low, and the higher prediction horizon, the faster the fit decreases. With this, some abrupt fitness breaks occur when the repaired model is not capable of detecting false positives. Nevertheless, this still remains at a higher fit than not using this model repair sub-process for the same prediction horizon.s

Gaussian fitting

Figure 3.16c also illustrates the result of applying the Gaussian fit (orange curve). This process fits the prediction to two semi-Gaussian curves, with reference at the maximum of the prediction. With the aim of finding the original bells, the prediction is first normalized and then fitted.

The impact of the combination of both processes (repair and Gaussian fitting, depicted in the two lower branches of Figure 3.15), is also analyzed in Chapter 4.

3.2.3.1 Hierarchy of models: The Sensor-Dependent Model Selection System

At this point, lets explain the Sensor-Dependent Model Selection System (*SDMS²*) introduced in Section 2.2.2.3. Our system is able to detect saturated or loose sensors by means of the **Sensor Status Detector**, and if a sensor fails it is able to select a different set of models not using that sensor.

Remember the reader that the system stores models according to fit or prediction horizon—and selecting one criterion automatically sets the other. So, the *SDMS²* can be seen as a pool of models that given the set of available

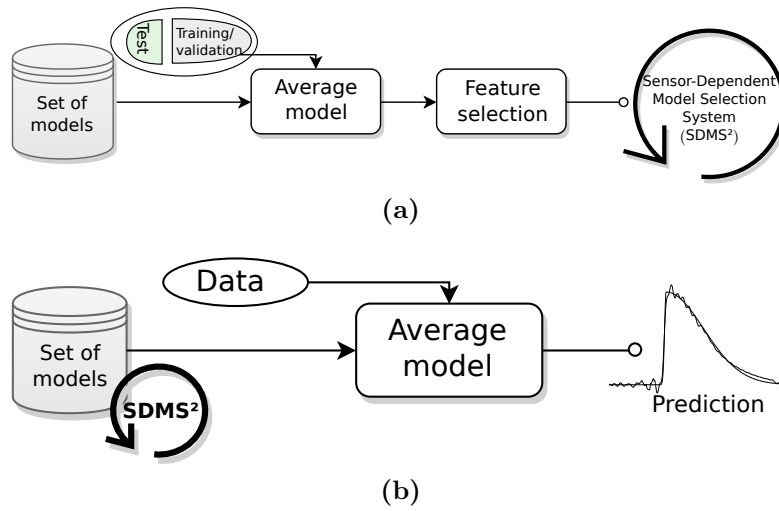


Figure 3.17: $SDMS^2$ design and usage for the real-time applications. Results using the SCS. (a) Sensor-dependent model selection system ($SDMS^2$); (b) Implementation of the system for real-time applications.

sensors, returns the best models according to a hierarchy based on fit or predictive horizon.

Figure 3.17 shows the design step of the hierarchy. All models are clustered in a hierarchy of sets of models, depending on the sensors/features the models use. The $SDMS^2$ senses the status of the sensors and chooses the best set of models according to their availability in real time (Figure 3.17b) and the desired criterion: accuracy (fit) or prediction horizon.

3.3 DEVS-based migraine predictor simulator system

3 Implementation

3.1 Overview

3.2 Sub-system I. Fine-grained modeling

3.3 DEVS-based migraine predictor simulator

3.3.1 DEVS formalization of the conceptual model

3.3.1.1 Coupled models

3.3.1.2 Atomic models

3.4 Sub-system II. Prediction support

3.5 Sub-system III. Prediction support

3.6 System implementation: CERPS

3.7 Energy aware prediction system

This thesis pursues the simulation of a Cyber-Physical System (CPS) to raise alarms for predictive modeling of symptomatic crises in chronic diseases, specifically, the migraine. To the best of our knowledge, this study is the first attempt to simulate a real device for the prediction of symptomatic crises.

Once the offline (*e.g.* in virtual, stand-alone mode) predictive modeling has been demonstrated, the next step is to test it in real-time. Prior to the expensive and slow hardware implementation of a complete prediction and monitoring device we proceed to develop a simulation system able to raise alarms and warn patients. The advanced simulator presented in this Section simulates a robust system against sensor failures that performs error signal detection and signal recovery (**Sensor Status Detector** in Figure 2.5). In case that the sensors are not available, it executes a hierarchical methodology of predictive models selection if signal recovery is not possible.

As it was mentioned, that a co-designed monitoring device had been developed in collaboration with the company M2C; our own prototype for data collection and real time experimentation with predictive models. However, our prototype does not have an alarm interface yet and, in addition, the access to patients for conducting the study is a difficult task. Before an actual commercial device is implemented in hardware, a hardware/software (HW/SW) co-simulation that includes hardware-in-the-loop (HIL) will be used. This will ensure that the system works in presence of actual hardware sensor failures and physical actuators, and triggers alarms accurately, as predicted by the simulation system. The specification of such a HW/SW co-simulation system is specified

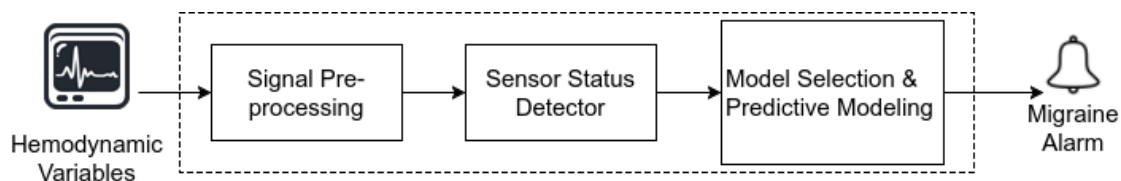


Figure 3.18: Conceptual system diagram. The dotted block represents the system.

using the Discrete Event Systems (DEVS) formalism that specifies unambiguous structure and behavior of any hybrid complex system.

It was decided to use xDEVS—published as Open Source under General Public License (GPL) [186]—because of the nature of the different predictive models that can be used (based on time series, GE or state-space algorithms among others). Furthermore, its semantics, performance, and its implementation into a hardware device is very straight forward, due to xDEVS is coded following the DEVS’ standards. Despite this simulation environment does not implement a GUI yet, as future work, it might be done implementing a Unified Modeling Language (UML) executable interface as described in [143].

It is going to be presented an incremental design with easy component substitution and rapid HW/SW swapping mechanism as previously shown through a DEVS-based transparent HW/SW modeling and simulation framework in [144]. In this Section it is implemented a DEVS-based model that will be the basis for the aforementioned HIL system. With this, it will be possible to start clinical experiments to inform patients when to take medications in advance followed by a study of the benefits of prediction in terms of complete or partial pain relief. This Section focuses on describing a top-down view of the advanced migraine prediction simulator system.

Conceptual design

The simulator has been implemented incrementally, leading to an advanced system where it can be simulated any kind of predictive model developed, frameworks of signal repair, or alarm generation system. It has been used DEVS for model verification. DEVS gives the formal specifications required to bring formal rigor to the modeling effort.

Because of the complexity of the system implemented in Figure 3.2, and the relevance of the fine-grained migraine predictive system based on hemodynamic signals for the system development, the simulation problem has been bounded to the fine-grained prediction based on hemodynamic data as shown in the conceptual diagram depicted in Figure 3.18. *Hemodynamic variables* and *Migraine alarm* represent sensors and a reactive device respectively, and can be

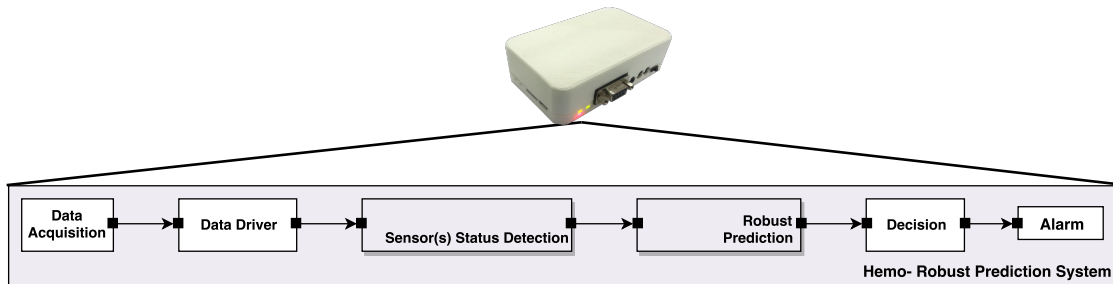


Figure 3.19: Representation of simulated system. All modules fit in the monitoring device, which also performs predictions and raise alarms.

easily replaced for HIL implementation. The *Sensor Status Detector* monitors the sensors operations and makes decisions if operations are below a specified threshold. *Model Selection* selects the set of operational sensors that continue to meet the operational requirements.

The aim is to demonstrate the robust methodology against sensor failures, the migraine predictive modeling and the generation of alarms in real time. Figure 3.19 shows clearly what the simulator encompasses: a MCC problem where the simulated device performs all tasks from data acquisition to local alarm generation without knowledge of external prediction from other sub-systems.

3.3.1 DEVS formalization of the conceptual model

We now describe the detailed migraine prediction system using the DEVS formalism.

Figure 3.20 represents the top view of the advanced migraine predictor simulator system. All blocks are described in detail Section 3.5. A detailed explanation is not necessary for a comprehensive understanding of this implementation, but main specifications are shown in subsequent sub-sections. Shaded boxes in Figure 3.20 represent coupled models, and there are seven of them divided into five types. Atomic models are represented with uncolored boxes, and there are twenty-five top-level atomic models divided into seven different types. Models surrounded by dotted lines are not part of the migraine prediction system, but they are required in the simulation framework. These will be removed in a real implementation of the system.

3.3.1.1 Coupled models

The five different types of coupled models in the system are the *EFsys*, the *EFgt*, the *SSD*, the *Predictor* and the *Graphs*. As the *RootCoupled* is the actual system,

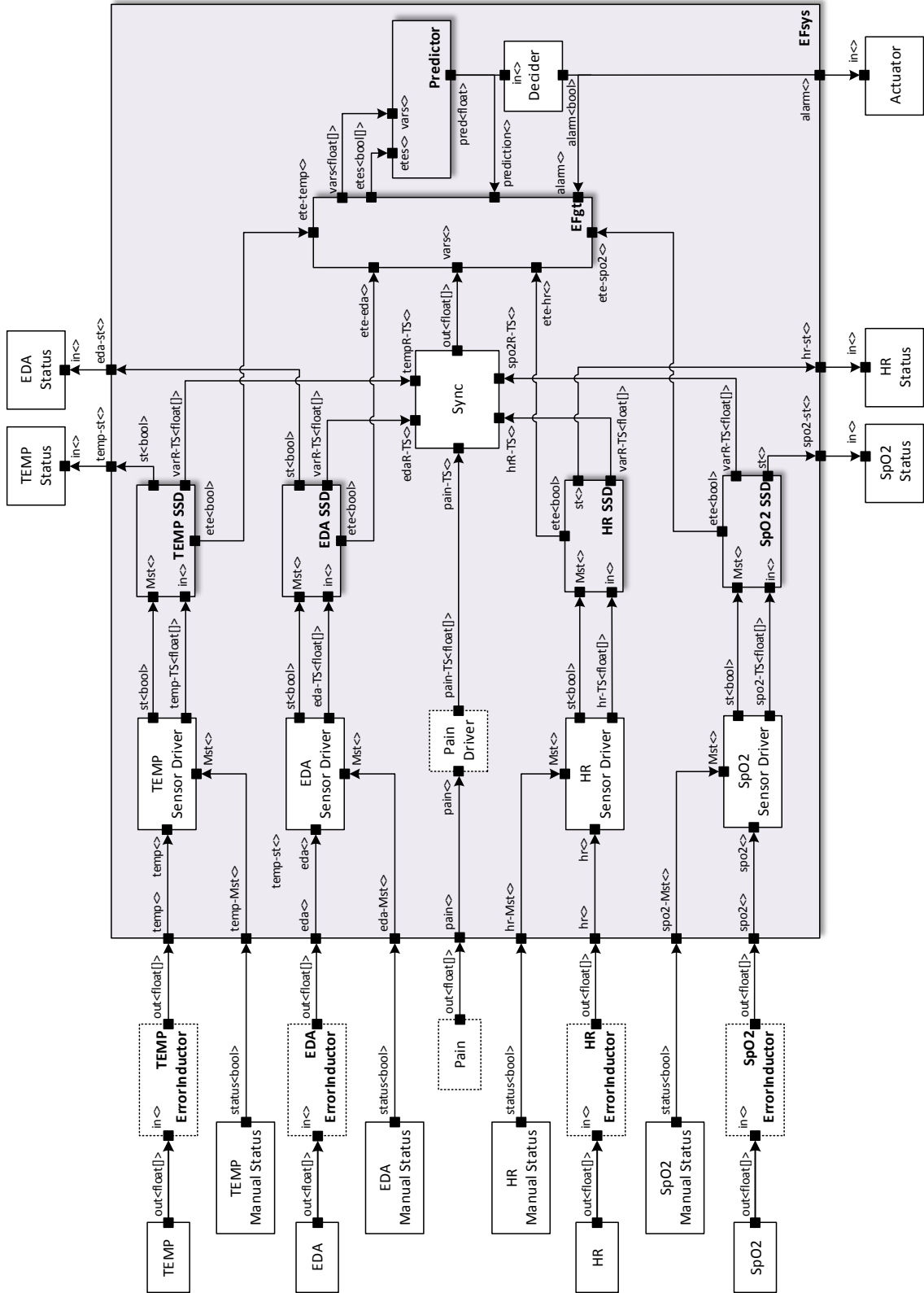


Figure 3.20: The RootCoupled model is the simulator framework containing all the coupled and atomic models in the system.

it is the entire Figure 3.20 itself. The coupled model `RootCoupled` is the simulator frame and interconnects the models that simulate the hardware modules, with the prediction system model in `EFsys`.

The `EFsys` (in Figure 3.20) hosts the remaining coupled models detailed in Figures 3.21a through 3.21c. The `EFsys` contains the intelligence of the system, which cannot be replaced by hardware. It also performs the processing from data of the single-output atomic models `TEMP`, `EDA`, `HR` and `SpO2` after data pass through the error sources `ErrorInductor`. The system also has as inputs other single-output atomic models named as `Manual Status` for each sensor, and gives an output to the single-input atomic models `Actuator`, and `Status` for each sensor as well.

The `EFgt` model (Figure 3.21a) contains two atomic models to control the data flow through model `G`, and to show simulation statistics through model `T`. If the simulator runs in simulated time, `T` activates the `stop` signal to finish the simulation after the simulation’s observation time has elapsed.

The simulator also has a coupled model (`Graphs`) that does not appear in Figure 3.20 for the sake of simplicity. `Graphs` has been included to improve the user experience. This model is only suitable for software simulation and plots the input data, the migraine predictions, the status of sensors, the manual resets of sensors and, the alarm event if it occurs.

The `SSD` is the `Sensor Status Detector` (see Figure 3.21b). This coupled model is able to detect three types of abnormal behaviors in signals: noisy signal, disconnection of a sensor (fall) and saturation. When an anomaly is detected, the `Anomaly Detector` raises an alarm signal `detect<bool>`. This alarm will indicate that the sensor has a problem and the core algorithm—implemented as a `GPML` algorithm—atomic model gets activated to recover the signal based on recent buffered data. The `Core-algorithm` model can perform a Gaussian Process Machine Learning, or a reparation using time-series models as `ARX` (both have been implemented, see Chapter 4). If the anomaly takes too much time to be removed, the `Core-algorithm` would not have enough recent data to estimate new values. In this case, when a predefined time is exceeded, the `ete<bool>` signal gets activated and the data from the sensor is not used for migraine prediction until a signal coming from the `Manual Status` indicates that the sensor has been restored. Then all alarms signals are disabled and `SSD` relays the data from the sensor without errors.

The `Predictor` (in Figure 3.21c) is the last type of coupled model, and it contains the migraine prediction models trained. It holds several atomic models. The `SDMS2` atomic model selects, subject to availability, the group of predictive models to perform the prediction. Each one of the atomic models

`Predictors*`—where `*` indicates the set of available sensors—computes several predictions (three in our case) and sends them to the atomic model `Linear Combiner`, which performs a linear combination of the three results.

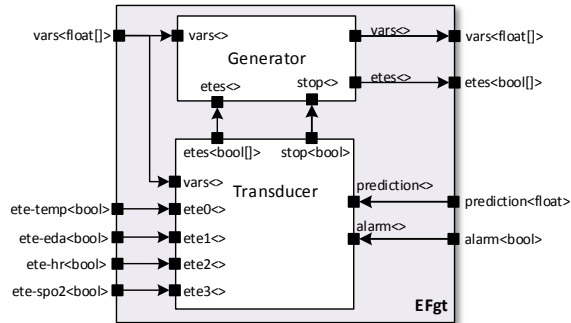
The simulated system has been tested using state-space and GE models for the migraine prediction; any of them, or others, can be used in the system. The `SDMS2` applies a hierarchy of models according to the availability of sensors to maintain prediction accuracy as it was shown. If the monitoring and prediction systems detects anomalies in sensors, the Sensor Dependent Model Selection System chooses an appropriate set of models that avoids the use of a damaged sensor and maintain an accuracy level for a given prediction horizon. Prior to changing the set of models, the system computes statistical averages to estimate lost sensor's values. If the failure in sensor exceeds a pre-defined wall clock interval, the `SDMS2` will choose it definitively.

As future work, it has aimed to provide an advanced feature of the system that will allow the injection of re-trained migraine prediction models (`Predictors*`) in real time. Then, the behavior of the system will be shown using variable structure DEVS [82].

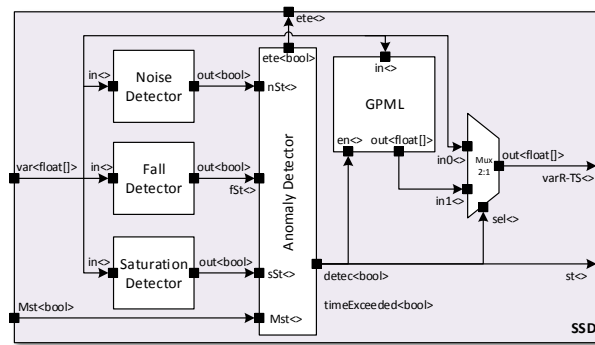
3.3.1.2 Atomic models

Among the twenty-five top-level atomic models, we distinguish seven different types: sensors and stimulus (`TEMP`, `EDA`, `HR`, `SpO2` and `Pain`), statistical error inductors (`ErrorInductor`), manual reset model for sensors (`Manual Status`), sensor drivers (`Sensor Driver`), signal synchronizer (`Sync`), an alarm evaluator (`Decider`) and the migraine alarm Actuator. All atomic models in the `EFsys` model represent HW modules that can be replaced by real HW components in a future HIL implementation.

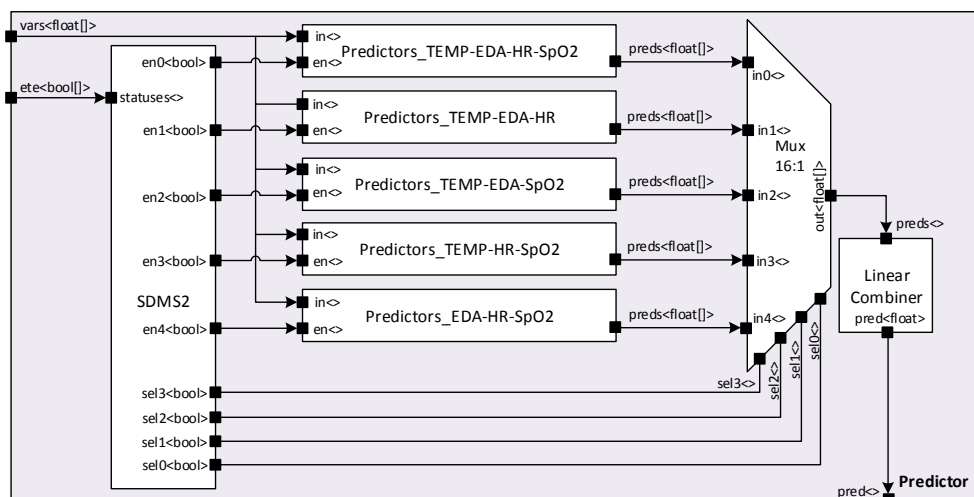
- The atomic models: `TEMP`, `EDA`, `HR` and `SpO2` pre-process biometric variables in their current implementation. In the future release of the prediction system, these models will provide the raw data from sensors, and will also incorporate real-time processing models (in the `Sensor Driver` models after the `ErrorInductor` models). Atomic models for hemodynamic variables remain outside of the `EFsys` coupled model so that they can be substituted easily for hardware devices facilitating the execution of HIL experiments. `Pain` is the symptomatic pain curve modeled from pain mark levels indicated by patients as shown in Section 3.2.1.3. This atomic model does not belong to the system and has been added to compute statistics when the framework is used with known migraine events.



(a) Coupled model *EFgt* that controls the data flow in the system.



(b) Coupled models *SSD* that check the status of sensors.



(c) Coupled model *Predictor* that computes the migraine predictions.

Figure 3.21: Three of the coupled models that the *EFsys* hosts.

- The error inductor modules `{TEMP-EDA-HR-Sp02} ErrorInductor` are atomic models that do not belong to the system but are useful in the simulation system to induce random errors. These models generate three types of errors: noise, saturation and disconnections (or falls). Errors are generated based on signals' error statistics, and these statistics are used in SSD coupled models to detect the errors.
- The driver `{TEMP-EDA-HR-Sp02} Sensor Driver` adds a timestamp to the data from the `RootCoupled`'s clock. In the implementation, these models do not perform any action. In a future release, they might include the signal processing of raw data coming from sensors.
- The model `{TEMP-EDA-HR-Sp02} Manual Status` represents hardware that raises notifications when a damaged sensor has been repaired. The generated signal resets the alarms. This leads the `SDMS2` model to again select the set of prediction models using all the sensors. In an HIL implementation they will be replaced by buttons or something similar.
- The `Sync` atomic model synchronizes and buffers the data for simultaneously supplying the values for the four biometric variables (`Pain`, if possible) to the coupled model `EFgt`.
- The `Decider` is an atomic model that determines if prediction results in a migraine event or not. The `Decider` is implemented as a threshold crossing model with a single level only. The numerical threshold value is 32 (normalized units) in the normalized objective symptomatic pain curve and this represents 50% probability of the maximum pain level as explained in Section 3.2.3.
- The `Actuator` is an atomic model that can be substituted by a hardware device, most likely an acoustic alarm. In the simulation system, this is a dummy model and it does not perform any action.
- The models `{TEMP-EDA-HR-Sp02} Status` indicate when sensors have data errors. In the simulation system, these are dummy models and they do not perform any action. They will be substituted by stimulus such as LED diodes.

As it shown in Chapter 4, simulation provides a lot of feedback about the system implementation and the methodology described in our MBSE problem. This simulation has opened this research to other areas that escape from the objective of this thesis. As an example, it is being proved that the simulation

allows a very straight forward implementation in re-configurable hardware platforms (FPGAs using VHDL language) to test the system in a real device [75]—thanks to the easy modularity of DEVS systems, the simulation tool used, is implemented. As future work, it will be suitable to test re-configurable hardware techniques over FPGAs to change, for example, re-trained predictive models in run-time.

3.4 Sub-system II. Prediction support: coarse-grained modeling from environmental variables

3 Implementation

3.1 Overview

3.2 Sub-system I. Fine-grained modeling

3.3 DEVS-based migraine predictor simulator

3.4 Sub-system II. Prediction support

3.4.1 Environmental data acquisition

3.4.2 Migraine predictive modeling from environmental data

3.5 Sub-system III. Prediction support

3.6 System implementation: CERPS

3.7 Energy aware prediction system

As stated at the beginning of this Chapter, in this section it is going to be presented the Sub-system II. This is one of the two modules for prediction support sketched in this research. This Section describes the basic implementation carried out to hold the sub-system's architecture shown in Figure 3.22. This sub-system has a marginal contribution to the current research results, but it has been partially implemented as part of the whole system architecture previously shown in implCerps.

Sub-system II is conceived for the prediction support based on a coarse-grained prediction obtained from environmental data. The environmental variables such as room and outdoor temperature, pressure or humidity are considered as trigger or precursor variables of the migraine disease. This is a controversial topic between specialized doctors; so much so that it can be found in the literature a wide variance in the percentage of migraine sufferers that relate environmental factors an headache—from 7% [79] up to 61% [181]. There are also several apps for

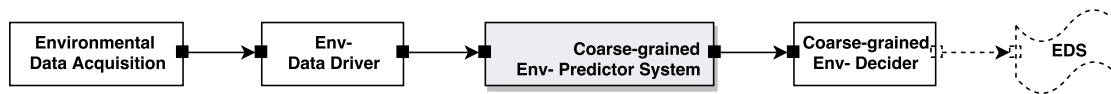


Figure 3.22: Architecture of the implementation of the Sub-system II for coarse-grained predictive modeling based on environmental variables. This sub-system provides information for prediction support to the **Expert Decision System**.

migraine management such as Migraine Buddy⁹ or Curelator Headache¹⁰ that contemplates weather conditions for their coarse-grained predictions. However it is, it seems that there is a reality that some migraineurs blame these to their pains. So, it has been considered that the information extracted for environmental variables would help the **Expert Decision System**, **EDS** to raise alarms and thus, we proceed to explain the implementation carried out following the structure of Section 3.5. Notice that this sub-system does not consider weather variable only, but environmental as well. This distinction adds, as an example, luminosity and noise, to the atmospheric ones.

3.4.1 Environmental data acquisition

As aforementioned, in addition to weather, there have been considered other environmental variables such as noise and luminosity. It has been distinguished between global—external—and local variables. Global variables refer to external weather variables in the geographical area of the patient. Local variables are those that occur within the exact indoor location where the patient is. (i) As global variables there have been considered the temperature ($^{\circ}C$), relative humidity ($\%RH$), pressure (hPa) and rainfall (mm). (ii) As local variables there are gathered the room temperature ($^{\circ}C$), relative humidity ($\%RH$), pressure (hPa), luminosity (lux) and noise level (W , dB). From them it can be derivated several features: average, maximum, minimum or differential levels within a period of time (not for *decibels*, they need to be converted to energy).

Global data have been obtained from meteorological Cloud services. Nowadays in addition to the national meteorologic agencies—such as AEMET¹¹ in Spain—there are thousands of private meteorological stations distributed all around the world. Some are part of global networks that submit the gathered information to open access databases. This is an easy, fast and cheap way to access to weather information not restricted to the specific locations provided by the national entities. In this research, to gather global weather information it has

⁹Migraine Buddy: <http://www.migrainebuddy.com/> (accessed February 2018)

¹⁰<https://curelator.com/> (accessed February 2018)

¹¹AEMET: <http://www.aemet.es> (accessed February 2018)

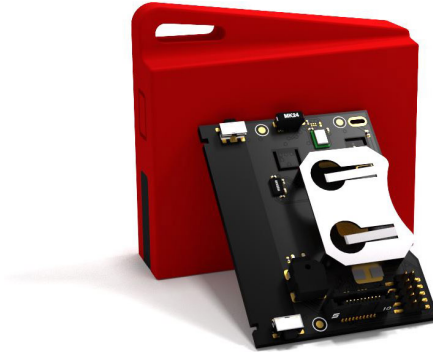


Figure 3.23: SensorTag CC2650 for local environmental monitoring. Device used to gather local environmental information for migraine prediction support.

been considered data recorded from the AEMET and the Cloud service Weather Underground¹². Figure 3.24 represents the variation of three external meteorological variables during 96 hours. Red lines indicate the pain of three migraines of a patient.

Local environmental data are gathered from the SensorTag CC2650 STK device (Figure 3.23), which is a commercial device of Texas Instruments¹³. Patients bring this device and leave it near them, and the device transmits data to the smartphone wirelessly via Bluetooth through a BLE interface. Figure 3.25 represents the variation of four local environmental variables one hour before a migraine episode and during the migraine episode. The episode lasts approximately seven hours. The red line indicates the pain period.

Table 3.2 shows the amount of environmental data gathered. The total amount of data (~ 3 kB) is small enough to be collected and used for a smartphone just in the case the coarse-grained prediction takes place on it.

Table 3.2: Environmental data acquisition parameters.

	Acquisition time (min)	Precision (bit)	Data-24h (kB)
Global weather information	60	Depends on the weather service	~ 0.4
Local Temperature	5	16	0.6
Local Humidity.	5	16	0.6
Local Pressure	5	24	0.8
Local Luminosity	5	16	0.6
	Total (kB)		~ 3

For the purpose of creating coarse-grained predictive models to help the EDS, the pain has been simplified to a binary sequence of 0/1 (yes/no) values that

¹²Weather Underground: <https://www.wunderground.com/> (accessed February 2018)

¹³Texas Instruments: <http://www.ti.com> (accessed February 2018)

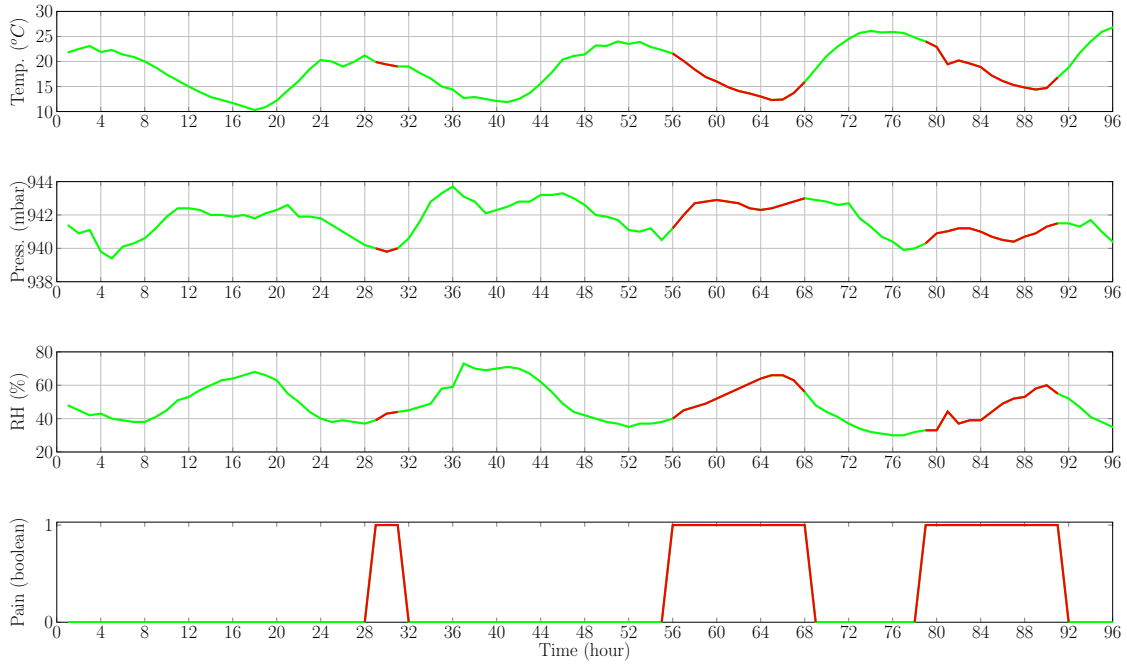


Figure 3.24: Global meteorological variables before and during three periods of pain of a migraineur (indicated as 1 in the bottom graph, and drawn with red lines).

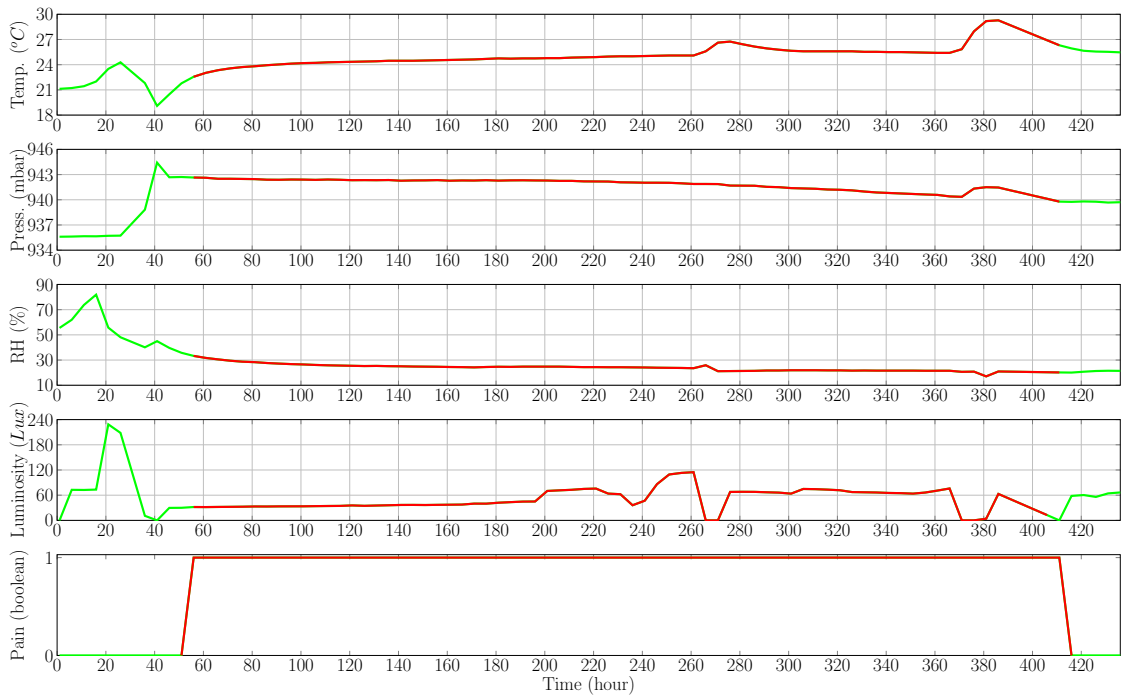


Figure 3.25: Local environmental variables before and during a migraine episode (indicated as 1 in the bottom graph, and drawn with red lines).

indicate if pain exists, or not. To do this, it is enough to know when the pain starts and ends. Patients indicate this in the aforementioned Android smartphone app developed in the research group. All the data are collected and merged by the smartphone. During the training phase, data are sent to a Cloud server. In a real time implementation, the predictive modeling can take place either the smartphone or the Cloud Computing infrastructures.

3.4.2 Migraine predictive modeling from environmental data

Environmental data have been considered for a simple purpose of serving as prediction support in the decision system. Data has been used in two ways: (i) to create coarse-grained migraine prediction models using the slow global weather variables, and the faster local environmental variables; (ii) to create classification/discrimination models (prediction horizon equals 0 minutes) using the local environmental variables.

The predictive methodology used here is pretty much the same the one used in Section 3.5 for hemodynamic variables. Models have been trained per migraine, validated as shown in Figure 3.14, and then the best models have been selected to create average predictions. Just few considerations must be taken into account. The following lines draw the most important differences undertaken in the implementation of this proof of concept. Due to not all the migraineurs relate weather conditions to the pain, models have been trained per person as well.

Following the data flow in Figure 3.12:

- **Preprocessing module.** Disconnections and noisy data are fewer in the device gathering local and environmental variables than in the ambulatory WBSN. Nevertheless, when failures are detected, similar implementations of the signal repair module are run. In addition, global meteorological variables have few lost values in the sequences. When this occurs, because they are slow variables, simple interpolations may solve the problem.
- **The training and validation blocks.** Due to the output of the system is not a curve, but a dichotomous simplification of the pain, the continuous predictive models used for the fine-grained migraine prediction sub-system, are not suitable. Instead, there have been implemented classic and well known machine learning algorithms. There have been studied (i) the decision tree-based *Random Forest* algorithm, (ii) the *k-means* clustering algorithm, and (iii) the Multilayer Perceptron (MLP), a kind of feedforwarded artificial neural network. These algorithms have been

computed using their implementations in the WEKA¹⁴ [70] data mining tool (using 10-fold cross validation and default parameters).

As it was previously mentioned, models have been trained for detection and prediction.

- **Detection.** Detection proves the direct consequence of the effect of the environmental variables over the pain of the migraine, and the prediction horizon is, of course, 0 minutes. This proof can only be performed over the local environmental variables because they are fast enough to be correlated with the hemodynamic variables.
- **Prediction.** To use the aforementioned machine learning algorithms, the classification variable, the pain, must be brought forward h minutes up to current time k . Thus, the training vectors $[inputs_k, pain_{k+h}]$ are correlated for the prediction horizon h .

For the sake of simplicity, (i) models using local environmental features have been trained for a fixed prediction horizon of 20 minutes (on the average of the best prediction horizon achieved for the fine-grained prediction). It has been considered, at least, 60 and 30 minutes of data without pain before and after the event respectively. In addition, (ii) Models using global weather information have been trained for prediction horizons of 3, 6, 12, 24 and 48 hours.

- **Hierarchy of models.** It has been assumed that the availability and robustness of each one of the environmental data sources (independent sensors of the wireless SensorTag and weather Cloud services) is so high that it is not necessary to implement the methodology of model selection (*SDMS*²) introduced in Section 2.2.2.3.

The models have been tested over basal data—periods without pain—to measure the false positive rate. Chapter 4 shows briefly some preliminary results obtained in the implementation of this Sub-system II for prediction support.

3.5 Sub-system III. Prediction support: clustering classification from qualitative variables

3 Implementation

¹⁴WEKA: <https://www.cs.waikato.ac.nz/ml/weka/> (accessed February 2018)

3.1 Overview

3.2 Sub-system I. Fine-grained modeling

3.3 DEVS-based migraine predictor simulator

3.4 Sub-system II. Prediction support

3.5 Sub-system III. Prediction support**3.5.1 Acquisition of qualitative variables****3.5.2 Classification of qualitative variables**

3.6 System implementation: CERPS

3.7 Energy aware prediction system

This section describes the second module for prediction support Sub-system III. In the same way as Sub-system II, this has a marginal contribution to the current research results, but preliminary results further shown, demonstrate that information extracted from qualitative data have a powerful interest for future improvements of the predictive methodology presented in this thesis.

There exist qualitative or descriptive information that surround a prediction problem that cannot provide accurate prediction by their own, but they can be beneficial for an eventual final decision making task, to raise alarms, for example.

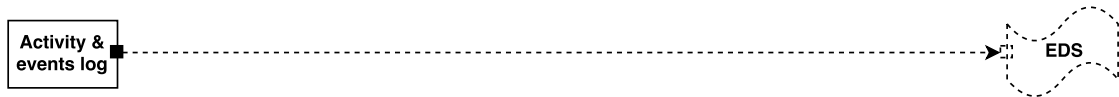


Figure 3.26: Architecture of the implementation of the Sub-system III for prediction support to the **Expert Decision System** by means of classification of premonitory symptoms and activity information.

In Section 3.5 and Section 3.3 it was shown a novel predictive system for migraine prediction based on hemodynamic, which is has been the first one described in the literature. However, in the state of the art, there are several contributions that relate symptoms suffered for the migraineurs with the pain, as well as other information regarding activity or life habits. Apps such as Migraine Buddy, or M-sense¹⁵, acquire information from activity and symptoms to identify triggers and warn of a possible pain in a near future. However, with this information they cannot provide accurate predictions and thus, their algorithms can act only as a diary.

In this section it is explained (i) the basic implementation and, (ii) the experimentation carried out to hold the sub-system's architecture (depicted in Figure 3.26). This offline implementation serves to identify those activity and symptomatic variables which will help the **Expert Decision System** in the

¹⁵M-sense: <https://m-sense.de/> (accessed February 2018)

Table 3.3: Groups of premonitory symptoms that can serve as prediction support for decision making.

Group	Symptoms
Alterations in moods	Euphoria, hyperactivity, sadness, apathy, irritability, anxiety
Cognitive disturbances	Difficulty writing, difficulty speaking, difficulty concentrating
Gastrointestinal/urinary disturbances	Nausea and/or vomiting, flatulence, constipation, fluid retention/swelling, increased urinary frequency (polyuria)
Alterations in appetite	Increase, decrease, appeal for specific foods, increased thirst
Sensory disturbance	Photo/phono/osmophobia, sensitivity to skin friction (alodynia)
Thermal disturbance	Feeling cold, feeling hot
Sleep disturbance	Insomnia, drowsiness
Others	Tinnitus, fatigue, dizziness, yawning, dermal/vascular changes, neck stiffness, blurred vision and others (open field)

decision making. Thus, in real time, the expert system is fed directly with the information from the most relevant qualitative variables.

3.5.1 Acquisition of qualitative variables

During monitoring of the hemodynamic variables, in the first stage of Figure 3.6 in page 50, patients also provide information relative to the onset—and maybe release—of premonitory symptoms (subjective and unspecific symptoms) and aura, before the pain starts. When the pain ends, they are encouraged to fill a questionnaire where they indicate if there was any activity (*e.g.* eating specific foods or sport activities) that they blame to the onset of the pain. All this information is stored with their timestamps.

Prior to the experiment, patients are trained to recognize most subjective symptoms related to migraine events. Using the already mentioned mobile app developed in the research group, patients mark subjective symptoms among the list shown in Table 3.3. As the reader may suppose, they also indicate the onset of the pain—if it occurs—thus it can be established a model that identify a premonitory symptom as predictive or not.

It may occur that patients indicate the end of a subjective symptom; however, this is very hard to identify, and as can be seen in migPhases, the premonitory symptoms usually accompanies the pain till the end. It may also happen that, despite having marked the onset of a symptomatic event, the pain never starts. In this case, the symptom resets after 72 hours, and it is considered

as a false positive event. If a pain occurs without premonitory symptoms, in the final questionnaire patients are allowed to indicate, *a posteriori* if they remember to have had a premonitory symptom and, in that case, indicate the approximate day and hour.

3.5.2 Classification of qualitative variables

In the research shown in this thesis it has not been studied the predictive effect of the premonitory symptoms, *i.e.* the effect of a symptom/s at time k to predict the beginning of the pain at time $k + h$. This is current research whose results will be discussed and published in a future work. However, it is an objective of this study to find—if it exists—the premonitory effect of premonitory symptoms, *i.e.* to find if premonitory symptoms are susceptible of indicating that a migraine is going to occur—it does not matter when ($\leq 72h$)—or not. For this purpose it has been trained a *Random Forest* algorithm using WEKA performing 10-fold cross validation and default parameters. It is also interesting to see which of the premonitory symptoms are more prone to be predictive. To do this, it has been performed a *Ranker* based attribute selection by mean of an *Information Gain* evaluation.

There is a clinical objective in this study that tries to correlate confidence of patients in their premonitory symptoms and the success effectiveness, to identify if those who are more prone to mark symptoms predict more or less migraines. This is interesting from the technical perspective too, thus it would be possible to weight the output of this Sub-system III according to the credibility on patients' subjectivity. In Chapter 4 there are shown the preliminary technical results of this sub-system and its utility to the Expert Decision System.

3.6 System implementation: Critical-Events Robust Prediction System

3 Implementation

3.1 Overview

3.2 Sub-system I. Fine-grained modeling

3.3 DEVS-based migraine predictor simulator

3.4 Sub-system II. Prediction support

3.5 Sub-system III. Prediction support

3.6 System implementation: CERPS

3.7 Energy aware prediction system

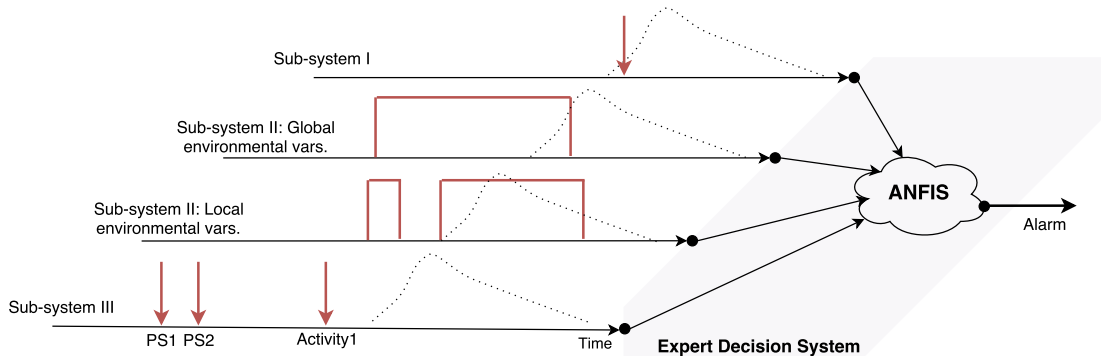


Figure 3.27: Expert Decision System implemented as a ANFIS model which receives data from Sub-systems I, II and III.

The Expert Decision System has been partially implemented. The assembly of all modules already presented and the implementation of the complete Critical-Events Robust Prediction System has not been totally achieved. This research has focused its implementation on a concrete and novel problem: the migraine prediction. All along the experimentation phase there have been several drawbacks and the implementation has been carried out by iterations. The nature of a real problem, working with real data, poses many difficulties: a lot of people involved, tools and resources, time and money. Due to this, at the time of writing this manuscript, it has not been possible to join all the sub-systems and thus, the implementation of the EDS—and hence the CERPS—shown in this section will draw only partial results.

During all previous sections it has been glimpsed the behavior of the EDS and few things lack to say. In this section it is briefly explained the structure sketched and partially tested. Figure 3.27 represents an example of the behavior of the EDS. On the left side of the figure, along the time axis, there appear events and predictions as outputs of the Sub-systems I, II and III. This example overlays the ideal migraine pain curve as a dotted line in the axes. The EDS generates a decision, an alert/alarm based on all this information.

ANFIS models have been successfully implemented for classification of neurological problems [68]. So, it was decided to implement ANFIS model for a control system of type Takagi-Sugeno. The ANFIS models take the adaptive capacity by propagation backwards from neural networks, and from fuzzy systems they take the representation of probabilistic knowledge of a set of inputs, as well as the smoothed response due to the interpolation of the fuzzy rules. It has been used the Fuzzy Logic Toolbox of the MATLAB software [88].

3.7 Energy aware prediction system

3 Implementation

3.1 Overview

3.2 Sub-system I. Fine-grained modeling

3.3 DEVS-based migraine predictor simulator

3.4 Sub-system II. Prediction support

3.5 Sub-system III. Prediction support

3.6 System implementation: CERPS

3.7 Energy aware prediction system

3.7.1 Energy aware predictive models

3.7.2 Compression techniques for energy saving

3.7.3 Workload balancing in an MCC system

Energy consumption is always a constraint to take into account in IoT scenarios. In Mobile Cloud Computing, sensing nodes might be far away from a stable and continuous power source. In most of the cases the form factor of the battery limits the size of the monitoring device such that high density integration is still a challenge to deal with. The problem addressed in this thesis requires long lasting monitoring periods, ideally 24 hours a day using small wearable devices. Thus, the optimization of the energy consumption becomes a major concern that must be tackled properly.

In this Section there are going to be shown different approaches to optimize the energy consumption in different study cases from lower to higher abstraction layers. (i) In the lower layer (Section 3.7.1) it is going to be optimized the energy consumed by the prediction models and the sensor selection in a monitoring node, (ii) the second layer (Section 3.7.2) applies compression techniques in order to reduce the amount of raw data transmitted from the sensing node, and (iii) the third layer (Section 3.7.3) encompasses all the elements in the system to balance the workload applying a global energy optimization.

3.7.1 Energy aware predictive models

This Section describes the workflow followed in the optimization process depicted in Figure 3.28. After the data acquisition and the offline data processing the predictive modeling block generates an output that represents the prediction of the system. This output depends on past values of the input variables. To illustrate the process it is used the already presented Grammatical Evolutionary algorithms in the prediction modeling block. In the same way, there are going to be used

biometric variables as system inputs to model the output of a migraine event. The energy optimization framework proposed in this section can be applied to other prediction modeling systems.

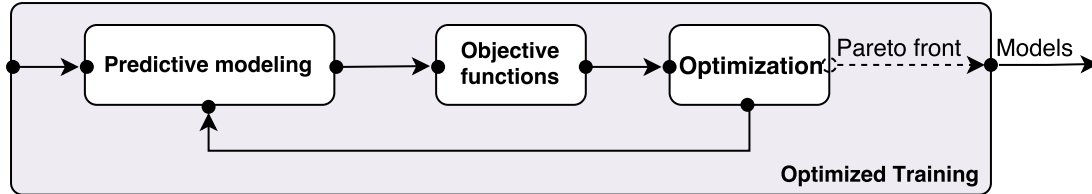


Figure 3.28: Overview of the energy optimization workflow proposed. The **Objective functions** block computes the metrics to be evaluated. The **Optimization** block weights metrics and depicts a Pareto front for every generation of the GE algorithm. Based on NSGA-II, the **Predictive modeling** block creates new models. Eventually, after convergence or the last generation established, the last Pareto front provides the best solutions to be manually selected.

This approach is applied in the implementation scheme depicted in Figure 3.29. This scheme is the one that also simulated using DEVS. The colored shadowed block indicates the element of the architecture that benefits by this energy optimization process. These models implement simple non-linear equations easily programmable in wireless monitoring devices obtained through Grammatical Evolutionary algorithms, and optimized for the architecture of the microprocessor MSP43016F1101.

The GE algorithms create a set of mathematical expressions from which we extract three optimization objectives. These objectives are: (i) the accuracy or *fit* of the predicted values, (ii) the number of clock cycles $\#clk$ that the expression takes to be computed in the embedded microcontroller, and (iii) the energy consumption $\varepsilon_{sensing}$ of the sensors used.

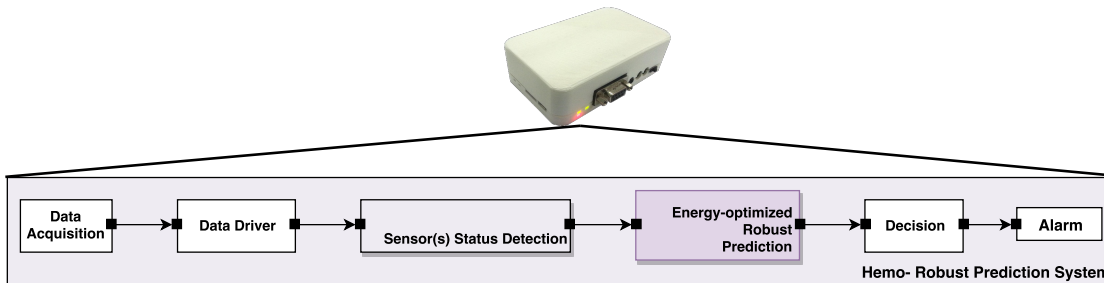


Figure 3.29: Optimal approach of low-power migraine prediction models for an implementation of the system where the CERPS fits in the monitoring device. The colored shadowed block indicates the element of the architecture that benefits by this optimization process.

Table 3.4: Parameters for the GE modeling with energy constraints.

Parameter	Value
Number of generations	500000
Population size	250
Probability of crossover	0.9
Probability of mutation	0.083
Chromosome length	100
Wrapping	No

Some constrains have been set: $0 \leq fit \leq 100$, and only solutions that use data provided by any of the sensors are considered. Those solutions with negative fit, infinite $\#clk$ or infinite $\varepsilon_{sensing}$ are removed. The multi-objective problem can be formulated as follows:

$$\min(-fit, \#clk, \varepsilon_{sensing}) \quad (3.8)$$

Energy optimization in the loop

The goal of this section is the energy optimization methodology and not the models themselves, and it has been introduced the energy optimization loop in the already developed GE predictive model (Section 3.2.2.1) where minor changes have been introduced.

The optimization is based on the Non-dominated Sorting Genetic Algorithm II (NSGA-II), which draws a bi-dimensional Pareto front.

To train the models, the parameters in Table 3.4 are used. The BNF remains unchanged with respect to the one in Figure A.4.

The objective functions

In the following lines there are defined the objective functions in Eq. 3.8, and a detailed explanation of them is further given.

The fit: measures the likeness of the predicted value by the system \hat{y} —the migraine pain level in our case—and the known real output y —in our case, the symptomatic curve reported by the patient. The fit definition is the same already used. To maximize this value, the optimization process tries to minimize its opposite value.

The number of clock cycles: the main goal of this section is the reduction of the energy consumption of an ambulatory monitoring device. As aforementioned, one of the ways to achieve it is through the reduction of $\varepsilon_{\mu C}$, the energy consumption of the microcontroller that eventually will compute the migraine prediction in real time. Given the impossibility of measuring the real consumption of the monitoring device to introduce this value into the optimization loop, several assumptions were taken. The authors did not find neither a simulator or instruction level energy consumption table for low performance microcontrollers.

$$\#clk_S = \sum_{i=1}^{\forall f \in FinS} \#clk_{f_i} \quad (3.9)$$

$$\#clk_{f_i} = \frac{\sum_{b=1}^B \#clk_b}{B} \quad (3.10)$$

The main assumption was that the number of clock cycles that the microcontroller takes in the execution of the prediction is proportional to its energy consumption, $\varepsilon_{\mu C} \propto \#clk_S$. The predictive modeling box is, in most of the cases, data dependent. In our case using GE, this dependence appears in the output range of the mathematical expressions generated. Data from sensors might be bounded, but not the result when they are introduced in a mathematical function. Thus, trying to make a profile via a static code analysis is not possible. For the same reason, any strategy of branch prediction could not be carried out.

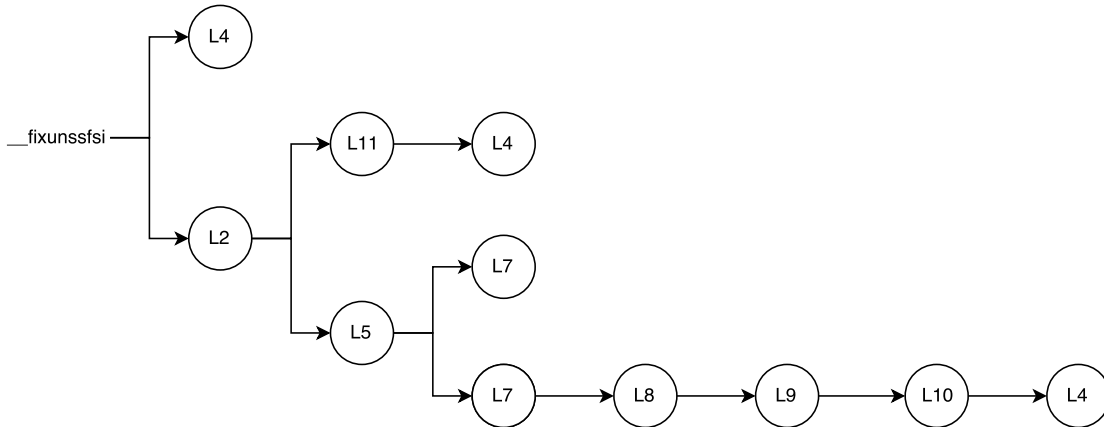


Figure 3.30: Unwrapped branches possibilities for the `__fixunssfsi` function of the `C` library `glibc`. Each node represents a label in the disassembled code.

It is also assumed that the total number of clock cycles for a GE solution S is computed as the sum shown in Eq. 3.9. Where the number of clock cycles for a given function $f_i \in F$ is the average value of the clock cycles of all branches B in

Table 3.5: Average #clk of all branches of `__fixunssfsi`.

# Path	Label	Total #clk
1	-L2-L11-L4	114
2	-L2-L5-L4	157
3	-L2-L7-L7-L8-L9-L10-L4	273
Average path		182

the compiled code for a function (Eq. 3.10), as shown in the example of Figure 3.30 and the corresponding values in Table 3.5. It is known that there exists a data dependence but it has been assumed that errors in function calls are negligible, thus the paths that branch directly to terminal labels (as L4 in the example) has been obviated.

The set of functions F is defined by the BNF. The number of cycles for the execution of a code depends strongly on the microcontroller architecture: hardware multipliers, float point unit, vector unit, *etc.*. This study is based on the well-known microcontroller MSP43016F1101, a low-performance 16-bit microcontroller suitable for monitoring devices. Its maximum frequency is 8 MHz, and it has 10 kB of RAM and 48 kB of flash memory. This is the one used by the Shimmer 2r platform of Figure 3.4a.

The assembler code for each instruction in F is obtained using the `msp430-gcc` cross-compiler. All clock cycles and branches are calculated manually, according to the technical specifications of the microcontroller. The number of clock cycles for unresolved calls of internal functions of the tool-chain are calculated as well. In those cases where calls to kernel functions were not able to be resolved, the mathematical functions were approximated by 10-degree Taylor series—only *sine* and *cosine* functions were approximated. The standard `math` library is used for the others.

The sensing energy consumption: $\varepsilon_{sensing}$ draws the amount of energy consumption due to the use of every sensor. As the sampling frequency and hardware of each sensor is different, it is preferred to use the minimum number of the least consuming sensors that allow the maximum fit.

The experimental set-up

The optimization is based on the architecture of the microcontroller available in the open source commercial Shimmer device that integrates the microprocessor MSP43016F1101. With this choice, and using real data, it is possible to implement

the obtained models in the device to compare their energy performance against the baseline scenario: monitoring devices without predictive intelligence that sense using all sensors and transmit the data wirelessly.

The sensors used are the ones described in Section 3.5: the NTC thermistor SA2F-TH-44031-40 from Omega to measure skin temperature, two differential electrodes to measure the EDA, a single-lead local differential bipolar sensor (three leads) to extract the HR from ECG, and the SpO2 using the 8000R SpO2 sensor and the OEM-III board.

For the optimization process, the number of clock cycles for each function, and the energy consumption of each sensor are needed. According to the technical reports of the microcontroller and the criteria aforementioned, the clock cycles for the base functions of the microcontroller are: 184 for **addition**, 177 for **subtraction**, 153 for **HW multiplication**, 37 for **comparison**, and 405 for **division**. For each mathematical function individually compiled from the F set, the average number of clock cycles of all possible paths are: 4443 for e , 6416 for \cos , 6612 for \sin , 1079 for $\sqrt{}$, 12344 for pow , 4890 for \log and, \max and \min values are proportional to the number of **comparisons**. Clock cycles for the computation of the HR are introduced too. To this end, it has been implemented by Boichat *et al.* in [19]; it takes 4672432 cycles (almost 0.5 seconds in a microcontroller running at 8 MHz).

For the third objective of the optimization problem it has been used a HAMEG HM8012 digital multimeter to measure the consumption of the sensors. TEMP and EDA are analog sensors, sampled every 5 seconds, that consume $0.32mJ$ ($-4.9dBm$) each in the Shimmer mote. The ECG is gathered at 250 Hz leading to an energy consumption of $396mJ$ ($26dBm$). The consumption of the OEM-III and the SpO2 sensor is far from these values, and it is $3665mJ$ ($35.6dBm$). The digital module OEM-III samples the SpO2 at 75 Hz, and sends through the microcontroller 3 kbit of data per second. The presence or absence of this sensor will have a high impact in the optimization process of the GE expressions. The HR computation leads to $34mJ$ ($15.3dBm$) extra, approximately.

It is used the HERO Java library using BNFs for 10 and 20 minutes prediction horizon. The experiments last several days executed in a heterogeneous distributed cluster of PCs with Intel Core i7-4710HQ and AMD A8-6600K CPU, with 16 GB and 8 GB of memory and, Ubuntu and CentOS respectively. In Chapter 4 it will be shown the three-dimensional Pareto front and the decision making task to select the best models according to the three optimization variables.

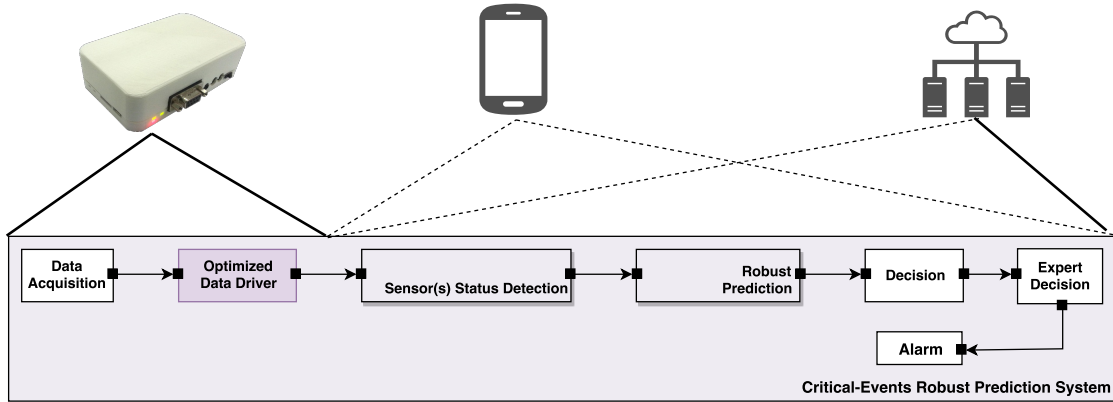


Figure 3.31: Optimized data transmission from the monitoring node to the coordinator in an implementation of the system where the CERPS is distributed among the different elements of the MCC architecture. The monitoring node acquires, compresses and transmits the data, whereas the data processing and prediction takes place either in the smartphone or in a Cloud Data Center.

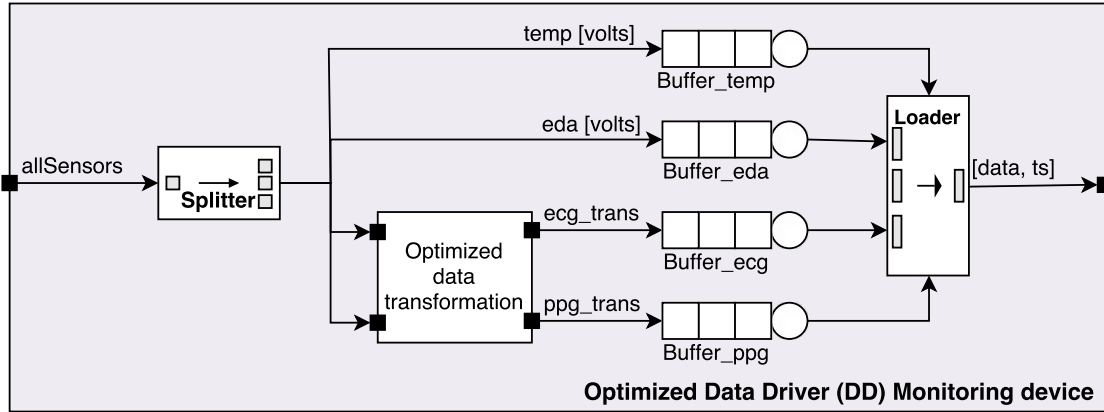
3.7.2 Compression techniques for energy saving in wireless transmissions

Figure 3.29 in the previous Section showed an optimization scheme for an isolated CERPS fitted in a wireless monitoring node. In this Section the MCC scheme changes and looks like the one presented in Figure 3.31. Solid lines behind the monitoring node indicate that data from sensors are collected and transmitted wirelessly from the monitoring node to a coordinator (smartphone or similar). As explained in Chapter 2, dotted lines indicate that data are processed out of the node either in the coordinator or in the Data Center. This scenario makes interesting if it is necessary to store the raw data in a data base for further data processing or visualization.

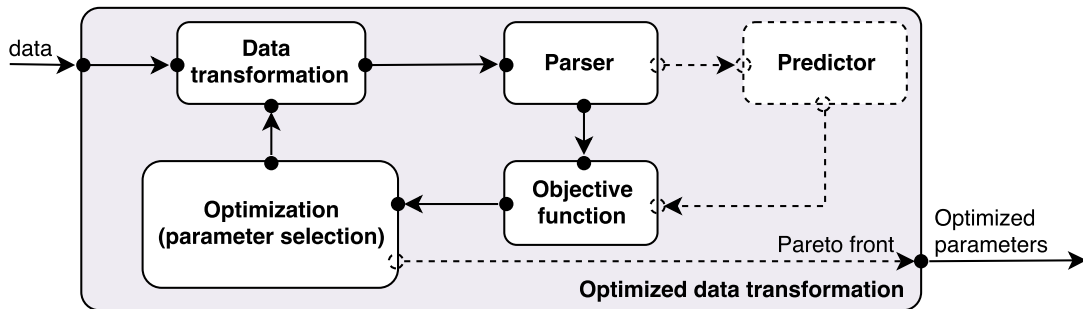
Data transmission is the most power hungry task in a monitoring node despite of the technological advances in the wireless transmission field (like Bluetooth Low Energy, BLE). In this Section there are going to be shown different digital processing techniques which have been used for lossy-compression to reduce the energy consumption in transmission from the monitoring node [113]: (i) the Digital Wavelet Transform (DWT), (ii) the Discrete Fourier Transform (DFT), and (iii) the Compressed Sensing (CS).

In our case, the compression it is only applied over ECG and PPG as shown in the *Optimized Data Driver* in Figure 3.32a. PPG is gathered from the OEM-III module, SpO₂ can be obtained from it. ECG and PPG are the two variables with a higher sampling rate (250 and 75 Hz respectively). The optimization methodology is shown in Figure 3.32b. Data are transformed using the selected parameters,

then, in the `Parser` block data are recovered (through inverse transformation) and the fit is computed in the `Objective function` block.



(a) Optimized Data Driver. Only few data of the transformed domain of ECG and PPG are sent (compressed data). Data recovery and parsing take place in the coordinator.



(b) Optimization of the parameters of the transformation functions. Parameters vary from one transformation function to another. The objective function is the fit. In the basic workflow the optimization minimizes the error over the data—parsed (HR) or raw data (PPG). In a more ambitious case parameters can be optimized to minimize the impact in the migraine prediction (dotted `Predictor` block).

Figure 3.32: Energy saving through data compression. (a) Optimized Data Driver; (b) Proposed optimization workflow. The optimized parameters are used in `Optimized data transformation` block in (a).

The problem to solve is a mono-objective optimization with the *fit* as optimization objective. The problem is bounded and it has been solved by an exhaustive search though an optimization loop. For each one the transformation functions:

- DWT: using a bi-orthogonal family base, the optimization minimize the number of approximation coefficients (A) in Eq. 3.11. In this case the most energetic A coefficients are sent to the coordinator, and the detail

coefficients (D) are removed. Thus, this minimizes the number of A coefficients, k , and maximizes the fit.

$$\min(k, -fit)$$

$$x[n] = \underbrace{\frac{1}{\sqrt{N}} \sum_k W_\varphi[j_0, k] \varphi_{j_0, k}[n]}_{\text{Approximation coefficients (A)}} + \underbrace{\frac{1}{\sqrt{N}} \sum_{j=j_0}^{\infty} \sum_k W_\psi[j, k] \psi_{j, k}[m]}_{\text{Detail coefficients (D)}} \quad (3.11)$$

- DFT: the node computes the FFT^N and sends the most relevant frequency components to the coordinator. The optimization problem filters the data with an ideal rectangular bandpass filter of bandwidth $BW = k_h - k_l$ [rad] to minimize the number of frequency components to be sent, while it maximizes the fit.

$$\min(k_h - k_l, -fit \mid k_h, k_l = 0, 1, \dots, N - 1)$$

It has been chosen an FFT with order $N = 2^{\log_2(\dim(x[n]))}$ to implement a Butterfly diagram programming too be faster and energy efficient.

- CS: in this case, the compressed data are transmitted. The optimization problem maximizes the compression ratio and the fit.

$$\max(cr, fit)$$

For CS, it is worth to do extra on-node data processing because the energy consumption due to data transmission drops drastically. In this case, on the contrary to what was done in Section 3.7.1, the energy consumption due to data processing has not been taken into account in the optimization loop. In this case, there have been coded optimized implementations of the transformation functions—Forward biorthogonal 9/7 discrete wavelet transform, CDF 9/7 for DWT, and an 8-point FFT which is basically a Decimation in Frequency of the FFT. The energy consumption has been measured over a commercial device using an ATmega328 microcontroller using the current monitor INA219. The optimization and energy consumption results are shown in Chapter 4.

3.7.3 Workload balancing in a Mobile Cloud Computing System

This section describes a high level abstraction implementation to apply global energy optimization. This encompasses all the elements in the system to balance the workload among the elements of the prediction system. Figure 3.33 shows that tasks can be performed almost everywhere. In this figure, the bidirectional shadowed arrows indicate that the energy optimization covers the whole system.

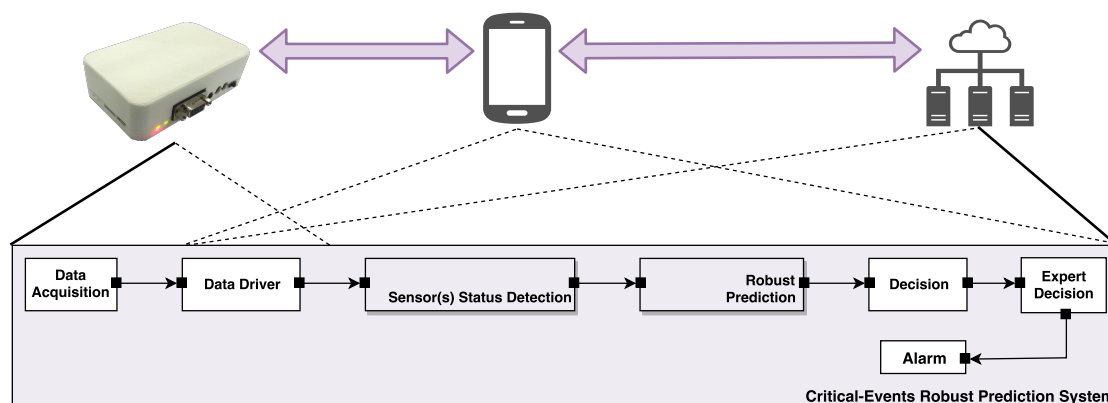


Figure 3.33: High level implementation of the CERPS to perform workload balancing optimization to achieve energy savings in the prediction system. The bidirectional shadowed arrows represent how the intelligent workload balance—aware of the network status and the energy consumption of each element—chooses what must be done, where and when for a energetic and rewarding global optimization.

Savings are not achieved for a particular processing block or element in the network, but it is achieved globally by an intelligent workload balance that chooses what must be done, where and when, aware of the network status and the energy consumption of each task.

This implementation varies the original implementation where the CERPS fits to the monitoring node. This experiment have been extrapolated to a potential real IoT scenario with thousands of people—target population—being monitored. The three main research goals are: (i) the energy efficiency for on-body channel transmissions in WBSNs, (ii) the predictive modeling of migraines in a real scenario in the context of a clinical study, and (iii) the optimization of energy consumption by means of workload balancing in Data Centers in an *eHealth* scenario.

Figure 3.34 depicts the scheme of the system to be implemented in this thesis. In the Figure, each monitored patient wears two monitoring devices and a coordinator node (a smartphone) that sends the biomedical data from the monitorization nodes to a data center. Data from patients of each country are sent to the Data Center in that country. As the Figure represents, Data Centers are understood as a federation of Data Centers distributed in Europe. Shadowed regions in Figure 3.34 show the countries included in our economic study.

The proposed scenario allows to scale our results and policies for ambulatory prediction of migraines under energy constraints. In this implementation each subject is monitored using two Shimmer 2r sensing nodes wirelessly connected to a coordinator. A coordinator is a computation system with a higher performance than the monitoring nodes (usually a smartphone). The sensing nodes monitor the biomedical variables. The coordinator may perform the predictions or it can

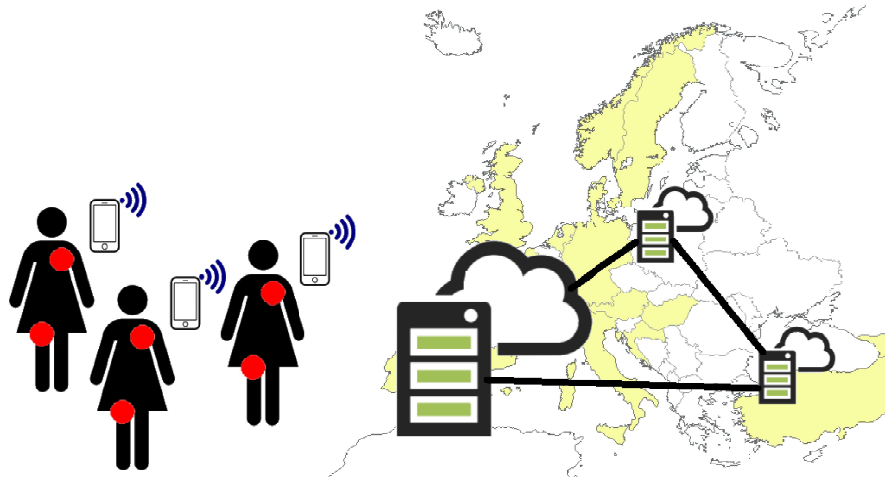


Figure 3.34: Scheme representing the structure of the system. Coordinator nodes (smartphones) communicate data from monitored migraine patients to a Data Center that belongs to a federation of Data Centers distributed in Europe. Shadowed regions are the countries included in the economic study.

send the data to the Cloud and the predictions could be computed there.

The expenses of electricity bills of such deployment are high, as the Data Center is an infrastructure with a high energy consumption and many smaller power consumers (nodes) are also deployed. This *eHealth* scenario must be economically rewarding to have a smooth establishment in the market. For that purpose, the policies for the energy efficiency are focused at two different levels of the *eHealth* scenario:

- In the monitoring nodes: the battery consumption of the monitoring devices must be managed in order to increase their lifetime and autonomy. This problem has been tackled by optimizing the energy consumption of the monitoring devices: (i) in the radio communication and, (ii) in the computation of the gathered data. The energy consumption required for sampling signals has not been considered in the optimization, but it is detailed in further below.
- In the high performance computing systems: this 24-hour monitoring study generates vast amounts of data that must be managed in order to create prediction models for the patients. This process requires computing capacities only available at state of the art Data Centers. Once prediction models have been created, the prediction of an incoming event can be computed at two points of this IoT scenario: (i) a coordinator node (assuming an intelligent monitoring device) or, (ii) a Data Center. The energy consumption of these facilities leads to unsustainable costs. In this

implementation it has been also optimized the energy consumption by balancing the workload between the two points of the monitoring and prediction framework (the coordinator node and Data Center), in order to save money and achieve energy efficiency.

To fulfill the goal of this framework it has been considered that each part of the *eHealth* scenario can work in different modes. There are described three different modes corresponding to the amount of energy used for gathering and preprocessing data, and making the predictions. The first mode represents a baseline where data is gathered and transmitted in streaming without saving energy in the monitoring nodes. The second mode applies energy reduction techniques in the monitoring devices and energy aware off-loading techniques between the coordinator and the Data Center. The third one applies Data Center energy minimization policies on top of the previous off-loading techniques.

In this experiment it has been applied an ECG signal processing algorithm to assess the impact on the global energy reduction. This is based on the implementation of Rincon *et al.* [142] for the ECG delineation using their single-lead optimization for the Shimmer node. For further implementation details, we encourage the reader to refer to the original work in [142]. The surface skin temperature and the electrodermal activity values do not need to be processed, as well as the SpO2 data that are gathered directly from the OEM-III module.

Data obtained using the WBSN is communicated to a smartphone, that forwards the data to the Cloud. In order to predict migraine, huge data sets must be analyzed. To deal efficiently with such computationally intensive tasks, part of the processing and storage will be local to the coordinator, while another part will be communicated and processed in the Cloud, *i.e.* in Data Centers—the extreme end of the network. As the target population is so large, the key challenge in MCC scenario is the definition of strategies to off-load and distribute efficiently the workload between the different elements of the system. This implementation devises new strategies to combine these two aspects: (i) workload off-loading and (ii) Data Center energy efficiency techniques, to further reduce the energy consumption of *eHealth* applications. In this section it is shown a realistic case study for migraine prediction and shows the overall impact of this strategies over the system.

The experimental set-up

In the following lines it is going to be explained the most important characteristics of the models for the on-body channel transmission and workload

balancing. But it is needed to explain firstly how the experiments have been carried out. Most of the experiments carried out in this thesis that use only one monitoring node (Plux, or our own device); however there have been realized other experiments using two nodes—those using the Shimmer node. To explain this experiment of workload balancing computing, there were used two Shimmer nodes placed in the body. These two nodes (S1 and S2) communicate with a coordinator, and in these experiments, there are considered the effect of on-body transmission—performing two communication links (L1 and L2). For the sake of simplicity, the study carried out in the clinic for the on-body channel transmission modeling is explained in Appendix B

The WBSN and the ambulatory monitorization

The ambulatory monitorization starts when the patient leaves the hospital. During a period that comprises from two weeks to one month approximately (day D in Figure 3.6), data are transmitted by the coordinator to the Data Center via a 3G link. The training phase of the predictive modeling through hemodynamic variables remains as explained in Section 3.2.2. Due to the high computational burden of these calculations when done for a large population, these models need to be generated in a Data Center. All models used in this experiment are state-space models created for a prediction horizon of 30 minutes. When the number of migraines is large enough, the ambulatory monitorization finishes. During day D-1 (Figure 3.6) models are validated and tested, and the more accurate ones are selected. The final prediction is the average of the prediction of several models in order to avoid overfitting or loss of accuracy.

The state-space matrices calculated in the training phase have been implemented in C-code to run them in the servers of the Data Center and in the coordinator node. In parallel to the prediction calculation, the workload balancing policies are carried out, as can be seen in the right side of Figure 3.6. Depending on the number of patients being monitored, the computational burden of the prediction, the Data Center utilization, and the battery status of the coordinator nodes, the off-loading policy decides during runtime where the prediction takes place: in the Data Center or in the coordinator node.

The sensing nodes. As aforementioned, in this experiment the WBSN is composed by two Shimmer nodes and the coordinator (a smartphone) wirelessly connected in a star topology. Two different working modes for the sensor-coordinator communication are analyzed: (i) streaming mode—where data are gathered and transmitted without preprocessing, and (ii) preprocessing—where data is preprocessed in the nodes before being

transmitted.

The main characteristics of the Shimmer devices were detailed in the implementation done in Section 3.7.1. These nodes incorporate the CC2420 chip radio, which performs the radio interface, and implements the IEEE 802.15.4 radio standard. To simulate the radio interface it has been used FreeRTOS. FreeRTOS is portable, open source, and it has a hard real-time mini kernel that includes support for the microcontroller and the IEEE 802.15.4-compliant radio chip used by Shimmer.

The data acquisition parameters remain the same as showed in Table 3.1. However, now that raw data are sent and it might be useful for data processing and visualization in the Data Center so, the PPG is also considered. Thus, Table 3.6 summarizes the amount of gathered data per day. In case of preprocessing in node S1, the HR is calculated every 1.25 seconds using 20 seconds of ECG data. This rate ensures the detection of peaks at 48 bpm for normal resting rates described in [114]. In case of using node S2 for data processing, SpO2 is only computed once a second, according to the OEM-III module bitrate.

In this work we apply an ECG signal processing algorithm to assess the impact on the global energy reduction. This algorithm bases on the implementation of Rincon *et al.* [142] for the ECG delineation using their single-lead optimization for the Shimmer node. For further implementation details, we encourage the reader to refer to the original work in [142].

The power consumption required for the sampling of the signal has not been considered in the optimization as it cannot be modified in some of the sensors, or it would require the application of compressed sensing techniques. As shown in Table 3.1, the sampling rate of the TEMP and EDA sensors is negligible. The sampling of the ECG sensor is 250 Hz—the standard quality level in most of the commercial ECG monitoring devices—so, it has not been contemplated a reduction of the ECG sampling frequency. It could be possible if we apply compressed sensing techniques, but that is out of the scope of this experiment.

Battery discharge or sensor disconnections occur frequently. Table 3.7 shows empirical results of the probabilities of disconnection or unavailability for the sensors of hemodynamic variables. Probabilities have been computed from 8 migraines lasting 7.7 hours on average. TEMP, EDA and HR sensors have the same probability of disconnection, 6.2%. This is due to the battery discharge rate of node S1. However, the SpO2 sensor has a higher probability of disconnection, 29.7%, mainly due to the misplacement of the sensor in the groin. On average, almost 1 of 3 packets of SpO2 are lost. Note that whenever a packet is lost, it has been run the signal repair module as part of the data preprocessing to recover the signal and be able to obtain a prediction. Thus, data loss increases

Table 3.6: The acquisition parameters in the Shimmer nodes. Processed data information shown in parenthesis.

	Sampling rate (Hz)	Data-24h (kB)
TEMP*	1/60	2.1
EDA*	1/60	2.1
ECG (HR)	250 (0.8)	31640.6 (67.5)
PPG (SpO2)	75** (1***)	31640.6 (84.4)
Total (MB)		61.8 (0.15)

* EDA and TEMP are considered negligible for energy consumption.

** Only PPG is needed but OEM III module captures and sends 375 Bytes.

*** SpO2 is internally calculated by the OEM-III.

the computational burden, and needs to be considered during runtime prediction.

Table 3.7: Probability of sensor data loss.

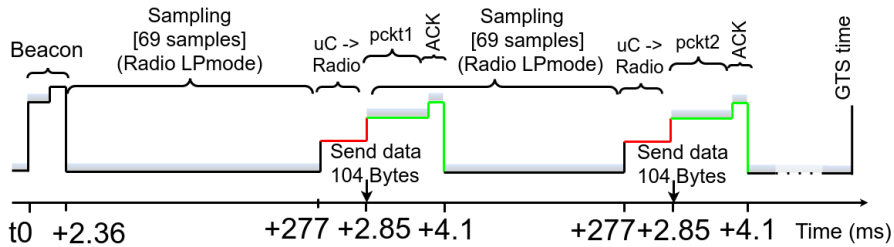
	TEMP	EDA	HR	SpO2
# breaks/hour	0.24	0.24	0.24	0.57
Average duration (min/break)	15.5	15.5	15.5	31.3
Probability (%)	6.2	6.2	6.2	29.7

Regarding to the radio, for the 802.15.4 standard, a 104 Bytes data payload has been used, with 250 kbps transmission rate. One transmission packet (including headers) is 128 Bytes long. Table 3.8 shows the transmission rate TX_r , the amount of data sent D (including headers), the transmission time cost TX_t , and the duty cycle (the ratio of transmission time to transmission rate) for nodes S1 and S2 in each working mode.

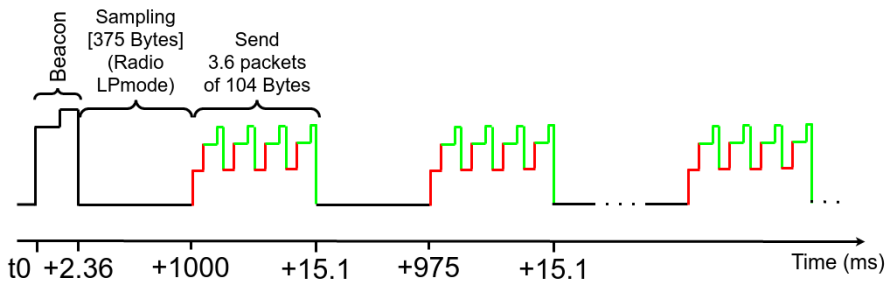
Table 3.8: Data transmission properties for Shimmer nodes.

	Streaming mode		Processing mode	
	Node S1	Node S2	Node S1	Node S2
TX_r (ms)	277	1000	60000	60000
D (Bytes)	128	471	72	84
TX_t (ms)	4.1	15.1	2.3	2.7
Duty cycle (%)	1.48	1.51	0.004	0.005

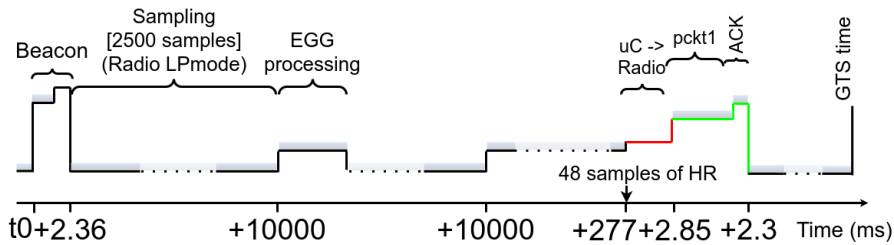
Figures 3.35a through 3.35d complement values in Table 3.8. These figures are an interpretation of the energy levels of the tasks and their timestamps. In the streaming working mode for node S1 (in Figure 3.35a), when 104 Bytes are recorded (69 samples of ECG), the acquisition stops and data flow from the buffer



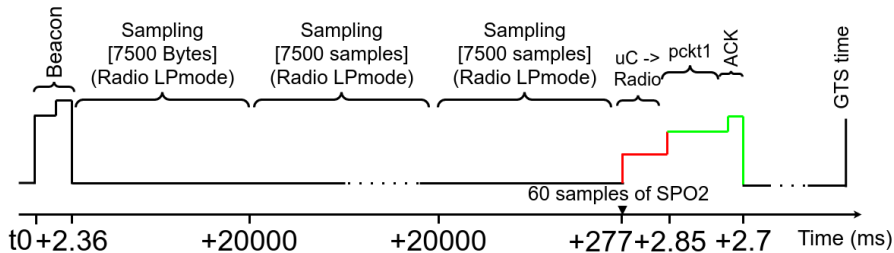
(a) ECG streaming.



(b) All NONIN data packet streaming.



(c) HR processed.



(d) SpO2 signal extracted.

Figure 3.35: Diagram of the streaming and processing working modes in the two sensing nodes. The y-axis (not in scale) of these figures represent the energy level of each task. The x-axis (not in scale) marks the most important events. Shaded areas represent the data acquisition.

in the microcontroller to the buffer in the CC2420 chip to be transmitted. Thus, every 277 ms the radio wakes up and transmits. During the acquisition time, the radio is in low power mode. The transmission takes 4.1 ms, as Table 3.8 shows. Every 60 seconds, when ECG is processed to reduce the amount of transmitted data, 48 samples (1 Byte each) of HR are sent.

Regarding the working modes in node S2 (Figures 3.35b and 3.35d), it can be seen some peculiarities. The data to be sent are always larger than the radio buffer size of 128 Bytes. With this, data have to be truncated and sent in burst mode. 375 Bytes of data have to be sent in the streaming mode; this leads to transmit 3 packets of 104 Bytes of payload and an extra packet of 63 Bytes. Thus, 4 headers of 24 Bytes are sent. In this way, we achieve the 471 Bytes in Table 3.8. The radio wakes up once a second and sends the data in 15.1 ms. In the second working mode, this node extracts only the SpO2 information once a second, and transmits data every minute. The SpO2 data extracted are 60 Bytes (one Byte per second) and 24 Bytes of header are added, leading to the 84 Bytes that appear in Table 3.8.

The beacon time of the communication protocol and the packet transmission time from the microcontroller to the radio chip in Figures 3.35a through 3.35d are based on previous work by Rincon *et al.* [142].

The coordinator node. The coordinator node in the real experiments is a smartphone. To simplify the energy characterization and isolate the contribution of the smartphone's processor from other components, it has been performed the power characterization in a BeagleBone Black platform¹⁶. This platform uses the same processor as the Samsung Galaxy S smartphone, and can be easily instrumented to measure power. The processor is an ARM Cortex-A8 at 1 GHz with 512 MB DDR3 RAM. Data are stored in an external 2 GB SD memory card. In order to isolate the energy measurements, as radio device, the same CC2420 as in the Shimmer devices is supposed to be used in the smartphone, and this interface has been attached to the BeagleBone platform. As it has been envisioned a case study to apply energy-aware off-loading policies that balance computation between the elements in the WBAN and the Data Center, the coordinator node—in addition to transmitting and receiving data—may perform two main actions: (i) data preprocessing (off-loading computation from sensors), and (ii) online prediction (off-loading computation from the Data Center).

Energy costs of the WBSN. The energy consumption of the microcontroller and the external devices has been measured using a high precision digital amperimeter, whereas, the consumption of the radio has been simulated, due to the complexity in measurement of energy consumption in the

¹⁶Beagle: <http://beagleboard.org/BLACK>

real scenario. Simulation has been carried out with the open source simulator Castalia¹⁷, which is designed specifically for WBSN networks and it includes a channel model based on data measured empirically, and a model of radio CC2420, used for our experiments.

In order to guarantee reliability, most approaches transmit at a fixed transmission power that ensures no packet loss. However, these techniques lead to a waste of energy. Thus, transmission reactive policies based on knowledge of body posture have been implemented. These accelerometry-based transmission power control policies allow to regulate the transmission power levels according to changing link conditions, and minimize the power consumption of the radio chip, while maintaining service quality.

As aforementioned, the algorithm requires the characterization of the patient in terms of the RSSI and the Packet Error Rate (PER) metrics with respect to on-body positions. Once the communication link has been characterized properly, the calculation of the transmit power optimal level is done offline using the experimental data, and as result, a LUT with the best power levels for each body position is generated. Controlling the transmission power is done online; the optimum level of power required to ensure the quality of the link is dynamically selected from the values stored in the LUT by using motion detection—based on accelerometry with low complexity and low overhead.

It is assumed for this experiment that the radio switches on for every transmission, going off after finishing. Simulations are executed at three different power transmission levels available in the radio CC2420 chip: $-15dBm$, $-10dBm$ and $0dBm$; then the reactive policies are applied. A re-transmission rate has been simulated as random re-transmissions along the time, simulating the channel effects of medium access, using a 802.15.4 MAC with two transmission attempts. The simulation adds a model with temporal variation to recreate the dynamics of the path-loss fluctuations and without collisions.

The configuration setup of the general parameters in simulation has been as follow:

- The energy consumption model: the power consumption in Castalia has two components (i) radio consumption and (ii) baseline consumption. The default baseline consumption value has been modified, thus, the energy consumption of a mote when the radio is off and the microcontroller is active, will be 0 mW. In this way, the baseline value does not affect the real values of energy consumption measured due to processing of the microcontroller. Finally, only the radio consumption is taken into account.

¹⁷Castalia: <https://castalia.forge.nicta.com.au/index.php/en>

Table 3.9: Parameters in the Castalia radio simulator for the Shimmer nodes and the coordinator.

	Streaming mode			Processing mode		
	Node S1	Node S2	Coordinator	Node S1	Node S2	Coordinator
Pkt _r (pkt/s)	3.610	3.596	1	1/60	1/60	1
Payload (Bytes)	104	104	20	48	60	20
Headers (Bytes)	18	18	18	18	18	18
MAC (Bytes)	6	6	6	6	6	6
Duty cycle (%)	1.56%	1.56%	1.56%	1.56%	1.56%	1.56%

- **Duty Cycle:** differences in energy occur because the radio is active for different periods of time in each node. The parameters, beacon order (BO) and superframe order (SO), define the duty cycle between the active and inactive periods. In Castalia, BO equals six and SO equals four, while a duty cycle of 25% is the default value for MAC protocol 802.15.4.
- **Collision Model:** in Castalia, the radio collision model is configured according to the *InterfModel* parameter, which can take three different levels: Level 0, 1 and 2. In Level 0, the simulator assumes no collision at all. We have used for this study Level 0.

Table 3.9 summarizes the configuration parameters of in Castalia for sensor and coordinator nodes in the two working modes. Notice that the values of the duty cycle in Table 3.8 and in simulation are different. This is due to the minimum duty cycle as parameter for simulation is 1.56%. For the simulation of the nodes working in streaming mode, this will lead to be an overestimation of the consumption.

The Data Center

The Data Center setup comprises two clusters: (i) a High-Performance Computing (HPC) cluster to train and validate the models, as these are CPU and memory intensive tasks, and (ii) pre-process data and, after that, repair the data if needed. The tasks of model training and validation are CPU and memory intensive, and require of a resource of computation during the whole execution of the tasks, that use a large amount of data. Therefore, they require an HPC cluster. On the other hand, the execution of the model for prediction is a task driven by requests, that can be executed on a virtualized machine to release the resources after that. Therefore, it is executed on a Cloud cluster.

The HPC cluster is composed of Quad-core Intel Xeon RX300 servers with

16GB of RAM, and the cluster is managed using the SLURM¹⁸ resource manager. The virtualized Cloud computing cluster consists on Intel SandyBridge S2600GZ servers with six cores and 32 GB of RAM. The SandyBridge servers are virtualized using KVM, and the OpenStack software is deployed to manage the deployment and allocation of Virtual Machines (VMs). Due to the intensiveness of prediction of models, it has not been oversubscribed VMs, *i.e.* it has been deployed only 1 VM per core, and it has been assigned 2GB of RAM per VM. In order to characterize power and performance, servers are monitored via IPMI software and using current clamps, allowing us to obtain power values with an error of ± 10 W.

There have been characterized in terms of power and performance all the offline tasks in the Intel Xeon server: (i) the data preprocessing—Gaussian Process Machine Learning (GPML)—(ii) training, and (iii) validation. There have been also characterized all the online tasks in the SandyBridge server: (i) the data preprocessing (GPML), and (ii) online prediction. There have been run stress and performance tests on the VMs of the virtualized cluster to obtain the maximum number of GPMLs and predictions that can be performed in each VM without degrading the throughput.

The characteristics of the HPC and the virtualized Cloud cluster to compute predictions are next outlined. For a detailed explanation on how these parameters are obtained, the reader is referred to Appendix C:

- The preprocessing, training and validation phase of 1 patient runs for approximately 3.5 hours in a dedicated server, and degrades up to 3.76 hours when the server is fully utilized, with a power consumption that varies from 157 W to 173 W. The HPC cluster consists on 2275 server, providing up to 9100 cores. Preprocessing, training and validation are performed in a simulated HPC SLURM cluster [110]. It has been considered that, in average, models need to be re-trained once a month per patient, to stay within allowable error margins.
- In the online phase predictions are computed every minute. Runtime prediction consists on a data preprocessing stage (*i.e.* data recovery via GPML in case there was a packet data loss), and migraine prediction 30 minutes forward, *i.e.* a migraine is predicted 30 minutes in advance. The virtualized cluster consists on 1638 servers that can run up to 9828 VMs. Up to 230 preprocessing instances and 250 prediction instances can be packed together in the same VM, without degrading performance. When a VM reaches this limit, OpenStack is responsible for automatically scaling

¹⁸<https://slurm.schedmd.com>

up the resources, launching as many VMs as needed to handle the workload.

The previous setup allows the HPC and Cloud clusters to handle in the offline and online phases up to 1,393,649 migraine patients. This represents a 2% of the migraine sufferers in Europe (see Appendix C and Appendix D), which is the target population in our case study.

It needs to be considered that runtime prediction is a light-weight process that can be performed either in the coordinator node or in the Data Center. In this sense, if the coordinators off-load computation from the Data Center, OpenStack will reduce the amount of VMs used. This, together with the usage of server turn-off policies, can drastically reduce the power usage of the Data Center, as shown in Chapter 4.

Regarding cooling, it has been assumed that the whole cluster is cooled via traditional raised-floor air-cooling techniques. As a baseline, it has been considered a room temperature of 22 °C and a PUE of 1.65, which are the most common value for room temperature and the world average PUE value respectively [159]. Cooling optimization policies will be applied to reduce cooling costs while keeping servers under save environmental conditions.

To simplify the deployment, patients have been divided into four large groups, according to their nationalities. Patients are included in the system gradually, as shown in Figure 3.36. The inclusion rates are: 50%, 25%, 15% and 10%, and the total evaluation period is 10 weeks. The gradual inclusion is an exercise of evaluation of the methodology proposed, and it is considered that all patients in a group are included at the same time. First, when the HPC cluster is still empty, 50% of the population is included (patients from Turkey, Germany and United Kingdom). Every two weeks, the remaining groups come into the system. The second group are two countries that represent 25% of European migraineurs (France and Italy). The third group comprises three countries (Spain, Netherlands and Switzerland) with 15% of patients, and finally the remaining 10% patients are distributed in eight countries. For further details, the reader is referred to Appendix D.

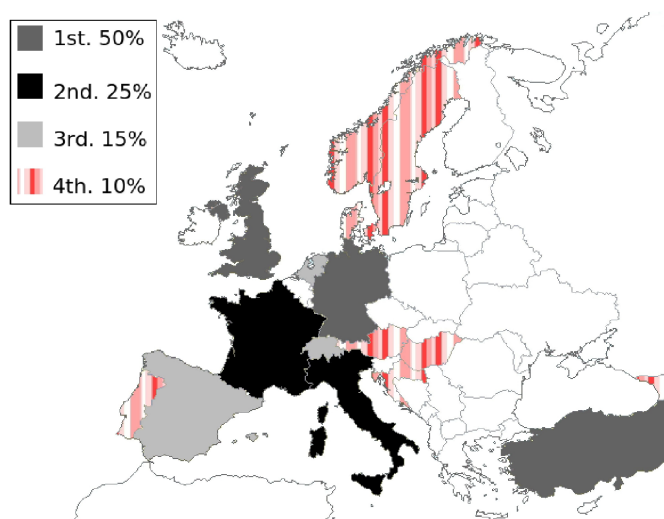


Figure 3.36: Gradual inclusion of migraine sufferers in the system. 50% of this population is firstly included in the study, when the HPC cluster in Data Center is still empty.

Table 3.10: Tasks that each level of the network is able to perform.

Tasks	
Sensor device	Collect and transmit data, process data (limited)
Coordinator	Receive, transmit and process data, perform predictions
Data Center	Process data, perform predictions

MCC *eHealth* optimization scenarios

In this work it is being tackled the energy efficiency challenge from the workload allocation perspective, including the processing and wireless communication level. From the signal processing level there have been performed very light preprocessing techniques in the sensor nodes—due to their limited processing capabilities. The wireless communication level focuses on the on-body channel communication between the monitoring devices and the central node or coordinator.

Each one of the three network elements can operate in various modes. Table 3.10 summarizes the tasks that each level of the network can perform when working cooperatively.

The energy efficiency policies take these possibilities into account in order to minimize the power consumption of the whole system. For the sake of clarity, among all combinations of scenarios, it has been chosen the five considered to be more significant from the energy perspective. These scenarios are shown

Table 3.11: Five scenarios for the workload balancing policies.

	Sensor device	Coordinator	Data Center
SC1	Collect + transmit data	Receive + transmit data	Process data + perform predictions
SC2	Collect + transmit data	Receive + process + transmit data	Perform predictions
SC3	Collect + transmit data	Receive + process data + perform prediction + transmit data	-
SC4	Collect + process + transmit data	Receive + transmit data	Perform predictions
SC5	Collect + process + transmit data	Receive data + perform predictions + transmit data	-

in Table 3.11. In this, it is claimed that using the WBSN and coordinator to off-load computation minimizes overall energy consumption. This technique, together with tailored energy optimization strategies at the Data Center level can drastically reduce the impact of the application deployment.

According to Table 3.11, the sensor nodes only have two different working modes:

- Streaming mode: in scenarios SC1-SC2-SC3. Node S1 collects and transmits data from ECG immediately. Node S2 collects and transmits data from the NONIN devices every second.
- Processing mode: in scenarios SC4-SC5. Node S1, after collecting ECG data, calculates the HR and transmits it every minute. Node S2 extract SpO2 data from the whole frame of 375 Bytes of the OEM-III device and transmit it once a minute as well.

Regardless of the working mode, partial or total data loss is usual in ambulatory studies. As mentioned before, these can occur due to two factors: (i) discharge of the node battery, or (ii) sensor disconnection. When a disruption occurs, a GPML is executed to recover the signals. This process can be executed both in the coordinator as in the Data Center. For the purpose of this experiment, the GPML is executed with the probabilities shown in Table 3.7, which have been experimentally obtained. The coordinator and the Data Center can also process raw signal from sensor nodes (in streaming mode). The online phase can also take place either in the coordinator or in the Cloud Data Center, whereas the offline phase—due to its computational burden—can only be carried out in the HPC Data Center. The results of this abstract implementation of the CERPS are shown in Chapter 4.

Chapter 4

Results

In this Chapter there are shown the results of the implementation of the robust predictive methodology applied to the migraine prediction problem. Following lines summarize the experimental design and data collection. Further below, this Chapter follows up the structure depicted in the preceding Chapter. (i) First, the results of the implementation of the Sub-system I are shown, and then (ii) the results of its DEVS-based simulation. Later, (iii) there are shown some preliminary results of the implementation of Sub-systems II and III and. Eventually, (iv) the results of the energy aware prediction system are exposed.

Data collection and metrics for the experiments

Patients are ideally monitored 24 hours a day. To create personalized prediction models, it is considered that it is necessary a moderate number of crises—8 for example. However, due to the difficulty of the ambulatory experiment it not always possible. The patient is under study for approximately 4 to 6 weeks (this depends on the number of migraine attacks registered). The developed models will correlate changes in the input variables (the hemodynamic variables) to changes in an output variable, in this case, the symptomatic curve.

Once the patient signs the corresponding informed consent (the study protocol was approved by the Local Ethics Committee of the Hospital Universitario de la Princesa of Madrid, see Appendix E), engineers and doctors of the group teach the patient how and where to place the sensors; they will not require a healthcare provider. The medical staff of the hospital selects the body locations where the sensors are placed. The position of the ECG sensors must be on the breast, near the heart. The SpO₂ sensor is located on a finger. EDA can be measured in several locations; this sensor is located on the arm near the temperature sensor (near the armpit), to join the sensor's wires.

The patient wears the WBSN continuously during the day. If the patient has awakeness migraines or migraines when he sleeps, the monitoring process also takes place during the night. During the monitoring process, patients do not change their daily life. If it is required for some activity, such as certain sports, the monitoring process is stopped. The patients replace the sensors when the monitoring restarts. The patients also try to avoid any drug ingestion that can alter the monitored signals during the monitoring process. This is crucial in order to register the real body response before and during a migraine crisis. The patient will be allowed to take some medication if a strong migraine occurs. In these cases, the hemodynamic variables and the pain evolution are also collected; nevertheless, how the real pain would end without the effect of medication will be unknown, due to the effect that medications will accelerate the end of pain.

To evaluate the results the statistical $F - value$ is used. The $F - value$ is the harmonic mean of precision (or positive predictive value, PPV) and recall (or true positive rate, TPR), all of them defined as follows:

$$TPR = \frac{TP}{TP + FN} \quad (4.1)$$

$$PPV = \frac{TP}{TP + FP} \quad (4.2)$$

$$F - value = 2 \frac{TPR \times PPV}{TPR + PPV} \quad (4.3)$$

The TPR shows how many positive detections, TP, are found in the prediction against the false negatives, FN (those events not detected). The PPV confirms how many of those detections are true, taking into account the spurious detections not removed, which are called false positives (FP).

4.1 Sub-system I. Fine-grained migraine predictive modeling

4 Results

4.1 Fine-grained migraine predictive modeling

4.1.1 Preprocessing block

4.1.2 Training block

4.1.3 Validation block

4.1.3.1 Model Repair: improving predictions

4.1.4 Model Selection block

4.1.4.1 Sensor-Dependent Model Selection System (*SDMS*²)

4.1.4.2 Advanced Sensor-Dependent Model Selection System

4.1.5 Test of the Sub-system

4.1.6 Fine-grained modeling using GE algorithms

4.1.6.1 Training of models

4.1.6.2 Validation of models

4.1.6.3 Test results

4.2 Advanced migraine prediction simulation system

4.3 Sub-systems II and III. Prediction support

4.4 Energy aware prediction system

This section presents the results obtained after the implementation of Sub-system I. Results are shown, as in Chapter 3, following the data flow depicted in Figure 3.12. We first start from data preprocessing. Then, the results of the training of the predictive modeling algorithms are shown (N4SID and GE). Afterwards, the results of the models validation are analyzed to show the prediction capabilities of the batch of models obtained during the training stage. The selection of the models is performed based on the validation results. After the validation, a test phase is run to analyze the capability of the generalization of the models. During the test phase, the same method followed in the `Model repair` block is applied to detect false positives using the `Linear decider`. Robustness against sensor failures is also checked during this stage. Some other tests are also checked to analyze the detection of false positives.

Along this section it has been stated a criterion by which a 70% for the fit value has been considered a minimum level quality. This is a conservative setup that will improve the confidence in the models. Thus, a true positive (TP) is considered when a detection is achieved and the fit in the migraine period is higher than or equal to 70%. This avoids spurious detections without a reliable fit. As it was described for Algorithm 1, values above 50% of the probability of the pain curve are marked as positives. Spurious detections not removed by the `Model repair` submodule are called false positives (FP). The alternative approach—to set the prediction horizon by loosing accuracy—is also tested in strategy. For the sake of simplicity most of the implementation results in this section have been evaluated for the state-space models; however, the modeling results for GE models are shown too.

In the test stage, a Sensor-Dependent Model Selection System (*SDMS*²) criteria is proposed, giving a way to deal with sensor failures or saturations. Finally, the results for a real-time scheme—a pool of models that given the set of available sensors, returns the best models according to a hierarchy based on fit or

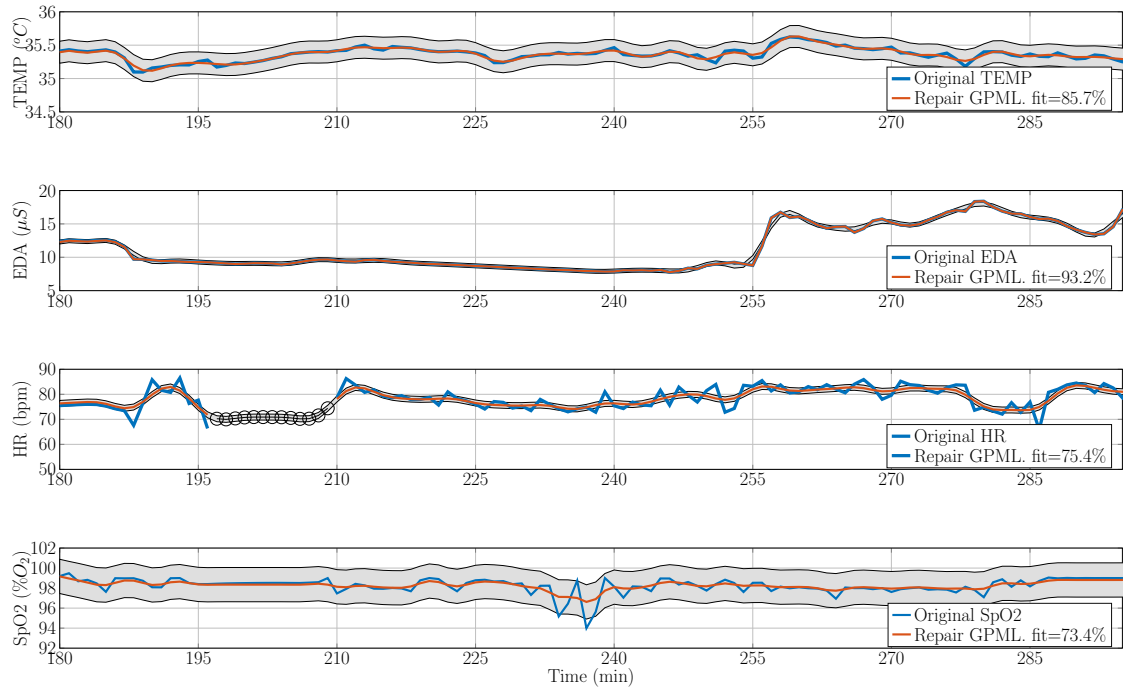
predictive horizon—are shown.

4.1.1 Preprocessing block

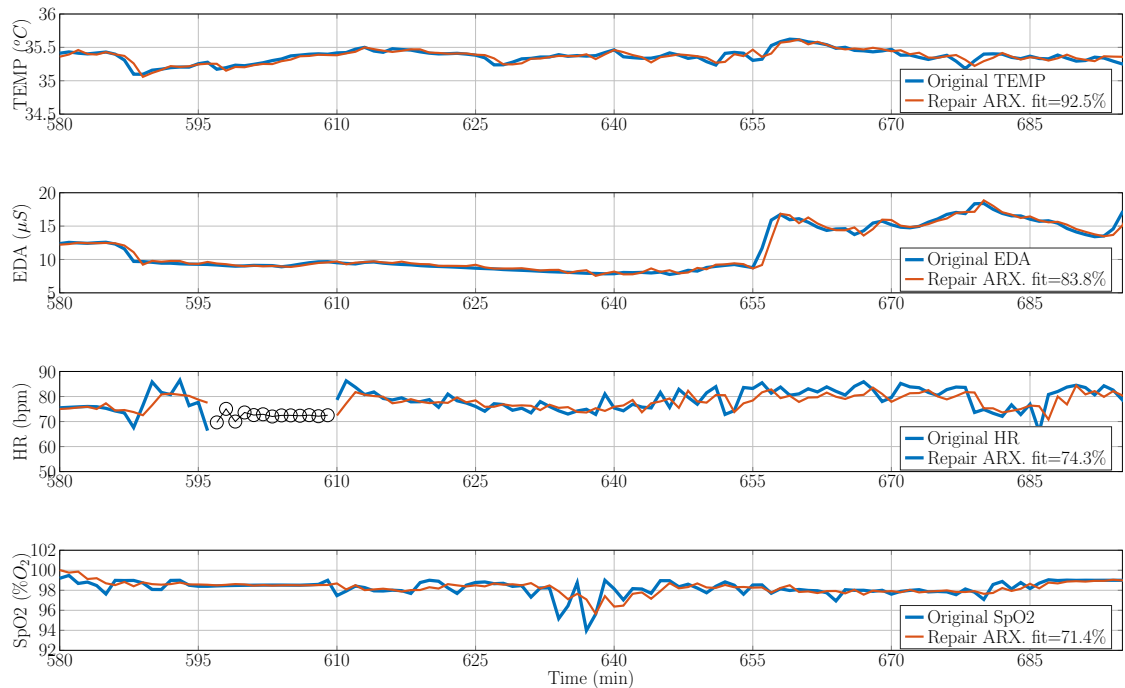
Prior to the synchronization and signal repair, features must be computed from the gathered data. Remember the reader that, for the case study that concerns us, only Heart Rate (HR) is extracted from the ECG, and the raw information of the remaining sensors is used after data parsing. Each migraine episode has a different duration. For the modeling process, the data acquired during some hours before the patient marks the start of pain are also analyzed. This is necessary to identify changes occurring in the monitored variables before the attacks start. After preprocessing all of the gathered data, a set of data is available for model training. This dataset is randomly divided into two groups: around 75% of the monitored migraine episodes (T) are chosen for training and validation of the models, and the remaining 25% are used to test the models.

As previously mentioned, this study pursues the search of personalized predictive models of the migraine attacks per every single patient. Therefore, from this point, the data and results presented belong to a single patient (Patient A). Later, the whole study is repeated for other patient (Patient B) in order to compare the results and draw conclusions. Data from Patient A correspond to a young female patient that suffers migraines with aura and is not under medical treatment. Twenty migraines have been acquired in two different experimental periods (almost a month each). Data from Patient B correspond to a middle aged female patient that suffers migraines without aura and without preventive medical treatment. Eleven migraines have been acquired in one experimental period (almost a month). The variables acquired in real time by our experimental setup comprise the four hemodynamic variables previously mentioned, and the subjective pain was collected from the patient by an electronic form.

Data from the four hemodynamic variables to work with are synchronized (TEMP, EDA, HR and SpO₂) and then repaired in the **Preprocessing module** (Figure 3.12 in page 58). The results are shown in Figures 4.1a and 4.1b for the two different core algorithms have been implemented in the **Signal Repair module**: GPML and ARX. Some lost data have been recovered by this sub-processes. These data are represented with black circles in the graphs. The gray area in Figure 4.1a represents the intervals of confidence of the GPML. The signals have been recovered with high accuracy, achieving similar result for both processes. The fit shown in the legends represent the accuracy of a longer signal but just an interval of almost two hours is drawn. These fits are good enough; nevertheless, a finer tuning of the modeling parameters in the GPML could improve the results. This work will be



(a) Signal repair using GPML.



(b) Signal repair using ARX.

Figure 4.1: Example of signal repair using Gaussian process machine learning (GPML) and time series algorithms (ARX).

tackled in the future.

4.1.2 Training block

As shown in Section 3.2.1.3, a symptomatic curve can be modeled as the junction of two Gaussian curves. The training of the models has been performed for Patient A with $T_A = 15$ randomly chosen symptomatic crisis. The training dataset for Patient B was of $T_B = 8$ migraine events. Each one of these attacks T_i leads to a model M_i . And, according to the stated in Section 3.2.2.2, there will be chosen $M_{best_A} = M_A/3 = 5$ models (first five models in validation) for Patient A, and $M_{best_B} = M_B/3 = 3$ models for Patient B.

State-space models have shown as a very good alternative for migraine predictive models. In the following paragraphs there are going to be presented the results for the implementation of the Algorithm 2 (page 201). It will be shown the test results for the best models. These will be tested with new signals: migraines not used in the training and validation sets.

Each migraine has been trained for 6 different horizons and 5 different feature combinations. Figures 4.2a and 4.2b summarize the training results for patients A and B respectively. Each value on the surface of these graphs represents the average of fits over all the trained models ($M_A = 15$ models for Patient A, and $M_B = 8$ for Patient B).

For Patient A, the fit decreases more quickly than for Patient B. As expected, triads of features have a lower fitness than the four features combination, and the fitness is even lower if duos are studied (not shown). However, as all of the three feature combinations still show acceptable fit values (close to the threshold of 70%), they will proceed to the validation stage. In addition, the training results for patient B are almost 15-20 points higher than the results for Patient A. This can be explained by the higher amount of data lost during monitoring periods of Patient A (in spite of the usage of the `Signal Repair` module).

In Figures 4.2a and 4.2b, maximum fits are reached for lowest horizons (10 and 20 minutes). The fit decreases with the horizon, but also depends on the features selected. The highest values of fit are reached for the combination of the four available biometric variables. It is worth to mention that a valley is found around the prediction horizon of 40 minutes. Surprisingly, this occurs for the TEMP-HR-SpO2 feature combination in both patients. As the number of individuals is not enough, this should not be considered as a conclusion. At this point, the fit for Patient A is 73.2%, and 94.8% for Patient B. This suggests that the time window for prediction is larger for Patient B than for Patient A. Additionally, fits increase with prediction horizons larger than 40 minutes (50 and 60 minutes); this is due

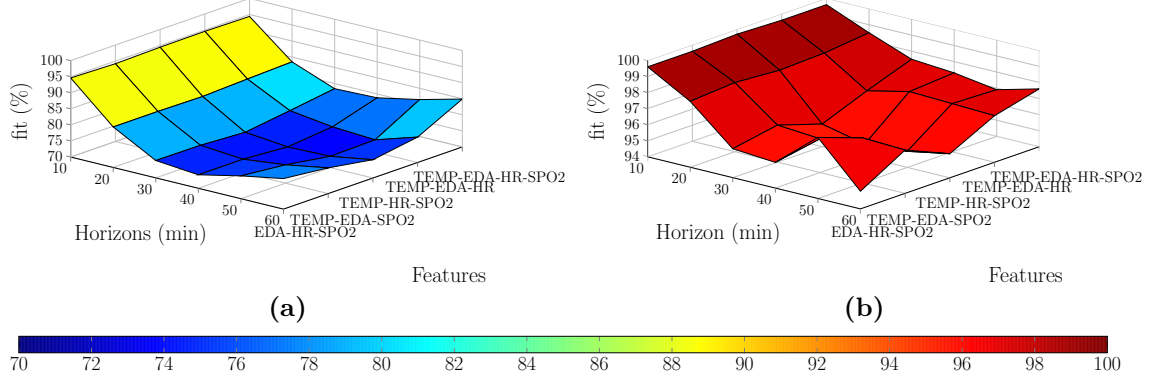


Figure 4.2: Average fits for training. Dependence with the future horizon and selected variables. (a) Data from Patient A; (b) Data from Patient B

Table 4.1: Training results for the TEMP-EDA-HR-SpO2 features set and 40 minutes forward horizon for patients A and B.

	Patient A															Patient B									
	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}	M_{13}	M_{14}	M_{15}	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8		
<i>fit</i> (%)	84.4	84.1	88.1	75.6	70.6	85.0	86.7	65.8	78.9	82.6	79.2	67.4	80.8	81.0	72.0	97.5	99.6	95.9	99.5	90.0	97.0	99.9	98.1		
<i>ph</i> (min)	25	105	60	40	30	30	70	75	15	100	60	95	20	105	90	12	30	14	20	25	25	15	18		
<i>nx</i>	6	4	7	8	5	9	5	6	10	9	7	8	7	4	7	6	9	7	6	10	8	10	9		

to overfitting during training. It seems that the modeling approach in the training stage reaches the limit for the migraine prediction at 40 minutes.

Table 4.1 shows the training results for each model for patients A and B. f is the fit achieved. pw and nx are the best past window (in $\{0, 5, 10, \dots, 100\}$) and the best state transition matrix's order reached by the N4SID algorithm (in $\{1, 2, \dots, 10\}$), respectively. Results for the horizon of 40 minutes—where the valley of accuracy occurs—for the TEMP-EDA-HR-SpO2 combination of features are shown. This setup corresponds to the minimum training error. The fits are 78.8% and 97.2% in average for patients A and B, respectively.

No correlation has been found between order, fit and past horizon; hence, some of the models achieve good fit values with low effort (low matrices' orders and short past horizon with high fit), such as model M_7 for Patient A, while others require a higher effort, such as model M_{10} ; this with a high order, and the large past horizon reaches a lower fit. As can be seen, fits reached are high for both patients, and they are always over 70% (except for two Patient A cases marked in bold in Table 4.1). However, models require large matrices (larger than order $nx = 7$) in most of the cases. Despite the high orders, past horizons are low for Patient B (they are always lower than 30, 20 minutes in average); but they are high for Patient A (61 minutes in average and up to 105 minutes backward).

Table 4.2: # Useful models after validation.

Features / Horizon (min)	Patient A						Patient B					
	10	20	30	40	50	60	10	20	30	40	50	60
TEMP-EDA-HR-SpO2	15	4	0	0	0	0	8	4	1	0	0	0
TEMP-EDA-HR	15	5	0	0	0	0	8	6	1	0	0	0
TEMP-EDA-SpO2	15	5	0	0	0	0	8	6	2	0	0	0
TEMP-HR-SpO2	15	7	0	0	0	0	8	7	4	0	0	0
EDA-HR-SpO2	15	5	0	0	0	0	8	7	1	0	0	0

The fit metric shows how some migraines behave far better than others for a given horizon. Some models show a fit over or equal 80%, and none show a fit lower than 70%. On average, for 40 minutes, these models with four features are able to fit a migraine event with 74.2% and 95.8% accuracy for Patient A and B respectively. For the remaining future horizons and feature sets, the average fit in training keeps high, always over the 70% for both patients.

4.1.3 Validation block

Now the results for model validation are presented. Here, trained models are tested as predictors of the other symptomatic crises of the training dataset. This will allow to analyze the overfitting effect as well. It is performed cross-validation between models, as mentioned in Section 3.2.2.1. Model repair (prediction repair) and linear decision are also carried out in this block. There were two stages during the development of this research: (i) first, it was performed the reparation of the prediction as shown in Algorithm 1 with the linear decision too. Several studies were carried out and the results are shown here for a fixed predictive horizon of 30 minutes. They are presented under the heading of *Simple-model repair Conservative Study* (SCS). (ii) Later, it was developed the methodology to add the Gaussian fitting and thus, more results were drawn; which are presented in the heading of *Complete-signal repair Strategic Study* (CSS).. Both stages (SCS and CCS) generated results not only in the validation bock but also in the model selection and test. It will be indicated properly in each case. The main objective of this is to discard overfitted models. In this way, there will be found those models that reach the longest prediction horizon. Results have been obtained firstly for the 6 different prediction horizons and the 5 feature combinations.

Table 4.2 represents the number of useful models with average fit over all the cross-validations exceeding 70%. As the average prediction is calculated over more than one model, this analysis will help on the selection of the models. It is considered that, at least, one third of the models must validate with high average fit to choose a feature set as relevant (for each prediction horizon). According to

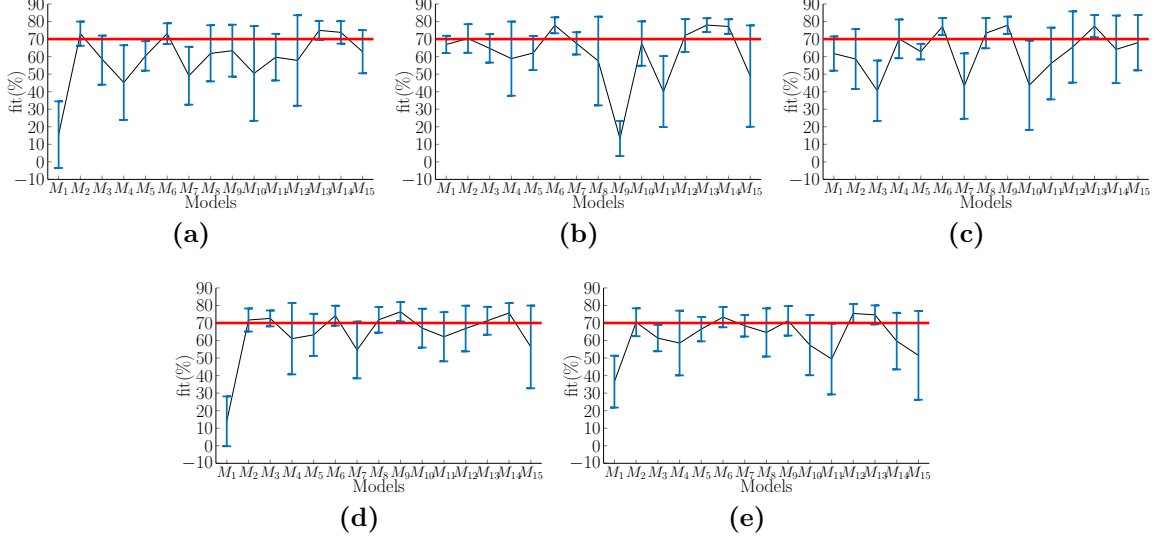


Figure 4.3: Validation results for models from Patient A. 20 minutes forward horizon. For each features set: (a) TEMP-EDA-HR-SpO2; (b) TEMP-EDA-HR; (c) TEMP-EDA-SpO2; (d) TEMP-HR-SpO2; (e) EDA-HR-SpO2.

the results in Table 4.2, no difference appears between the selected features for a forward horizon of 10 minutes. In general, no model is able to validate for higher horizons than 20 and 30 minutes for patient A and B respectively. This confirms that the valley in training in Figures 4.2a and 4.2b marks the limit of prediction for state-space models, and models trained over 40 minutes are overfitted. Remember the reader this point, which we will come back to later.

The four-features combination is always the worst combination. For Patient A and 20 minutes forward horizon, the combinations of three features, except for TEMP-HR-SpO2, show 5 available models (in the limit of our criterion to consider the features as relevant). For Patient B, all the combinations of features look good (more than 3 models over the average fit of 70% in this case) for 20 minutes, but only the TEMP-HR-SpO2 feature combination is useful for 30 minutes. As aforementioned, high fits in training do not assure good models, and some of them must be discarded in the validation phase.

As the 20 minutes prediction horizon seems to be the safest horizon, we use this to show the results for Patient A in Figures 4.3a through 4.3e and for Patient B in Figures 4.4a through 4.4e. Horizontal axes in these figures represent each one of the validated models. Vertical axes represent the average fit achieved, obtained as the average of the $M - V_{ov} - 1$ validations. The whiskers represent the standard deviation, $\sigma.V_{ov}$, are the overfitted validations (negative fit). These were removed to calculate the average. The red line indicates the threshold set as fitting criterion.

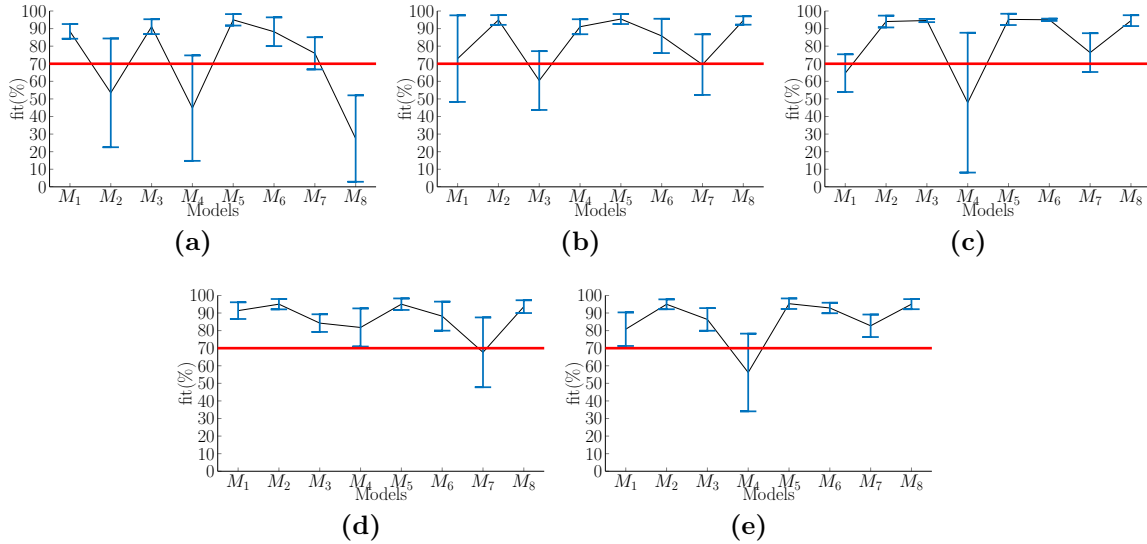


Figure 4.4: Validation results for models from Patient B. 20 minutes forward horizon. For each features set: (a) TEMP-EDA-HR-SpO2; (b) TEMP-EDA-HR; (c) TEMP-EDA-SpO2; (d) TEMP-HR-SpO2; (e) EDA-HR-SpO2.

The deviations (the whiskers) for validations in Patient B (σ_B) are lower than deviations in Patient A (σ_A). This means that, the confidence of models from Patient B should be higher than from Patient A. We can also state that these models are more generalizable because the results for Patient B are more consistent than those for Patient A, as data acquired from Patient B have less discontinuities during monitorization.

Regarding the average values in Figures 4.3a through 4.3e, as the four-features combination is a poor election, only 4 models have an average fit higher that 70%. For the TEMP-HR-SpO2 combination of features (Figure 4.3d) we achieve the best results. In this case, 7 models exceed the threshold of 70%. Something similar occurs with the results for Patient B.

As aforementioned, to calculate the average fit, validations with negative results have been removed. In some cases, the number of useful validations is really low. This happens, for example, with the validation of the model M_9 in Figure 4.3b, that only validates 3 migraines. The model M_1 for Patient B validates also the same number of migraines in Figure 4.4a, and only 2 migraines are validated in Figure 4.4b and Figure 4.4c (despite its high fitting).

As a result of the validation study: (i) the four-features combination is never the best option to predict migraines for any horizon length, and (ii) it seems that 20 minutes forward is the best window to predict migraines for both patients. The first idea means that some biometric variable worsens the prediction in

combination with others (but not itself). Hence, by removing one variable we achieve more useful models. The second result achieves a prediction horizon close to the constraints imposed by the pharmacokinetics of the drugs.

Let us summarize the study above into a key point: in view of the fact that models trained at 40 minutes are overfitted, it has been decided to use predictive models at 20 minutes, which do work but the predictive horizon seems a little bit low for the pharmacokinetics of the drugs (30 minutes in the worst case).

At this point, to tackle this issue, the two stages developed in this research are introduced. As aforementioned, we refer to them as the *Simple-model repair Conservative Study* (SCS), and the *Complete-signal repair Strategic Study* (CSS): (i) SCS uses the M_{best} predictive models in validation at 30 minutes (the horizon expected to be reached according the pharmacokinetics) and forces them to predict closer and further taking average predictions. SCS applies only signal repair (Algorithm 1) and linear detection. (ii) CSS contemplates the use of one of the schemes proposed in Figure 3.15, repairing signals and taking a set of M_{best} models too, to lead to an average prediction and a strategy for decision making. CSS applies both signal repair and Gaussian fitting..

The results of the validation of the SCS are shown below for both patients and all features combinations at 30 minutes. Section 4.1.3.1 focuses on the study of the schemes proposed for the CSS.

4.1.3.1 Model Repair: improving predictions

This section is devoted to improve the prediction horizon. In this section, the results of the schemes proposed in Figure 3.15 in Section 3.2.3 are shown.

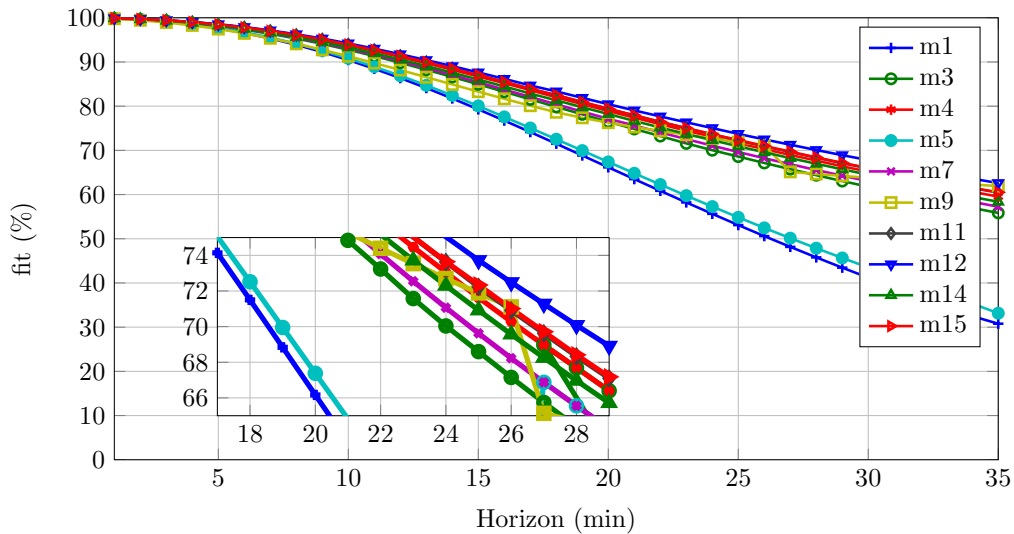
Simple-model repair Conservative Study (SCS)

Here, for Patient A only, it is validated each M_{A_i} ($i = 1, 2, \dots, 15$) model obtained for each j -th migraine ($i \neq j$). Despite the models have been trained at 30 minutes, the validation is performed for different prediction horizons: from 1 to 100 minutes. These validations have been obtained after repairing the models (Proposal 1 of Figure 3.15 in page 67). The results are shown in Table 4.3 for fits at 70%. The effort to reach fittings greater than 80% in the training stage is hard, as shown in Figure 4.2a.

Table 4.3: Simple-model repair Conservative Study. Validation for every M_i model from Patient A. Future horizons for fit at 70%.

	M_6	M_{13}	M_2	M_8	M_{10}	M_4	M_{15}	M_{11}	M_9	M_{12}	M_3	M_7	M_5	M_1	M_{14}
$fh_{average}$	22	20	20	20	16	15	14	14	13	13	12	9	5	5	5
fh_{min}	18	19	17	17	10	10	9	8	10	8	7	4	3	2	1
fh_{max}	25	21	22	24	22	20	19	29	14	21	15	11	7	8	10

Models have been sorted according to the average horizon and then according to the minimum horizon required to ensure the higher minimums. Each model was trained with one migraine and 30 minutes of prediction horizon; hence, validation over other migraines will not reach this horizon, as they have not been used in the training of the model. The validation of one model generates 14 fitting curves with 100 points each, as previously mentioned. These curves reach 70% of fit at some point; these crosses lead to a minimum and a maximum horizon, and hence, an average value represented in Table 4.3.

**Figure 4.5:** Validation applying the average model for Patient A. Average model with M_{best} models applied over the remaining 10 migraines.

According to these results, it is wondered if the best model, M_6 , is good enough for performing the migraine prediction. As seen in Table 4.3, the maximum horizon is not achieved by M_6 , but for M_{11} . This possibility was taken into account in the design of the experimental setup, and a ranking of the models is performed. With this purpose, the best M_{best_A} models (first five models in Table 4.3) are chosen to define an average model—average prediction actually. This works as follows: for each migraine, each model $M_{best_i,A}$ is applied. Five predicted symptomatic curve are achieved. The result is the average of these five validations for a given horizon.

Table 4.4: Simple-model repair Conservative Study. Prediction horizons for all the combinations of features for Patients A and B.

	TEMP-EDA-HR-SpO2		TEMP-EDA-HR		TEMP-EDA-SpO2		TEMP-HR-SpO2		EDA-HR-SpO2	
	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B
$fh_{average}$ (min)	24	28	25	35	24	47	23	45	25	29
fh_{min} (min)	19	17	16	28	19	43	17	37	19	25
fh_{max} (min)	28	32	31	50	28	52	32	52	34	33
fh_{σ} (min)	3	6	5	10	3	4	4	6	4	3

At the end, the false positives are removed with the `Model Repair` submodule. The result is shown in Figure 4.5. Axis x represents the prediction horizon, meanwhile axis y represents the fit between the average prediction and the real symptomatic curve. In this process, the average, minimum and maximum horizons achieved at 70% are: 25, 18 and 28 min, respectively.

Notice that, while some models such as models M_1 , M_3 and M_7 exceeded 80% of the fit in the training stage (achieving 82.0%, 85.2% and 82.9% respectively), they have not been selected as part of the best models in the validation. This is due to these models being overfitted.

The same process has been followed for all combinations of three features—training, validation, ranking, best models selection—and, finally, the average model technique is applied. The results will show the most useful features to predict a symptomatic curve for Patient A. These results will also show the importance of the features in the prediction.

In Table 4.4, a summary of the results is shown. It introduces data from Patient B in order to check how good or generalizable the training and validation stage developed is. The analysis of this table brings meaningful results. First of all, it can be observed how the inclusion of some features in the model worsens the prediction. For instance, for Patient A, the absence of the EDA features worsens the average prediction horizon in 2 minutes. On the other hand, the remaining three combinations of features maintain or improve the average and maximum prediction horizon. For Patient B, this does not occur in the same way. In this case, the worst scenario occurs for the four feature combination; in the others, the absence of one feature improves the prediction horizon. In addition, it can be noticed that the models of Patient B result in better predictions than the models of Patient A. The best result is achieved for the TEMP-EDA-SpO2 feature combination, with up to 52 minutes of a horizon and an average of 47 minutes. The remaining three feature combinations are almost as good as this. In most of the cases, the average horizons achieved exceed or they are close to the desired 30 minutes for both patients.

The analysis of these results shows how not only the selection of the acquired

Table 4.5: Results of the prediction repair methodologies. F_{score} to compare the schemes proposed for the methodology

	Base		Repair		FitGauss		Repair + FitGauss		FitGauss + Repair	
	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B
TPR (%)	24.7	30.4	29.8	36.3	31.0	39.0	33.1	39.6	30.6	39.0
PPV (%)	45.1	60.7	57.2	90.4	75.1	71.6	80.5	81.1	78.5	87.9
F (%)	31.9	40.5	39.1	51.8	43.8	50.5	46.9	53.2	44.1	54.0

variables, but also that their combination in the matrices of the N4SID models can result in better or worse predictions. Furthermore, it can be observed how the selection of features (the selection scheme explained further below) changes for every patient, providing a different set of models per patient; even when the best set of features is the same for both. So, the use of a set of models trained at 30 minutes is quite feasible, and it brings them to the limit in order to lead to higher horizon and more robust predictions.

Complete-model repair Strategic Study (CSS)

There have been studied all the repairing schemes during the validation stage using the four-features combination (TEMP-EDA-HR-SpO2) and the 6 prediction horizons (from 10 to 60 minutes). The F_{score} is used as the metric to compare all the schemes with the basic one: the *Simple-model repair Conservative Study* (first scheme in Figure 3.15). To compute the F_{score} , the sensitivity (TPR) and the precision (PPV) values are calculated. All values are based on the results of all the $M - 1$ predictions of each M_i model. This means that the true positive (TP) account should be ideally $6 * M * (M - 1)$. The results are shown in Table 4.5.

Table 4.5 shows low levels of the F_{score} because it has been calculated as the average of the F_{score} s for every horizon. The higher the horizon, the lower the F_{score} , worsening this average F_{score} . The results show a high rate of false positives and low number of detections for horizons higher than 40 minutes, as expected from the training in Section 4.1.2.

The best scheme for the proposed methodology is the combination of repairing the prediction (remove spurious) and the Gaussian fitting. The order (first repair then fitting) affects more to Patient A than to Patient B. Therefore, the scheme Repair+fitGauss in Figure 3.15 is chosen as the repairing scheme of symptomatic crises prediction.

Now, the selected scheme is applied and compared with the base scheme. For the sake of simplicity, only the results for Patient B are presented. Figure 4.6a presents the results of validation for Patient B and all the trained

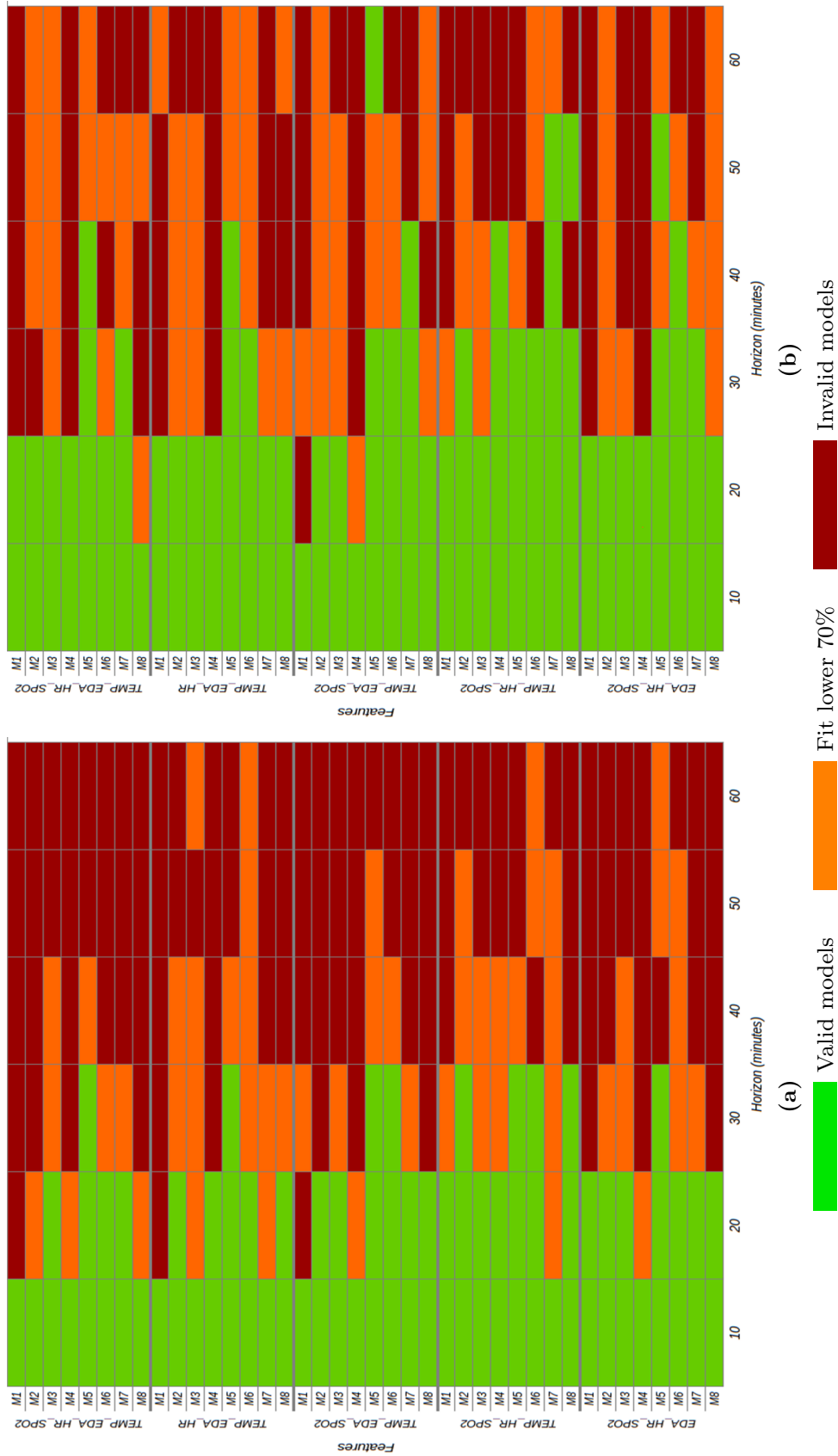


Figure 4.6: Comparison of SCS and CSS results. Useful models in validation for Patient B at a 70% of fit. (a) Without reparation of prediction; (b) After reparation of prediction and Gaussian fit.

future horizons (10 to 60 minutes forward). Horizontal axis represents the six different horizons trained and the vertical axis represents the $M = 8$ models for each feature combination. Colors represent those models good enough to be used as predictors in a real time implementation (green), those with an average fit lower than 70% (orange) and the discarded ones because the overfitting (red, less than one third of the migraines available are validated).

All models validate all migraines for a prediction horizon of 10 minutes. As aforementioned, for 20 minutes forward almost all models are useful, except model M_1 for some feature combinations. From 30 minutes ahead, there are not enough useful models, except in the TEMP-HR-SpO2 feature combination, where 4 models validate quite well. For prediction horizons equal and greater than 40 minutes, migraine prediction is not possible, as pointed out in Section 4.1.2.

Applying repairing techniques to the prediction can increase the prediction horizon. In this case it has been applied reparation of the prediction and Gaussian fitting, in this order. This is proved by Figure 4.6b, again for Patient B. The average prediction horizon has been incremented in 10 minutes (compared to Table 4.2), and some models validate migraines with a future horizon equal to 40 minutes. There are improvements in models for most of the prediction horizons and all combination of features. These increments are due to removing false positive detections, negative values, and values higher than the maximum, 100, in the normalized symptomatic pain curve.

Regarding results for Patient A, the improvements achieved have been lower. Although some more models are useful for 20 minutes, no one is useful for 30 minutes of prediction if 70% of fit is expected (validating, at least, $\lceil M/3 \rceil$ of the symptomatic crisis in the training dataset).

As shown, the prediction horizon can be improved applying repairing techniques to the predicted signal, reaching prediction times close to the current time of pharmacokinetics and even exceeding it. Additionally, it has been shown a method to test the limits of a given predictive model, that must be applied for each patient individually. In particular, it has been found that the maximum prediction horizon for Patient A is in the interval $[20, 30]$ minutes, and in the interval of $[30, 40]$ minutes for Patient B, the same results as previously shown in Table 4.4 for results of the *Simple-model repair Conservative Study* forcing 30 minutes predictive models to predict beyond.

Finally, in view of the results obtained, it can be said that both simple-model repair and Gaussian fitting (SCS and CSS) are useful techniques to extend the limits of prediction horizon of predictive models.

Table 4.6: Simple-model repair Conservative Study. Robustness analysis. Horizons of prediction for fit at 70%.

	Patient A						Patient B					
	EDA (s)		HR (bpm)		SpO2 (%)		TEMP (°C)		EDA (s)		SpO2 (%)	
	0	25	0	100	0	80	0	35	0	25	0	80
$fh_{average}$ (min)	22	23	11	18	11	20	11	38	46	41	11	28
fh_{min} (min)	18	18	10	15	9	16	10	33	42	37	9	24
fh_{max} (min)	26	33	11	19	12	30	12	45	50	46	13	30
fh_{σ} (min)	3	4	1	2	1	4	1	6	3	3	2	3

* Disconnection means that data do not exist, 0 volts that lead to 0 in each magnitude.

** Saturation levels have been set according to the recordings in our database. HR higher than 100 is quite common while doing exercise, but there are not such a kind of monitoring periods in our database.

4.1.4 Model Selection block

Previous section was devoted to improve the prediction horizon. To motivate the need of that module and its relation with the **Model Selection** block, the models are going to be forced to extreme situations to evaluate the robustness against failing sensors (total or partial loss and saturation). Later, the strategy for the selection of models based on the Sensor-Dependent Model Selection System ($SDMS^2$), is shown.

To force models to extreme situations, for illustrative purposes, let's take the best feature combination, EDA-HR-SpO2 for Patient A and TEMP-EDA-SpO2 for Patient B—as well as their selected models M_{best_A} and M_{best_B} —for the 30 minutes prediction horizon and simple model repair (SCS).

Table 4.6 shows the average, upper and lower prediction horizons (in minutes) achieved in different situations of failing sensors (again, for a fit reference of 70%). As in Table 4.4, the $M - M_{best}$ migraine crises (not used for the average model) are used to validate the robustness experiment. As seen, the best cases of prediction (34 and 52 minutes for Patient A and Patient B, respectively) are never reached in the case of lossy (variables equal zero) sensors or saturation. Only for Patient B, the maximum is almost reached when the temperature and EDA sensors are saturated. The average horizons (25 and 47 minutes for Patient A and Patient B, respectively) are not exceeded either, and only for Patient B, the average horizon reaches 46 minutes when the EDA sensor is lost. So, a strategy for model selection is needed in order to detect failures in sensors and act to maintain the predictability of the system.

4.1.4.1 Sensor-Dependent Model Selection System ($SDMS^2$)

At this point, we can introduce the Sensor-Dependent Model Selection System ($SDMS^2$). This system, shown in Figure 4.7a, is able to detect saturated or lossy sensors. The $SDMS^2$ senses the status of the sensors and chooses the best set of models according to their availability in real time. The $SDMS^2$ is very conservative and bases its strategy only in the accuracy of the models, sorting them from better to worse quality.

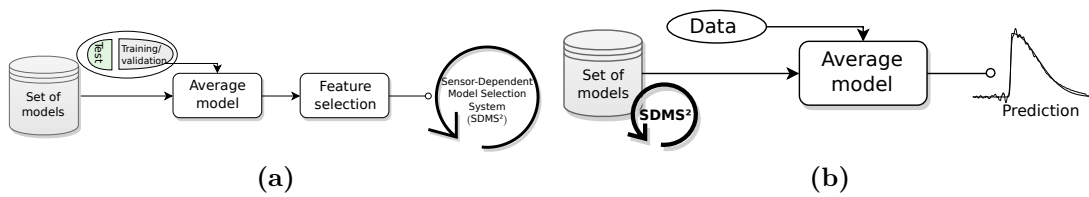


Figure 4.7: $SDMS^2$ design and usage for the real-time applications. Results using the SCS. (a) Sensor-dependent model selection system ($SDMS^2$); (b) Implementation of the system for real-time applications.

For each patient this system implements—after creating the whole set of models for every combination of features—a hierarchy of sets of models, depending on the availability of sensors. The hierarchies of sets of models for Patient A and Patient B are shown in Figure 4.8. The ordination is represented from top to bottom; vector h represents the information of minimum, average (highlighted) and maximum horizons from Table 4.4.

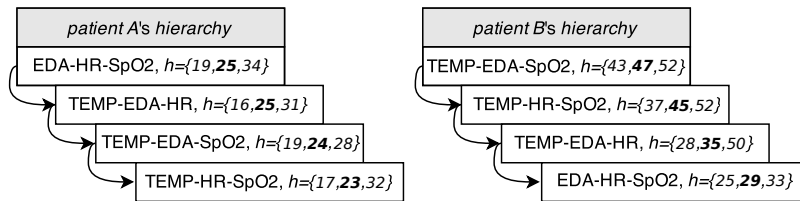


Figure 4.8: Hierarchies of sets of models for Patient A and Patient B using the SCS.

In an ambulatory monitoring, patients wear the four sensors along the day. The selected model (and set of features) for the individual will be applied to predict the migraines. If one of the sensors fails or saturates, then the next model according to the set of available features will be selected, trying to maintain the prediction horizon. It is possible to eliminate one of the sensors if the patient desires it; however, it is worth noting that, if one sensor fails or saturates, the prediction horizon will be lower, according to Table 4.6.

4.1.4.2 Advanced Sensor-Dependent Model Selection System

For the results that have been presented in Section 4.1.4.1, the fit value has been prioritized, setting a fixed value of 70% and observing the achieved prediction horizon. In this section we present a dual strategy to select the models: (i) regarding the fit, or (ii) regarding the prediction horizon. In all the cases we always maintain that a model is considered good when it is able to validate at least $\lceil M/3 \rceil$ of the migraines from the dataset at a given fit (criteria stated in Figure 4.6b). The number of migraines to validate are 3 for Patient B (8 migraines available in the training dataset) and 5 for Patient A (15 migraines available in the training dataset). To avoid overfitting and to calculate the average prediction, we still consider as good the selection of, at least, $\lceil M/3 \rceil$ migraines for each feature combination.

Figure 4.9 shows the number of models available for every horizon at a desired fit level. As a reference, the vertical bars mark the fit where $\lceil M/3 \rceil$ models can predict at least $\lceil M/3 \rceil$ of the migraines in the training dataset. If we focus more on the prediction, (i) the first strategy works setting a desired horizon and looking for the best feature combination that reaches the maximum fit. On the contrary, (ii) if we focus more on the fit level, the second strategy works setting a desired fit and looking for the feature combination that reaches the farthest horizon. This is a more conservative selection, for which we set the desired fit or goodness of the prediction and we settle down the available horizon.

For instance, regarding Figure 4.9, if we are looking for the best prediction horizon 20 minutes forward, we should use the models calculated with EDA-HR-SpO2 in Figure 4.9e, because the second vertical bar has a higher fit for this feature combination than for the others. There, there are found 3 models validating at least 3 migraines each one, in average, with a 97% of fit. But if a prediction of 30 minutes forward is desired, the best option is to select the models using the TEMP-HR-SpO2 feature combination in Figure 4.9d. For 50 minutes forward we will select the models using TEMP-EDA-SpO2 Figure 4.9c, and accept only a 50% of fit.

Figure 4.9 can also be used to look for a desired fit. For example, if the HR sensor is not available (Figure 4.9c) and we look for predictions with a fit equal 60% or higher, it could be only satisfied for a horizon of 30 minutes. On the other hand, if the EDA sensor is not available (Figure 4.9d), for the same minimum fit of 60% we might predict up to 40 minutes.

It is worth noting that Figure 4.6b is just a representation of the Figure 4.9 at 70% of fit. In this figure, it can be seen at 1% of fit how many models that are not able to validate more than 3 migraines still remain.

This methodology leads to a versatile tool for the improvement of predictions

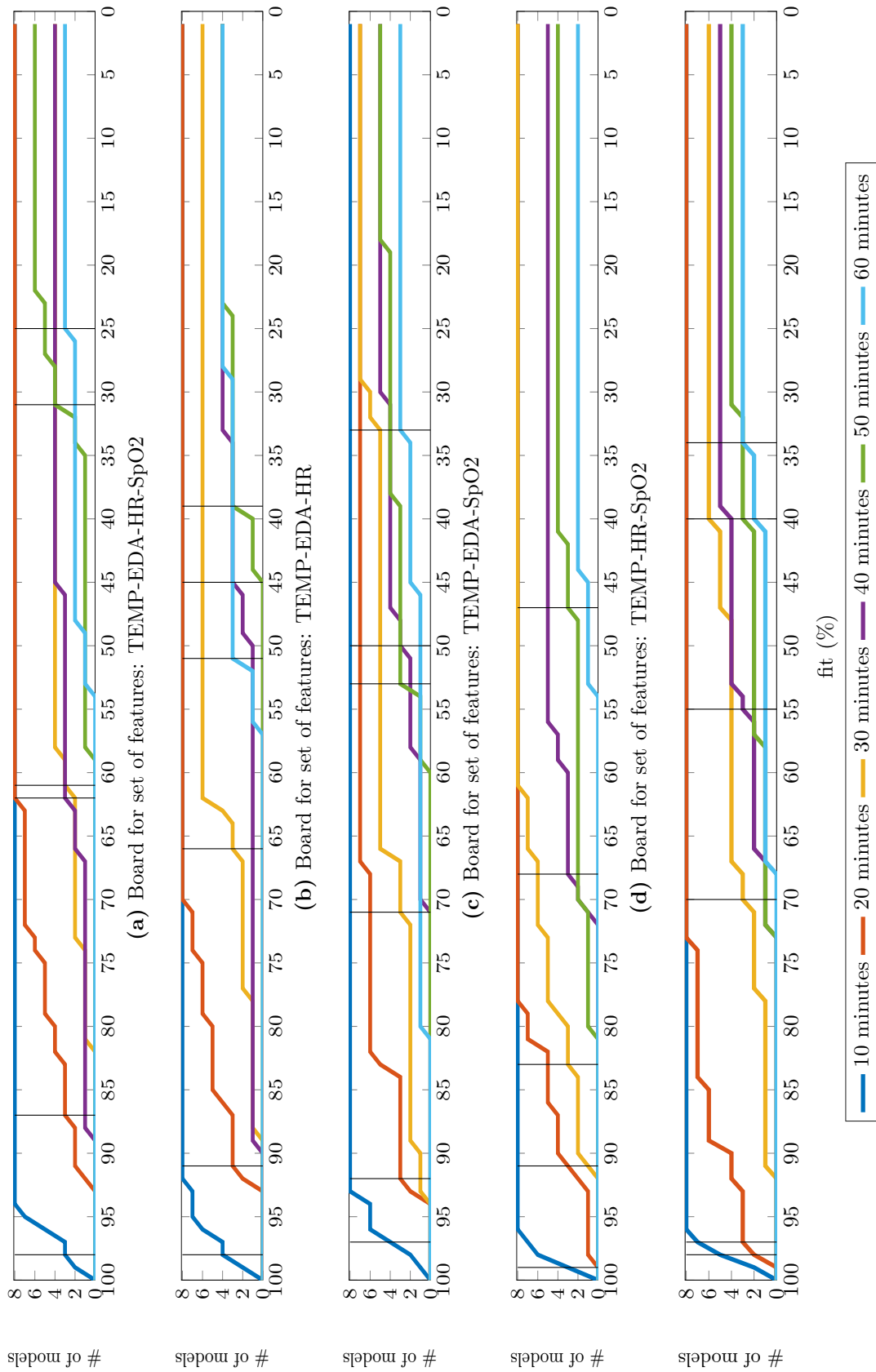


Figure 4.9: Board of strategies for model selection for Patient B. Results using the Complete-signal repair Strategic Study.

and the selection of models for predictions of symptomatic crises in ambulatory real environments. The effectiveness of the solution is studied, and the results have proved to meet the pharmacokinetics limits required to avoid the negative effects of symptomatic crises. The results also show that for Patient A the limits of predictions are between 20 and 30 minutes, and between 30 and 40 minutes for Patient B, achieving fits of 70% in both cases. In the next section the methodology is applied to the test datasets.

4.1.5 Test of the Sub-system I

In this section, we present some test results. The results are divided, as in previous sections, in results obtained under the *Simple-model repair Conservative Study* (SCS), and under the *Complete-signal repair Strategic Study* (CSS). The average models have been applied to the remaining migraine episodes of the dataset—five for Patient A and four for Patient B—5 and 6 asymptomatic periods of time for Patients A and B respectively.

To evaluate the results the statistical F_{score} is used (Eq. 4.3). A true positive (T_p) is considered when a detection is achieved and the fit in the migraine period is higher than or equal to 70%. As was described in Algorithm 1, values above 50% of the probability of the pain curve are marked as positives.

Simple-model repair Conservative Study (SCS)

All tests have been run with the $fh_{average}$ achieved for each feature combination up to a maximum of 30 minutes (see Table 4.4) and applying the real-time stage in Figure 4.7b.

Table 4.7: SCS. Test results: trust levels on models selected for a fit reference of 70%.

Features	Patient A			Patient B		
	TPR (%)	PPV (%)	F_{score} (%)	TPR (%)	PPV (%)	F_{score} (%)
TEMP-EDA-HR-SpO2	60	100	75	50	100	67
TEMP-EDA-HR	67	100	80	60	100	75
TEMP-EDA-SpO2	47	100	64	90	100	95
TEMP-HR-SpO2	53	100	70	90	100	95
EDA-HR-SpO2	67	100	80	40	100	57

Table 4.7 shows the results obtained for both patients. The confidence in the selected models is good enough for most of the sets of features. Only the TEMP-EDA-SpO2 combination for Patient A and the EDA-HR-SpO2 combination for Patient B present quite poor results. Avoiding these cases, in general, the TPR

Table 4.8: Generalization of models for other patients for the reference fit of 70%.

Features	Patient A Models over Data from Patient B			Patient B Models over Data from Patient A		
	TPR (%)	PPV (%)	$F_{score}(\%)$	TPR (%)	PPV (%)	$F_{score}(\%)$
TEMP-EDA-HR-SpO2	0	100	0	0	100	0
TEMP-EDA-HR	22	64	33	0	100	0
TEMP-EDA-SpO2	44	100	62	0	100	0
TEMP-HR-SpO2	33	100	50	0	100	0
EDA-HR-SpO2	33	73	46	0	100	0

is above 70%. In addition, the PPV is 100% for both patients; so, the predicted migraines are all truly migraines. The best case for Patient A is for the TEMP-EDA-HR and EDA-HR-SpO2 combinations with an $F_{score_A} = 80\%$. For Patient B, the combinations TEMP-EDA-SpO2 and TEMP-HR-SpO2 with $F_{score_B} = 95\%$. This does not mean that these are the features that achieve the best horizons (as seen in Section 4.1.3), but they are the most reliable. Notice that the best set for both patients (EDA-HR-SpO2 for Patient A and TEMP-EDA-SpO2 for Patient B) are the most reliable ones. The less reliable combinations of features are allocated at the end (or nearly) of the hierarchy. None of the asymptomatic periods leads to false positives for any set of features.

Generalization of models for other patients

How generalizable are models for other individuals? Lets apply the models selected from one patient to all of the migraine events from another. Asymptomatic time periods are also taken into account, in order to check if baseline periods are equal for everyone. Again, a true positive (TP) is considered when a detection is achieved and the fit in the migraine period is higher than or equal to 70%. Results are shown in Table 4.8.

For Patient A and the combination of features TEMP-EDA-SpO2, models are capable of detecting some migraines from Patient B (a TPR of 44% means that this percentage of Patient B's tested migraines have been detected). Nevertheless, this does not occur again in other sets of features, and the remaining F_{score} are quite low.

The results for Patient B are good enough: no detections with a fit higher than 70% lead to false positives during symptomatic periods. The maximum fit reached is 41.3% for the TEMP-EDA-SpO2 set of features. No asymptomatic period produces detections. For Patient B, if the calculated average horizons (larger than 30 min) are applied the TPR and F_{score} fall to the 40% and 53% on average, respectively. This means that only around 40% of migraines will be detected. This low rate is accepted, due to these average horizons being quite

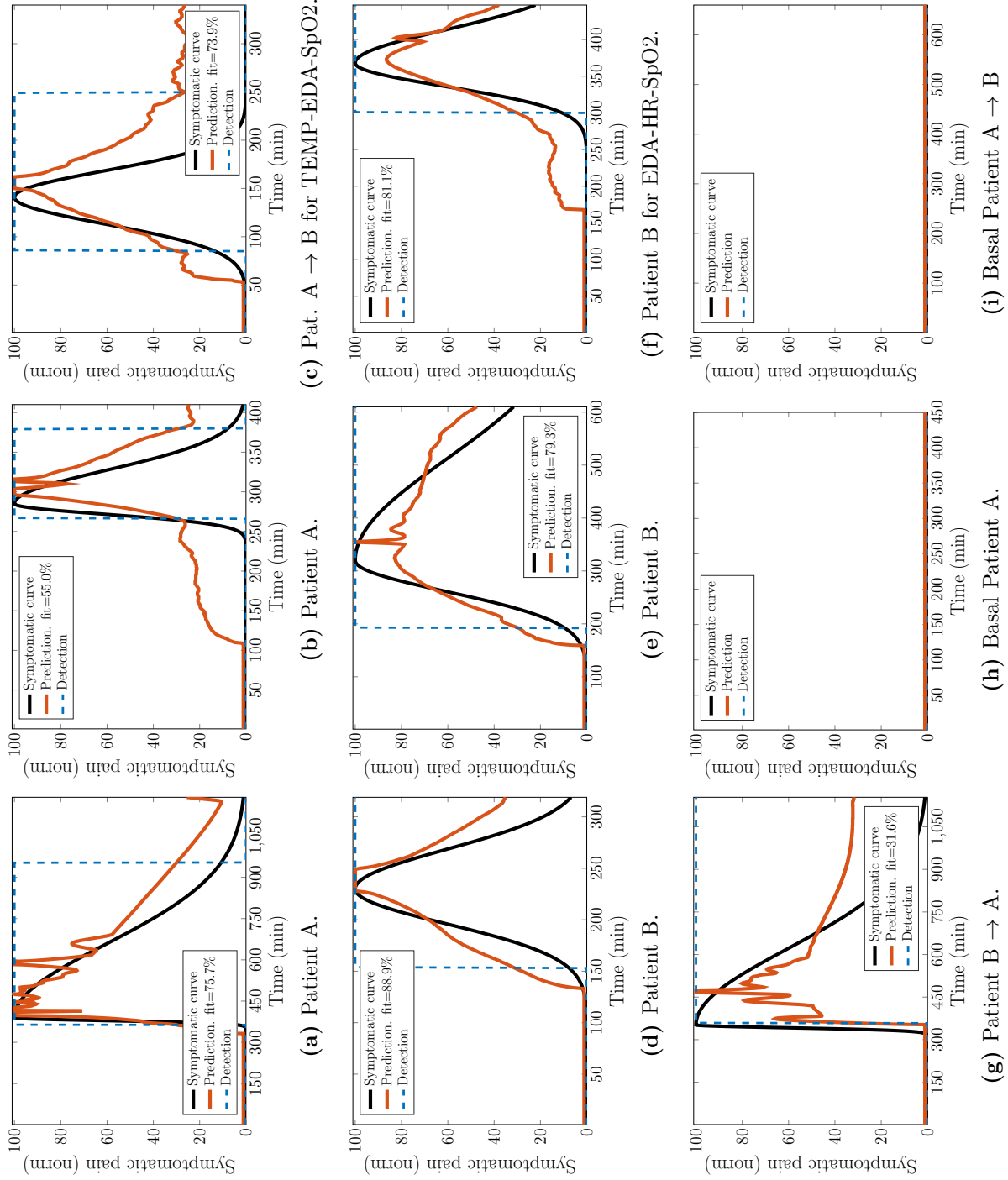


Figure 4.10: SCS. Test results: symptomatic and baselines periods for the trained patients.

high, and it is hard to generalize them.

These results support the initial idea that models must be trained per patient, as it was assumed from the beginning in accordance with the fact that the behavior of the autonomous system depends on the patient. None of the asymptomatic periods led to false positives. This brings the conclusion that variations really appear in the hemodynamic variables measured before the pain starts, and these changes are patient dependent.

Figure 4.10 shows several results for the cases previously mentioned. All results correspond to the first set of features for both patients (EDA-HR-SpO2 for Patient A and TEMP-EDA-SpO2 for Patient B), unless something different is mentioned. Remember that all of the tests have been run with the $fh_{average}$ achieved in validation up to a maximum of 30 minutes. The orange solid line represents the prediction response of the average model. The rectangular blue dotted line is the detection made by the linear decider. It provides the prediction at the 50% of probability of pain.

Figure 4.10a,b shows the best and the worst test cases for Patient A: 75.7% and 55.0% of fit, respectively. Notice that in Figure 4.10b, the migraine has been detected, but our metric—the fit—evaluates this case as no detection, due to the delay in the detection. Figure 4.10c represents one of the detections achieved for the set of features (TEMP-EDA-SpO2) with the average model for Patient A over a migraine of Patient B. As can be seen, the prediction and the detection are good enough, and this leads to a high F_{score} in Table 4.8.

Figure 4.10d,c shows the best and the worst test cases for Patient B: 88.9% and 79.3% of fit, respectively. Figure 4.10f represents the best detection made for the set of features EDA-HR-SpO2 with models from Patient B over a migraine of the same patient. As previously mentioned, this set is less reliable for Patient B; here is presented a good detection. Figure 4.10g represents one of the fault detections with the average model for Patient B over a migraine of Patient A. This detection has a fit of 31.6%; nevertheless, it seems that the prediction is good, at least for the start of the migraine; but the prediction is not able to reach the maximum and follow the decay, and it fails.

Figure 4.10h,i represents no detections over asymptomatic periods. In the first one, the average model from Patient A runs over an asymptomatic period of the same patient. The second one, on the other hand, runs over an asymptomatic period from Patient B.

As expected, the provided model seems be patient oriented and fails to predict the migraine for other patients who are not included in the training of the model for the metric—fit—used. However, this fact can be explained because the dysautonomia is differently affected in migraine patients, and a general

model of the migraine cannot be found at this stage of the research.

Complete-signal repair Strategic Study (CSS)

Tests have been run using the average model in this case too. This is the average of the prediction given by the best $\lceil M/3 \rceil$ models for each feature combination. Over each prediction, as well as over the average of these, the selected improvement scheme has been applied: spurious removal and Gaussian fitting.

Table 4.9: CCS. Test results for Patients A and Patient B at 20 and 30 minutes of prediction horizon respectively at 70% of fit.

	TEMP-EDA-HR-SpO2		TEMP-EDA-HR		TEMP-EDA-SpO2		TEMP-HR-SpO2		EDA-HR-SpO2	
	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B	Pat. A	Pat. B
TPR (%)	50.0	90.0	80.0	100	100	70.0	70.0	100	90.0	40.0
PPV (%)	100	57.1	100	90.0	100	70.0	100	100	90.0	40.0
F (%)	66.7	47.1	88.9	90.0	100	77.8	82.4	100	90.0	47.1

A summary of the results is shown in Table 4.9. As expected, for Patient B, the results of the F_{score} follow the trend of the vertical bars in Figure 4.9, and the best results are achieved for the feature combination TEMP-HR-SpO2. Best results for Patient A are achieved for the feature combination TEMP-EDA-SpO2. Besides, the worst results are achieved, for both patients, with the combinations of four features. This leads to conclude that the best model selection depends on: (i) the features used, (ii) the desired horizon or (iii) the desired fit.

The results have been calculated using only data from two patients; therefore, any generalization of the clinical conclusions obtained by this study could be risky. However, the presented methodology, aim of this work, can be validated by these results. In addition, it has been shown that an analysis of the prediction horizon is needed in order to improve the accuracy of the results, supporting our initial hypothesis.

Figures 4.11a through 4.11c shows some test results for Patient A and Figures 4.11d through 4.11f for Patient B. Several models applied over different feature sets are presented to show the accuracy of the trained models. For all the graphs, (i) black curves represent the original symptomatic curve that must be predicted, (ii) the orange curves are the final result after the reparation of the prediction and the Gaussian fitting.

When a migraine occurs, models provide a prediction of some symptomatic pain levels hours before the pain starts. Nevertheless, these are false positive predictions, and the repairing process removes them. The same happens with

negative predictions or those levels higher than 100. For all cases, repairing the prediction and applying a Gaussian fitting leads us to improve the fit. Figures 4.11c and 4.11f represent asymptomatic periods of time. The latest present a false positive event not removed, obtained from prediction using the TEMP-EDA-HR-SpO2 feature combination, that presents a high false positive rate. The usage of the board of strategies to select the best models (Figure 4.9) would have avoided these false positives.

4.1.6 Fine-grained predictive modeling using GE algorithms

Until now, we have shown the predictive methodology with the N4SID algorithm. However, as aforementioned, we also wanted to check other heuristic techniques like GE. In this Section, we show the results after applying our methodology based on GE for the prediction of migraines. First, the results of the training of the models are drawn, followed by the cross-validation results. After that, we present how these results can be benefited from the application of improvement techniques, and how the model selection phase is conducted. The dependency of the models with the inputs is studied, in order to select those suitable for implementation in a real monitoring device. With the selected models, a test is performed with the test dataset. Finally, an example of robustness against sensor failure is shown.

4.1.6.1 Training of models

After several tests, each model has been trained along 150000 generations with a solely objective: the minimization of the NRMSE. The four available hemodynamic variables (i) skin temperature ($u_1[k]$), (ii) electrodermal activity ($u_2[k]$), (iii) heart rate ($u_3[k]$) and (iv) oxygen saturation ($u_4[k]$) have been used as inputs. The output of the GE, $y_p[k]$ tries to fit the Gaussian adjustment of the real subjective pain, $y_r[k]$. Six different prediction horizons have been computed: from 10 to 60 minutes in step of 10 minutes, and no restrictions in the data selection have been imposed to the algorithm. 15 migraine events perform the training dataset for Patient A and 8 migraines for Patient B.

Regarding previous results using state-space algorithms, it has been hypothesized that models to predict migraines must be derived per patient, and thus, both patients are going to be trained separately. The training results for Patient A and B are shown in Figures 4.12a and 4.12b respectively. In these figures each circle represents the fit achieved by each model, and for every prediction horizon. The horizontal black lines represent the average fit of all the models for every horizon. As expected, in average, lower fits are achieved for

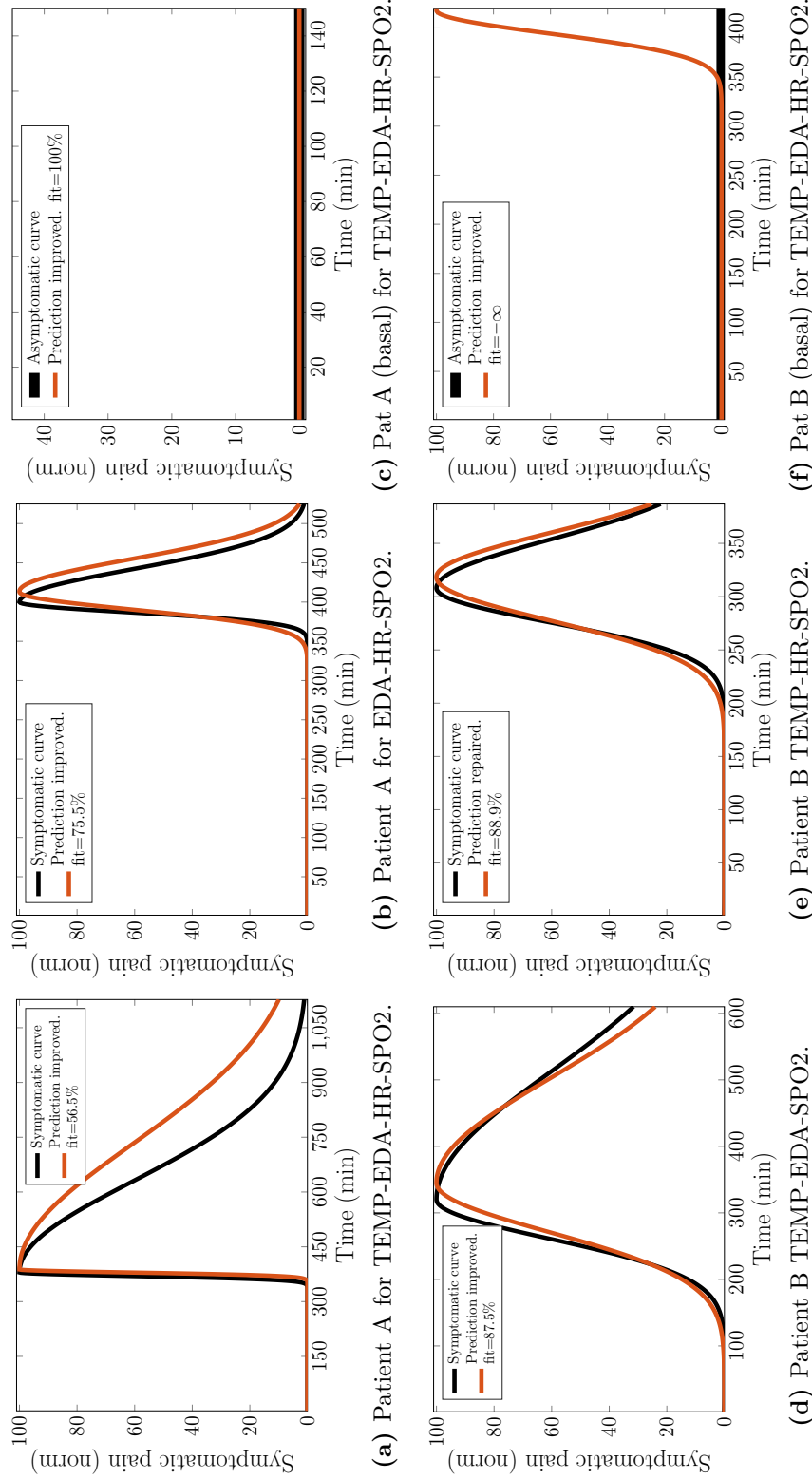


Figure 4.11: CSS. Test results: symptomatic and baselines periods for the trained patients. (a-c) Models for Patient A. Prediction horizon: 20 minutes; (d-f) Models for Patient B. Prediction horizon: 30 minutes.

larger prediction horizons. In this case, again, a fit of 70% represents a good level of similarity and, with this threshold, the best results for Patient A are for 10 minutes of prediction horizon. The results for 20 minutes achieve a slightly lower fit ($65.8 \pm 8.7\%$ on average). Models trained for Patient B achieve better results, with average fits higher than 70% for 10 and 20 minutes of prediction and 65.615.0% of fit for 30 minutes (see Figure 4.12).

To remove overfitted models in training, a cross-validation is going to be performed in the next sub-section.

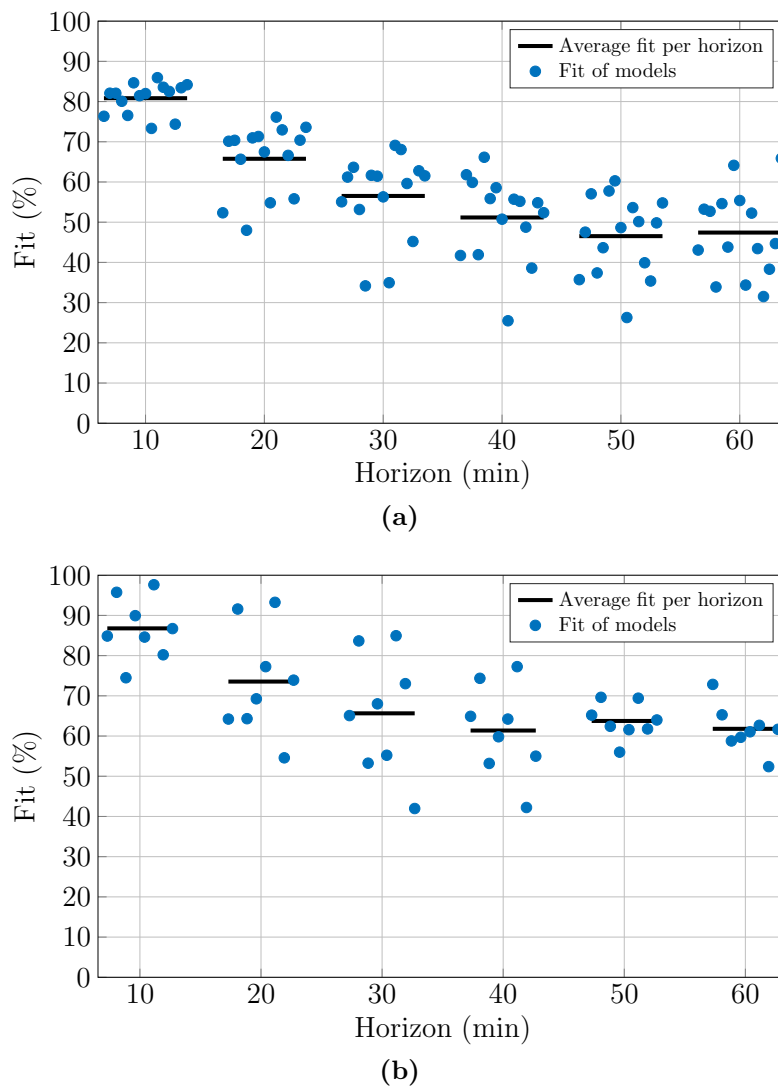


Figure 4.12: Training results. (a) Results for Patient A are good enough up to 20 minutes with an average fit of 65.8%; (b) Results for Patient B reach the 30 minutes of prediction with 65.6% of fit.

Table 4.10: Validation results for Patient A. The models marked in bold are the best ones chosen for each prediction horizon.

# Model/ Horizon (min)	M1	M2	M3	M4	M5	M6	M7	M8	M9	M10	M11	M12	M13	M14	M15
10	81.5 ±2.5	NA	74.2 ±9.0	78.2 ±6.7	86.2 ±2.9	86.7 ±2.3	77.1 ±6.1	72.5 ±12.6	77.9 ±6.7	85.5 ±6.3	80.5 ±5.7	80.2 ±5.8	84.2 ±4.3	73.4 ±9.4	78.1 ±9.8
20	44.4 ±39.8	58.4 ±15.2	65.5 ±12.3	71.3 ±8.0	58.7 ±11.2	59.1 ±15.8	58.4 ±13.1	56.9 ±17.1	53.3 ±0.7	64.2 ±15.7	69.4 ±18.5	60.9 ±12.3	54.3 ±21.8	60.5 ±14.4	64.7 ±17.0
30	66.9 ±0.4	44.7 ±22.2	52.6 ±7.9	45.4 ±19.8	23.4 ±12.8	44.5 ±20.1	43.5 ±20.2	43.1 ±21.5	73.5 ±0.0	67.0 ±16.0	30.2 ±21.1	54.9 ±14.2	64.6 ±2.6	45.3 ±25.8	45.6 ±30.6
40	24.4 ±2.5	52.8 ±15.8	35.0 ±18.6	42.6 ±10.1	78.2 ±0.0	48.6 ±11.1	39.7 ±7.9	39.0 ±0.0	NA	44.6 ±21.8	48.9 ±11.7	45.8 ±26.2	2.3 ±0.0	41.0 ±21.9	20.2 ±21.6
50	5.6 ±0.0	43.7 ±16.1	37.9 ±22.0	39.3 ±13.8	56.0 ±25.4	45.7 ±19.5	36.1 ±8.2	NA	20.4 ±0.0	40.4 ±8.7	46.9 ±22.4	50.0 ±6.3	NA	47.2 ±20.8	NA
60	22.4 ±11.1	31.1 ±1.6	30.7 ±19.2	NA	54.2 ±17.8	37.9 ±19.8	32.8 ±19.6	30.4 ±0.0	15.7 ±0.0	46.5 ±19.5	28.5 ±9.4	26.6 ±11.6	1.3 ±2.6	26.5 ±10.0	NA

Table 4.11: Validation results for Patient B. The models marked in bold are the best ones chosen for each prediction horizon.

# Model/ Horizon (min)	M1	M2	M3	M4	M5	M6	M7	M8
10	69.5 ±7.1	87.2 ±10.3	55.1 ±11.4	85.8 ±5.7	70.2 ±8.4	79.2 ±23.7	41.6 ±13.5	91.7 ±5.9
20	47.4 ±6.1	72.2 ±19.9	52.0 ±18.2	49.5 ±17.6	76.6 ±7.7	78.2 ±13.6	46.7 ±9.3	50.8 ±11.4
30	74.1 ±0.0	60.1 ±11.7	33.0 ±14.5	55.8 ±5.4	36.2 ±0.0	70.3 ±10.9	53.4 ±9.7	47.1 ±0.0
40	65.7 ±0.0	57.0 ±15.9	51.1 ±11.3	65.9 ±4.6	52.6 ±0.0	36.0 ±29.5	NA	28.9 ±14.8
50	45.8 ±0.0	32.3 ±49.8	NA	51.0 ±2.4	2.3 ±0.0	36.4 ±26.8	NA	NA
60	39.5 ±0.7	44.0 ±33.6	93.6 ±0.0	34.6 ±20.1	65.3 ±0.0	17.9 ±11.3	6.3 ±0.0	NA

4.1.6.2 Validation of models

In this section, it is computed an M cross-fold validation to find those models that predict a migraine with higher accuracy for each prediction horizon. In this stage, predictions have been improved by means of both repairing techniques: signal repair and Gaussian fitting (*Complete-signal repair Strategic Study*). It was seen in Section 4.1.3 how these techniques improve the fit, in average, in more than 10 points. We present an example of such improvement at the end of this section.

For each patient, a model M_i , with $i = 1, 2, \dots, M$, trained with the $i - th$ migraine, is selected and validated with the remaining $M - 1$ migraines in the training set. The average of the $M - 1$ validations is calculated after removing the overfitted validations (those with negative NRMSE), and the results are shown in Tables 4.10 and 4.11 for Patient A and B, respectively. These tables represent the average fit and also the standard deviation. Results with 0.0% of deviation

represent models only able to validate 1 of the $M - 1$ migraines in the training dataset. The higher the horizon, the lower the number of migraines correctly predicted. Notice that some models are not able to predict any migraine in the dataset (labeled as NA).

As expected from the training results, models are not able to reach more than 20 minutes with the defined level of accuracy (close to 70%). Thus, from this point, we discard prediction horizons greater than 30 minutes. Next we show some examples of expressions obtained from Patient A and B, after removing all autoregressive models, which do not depend on the inputs u_x , and only depend on the real output, y_r , and/or the predicted one, y_p . They are removed because they cannot be implemented in a real scenario.

Expressions for GE models for Patient A:

$$M_{A11}^{20} : y_p[k + 1] = y_r[k - 20] + e^{(\log(49 \cdot 10^{-5} + u_2[k-108]) - y_r[k-113] + \sin(u_2[k-114]))} \quad (4.4)$$

$$M_{A3}^{20} : y_p[k + 1] = y_r[k - 20] + \frac{e^{\frac{u_3[k-81]}{u_2[k-70]}}}{u_3[k - 20]} \quad (4.5)$$

$$M_{A4}^{20} : y_p[k + 1] = y_r[k - 20] + (65 \cdot 10^{-4} - u_3[k - 20]) * \Delta \left(u_2[k + \tau] \Big|_{-100}^{-20} \right) \quad (4.6)$$

Expressions for GE models for Patient B:

$$M_{B5}^{20} : y_p[k + 1] = \frac{y_r[k - 20] * u_4[k - 134]}{\frac{1}{3} * u_4[k - 134] + y_r[k - 46] + \Delta \left(u_4[k + \tau] \Big|_{-37}^{-20} \right) + \cos \left(\Delta \left(u_3[k + \tau] \Big|_{-140}^{-20} \right) \right)} \quad (4.7)$$

$$M_{B2}^{20} : y_p[k + 1] = y_r[k - 20] + y_r[k - 29] - y_r[k - 45] + \log(u_3[k - 53]) \quad (4.8)$$

$$M_{B6}^{20} : y_p[k + 1] = y_r[k - 20] * \log \left(\log \left(\max \left(u_2[k + \tau] \Big|_{-136}^{-20} \right) \right) \right) \quad (4.9)$$

To avoid overfitting, 3 models have been selected to compute an average prediction. The models marked in bold in Tables 4.10 and 4.11 have been chosen as the best ones for each prediction horizon. The criteria have been as follows:

(i) first, to sort the models according to the number of migraines validated with a fit higher than 70%, then (ii) to sort them by average fit and take the first three models. This may cause some undesired selections, such as in the case of the 30-minutes model for Patient B. In this case, it is desirable to select one model (model $M1$) validating at least one migraine with a high fit, instead of a model (like model $M4$) that barely reaches the 60% of fit.

Equations 4.4 through 4.6 show the expressions corresponding to the selected models in Table 4.10 for 20 minutes of prediction horizon for Patient A. For Patient B, the expressions for 20 minutes ahead are those in Equations 4.7 through 4.9. These models do not depend on the inputs u_x , and only depend on the real output, yr , and/or the predicted one, yp . These models cannot be implemented in a real scenario, and must be considered as purely mathematical solutions.

This work is part of a real clinical study, and one of the goals in the envisioning of GE models is the implementation of low power consuming predictive models to be executed in the sensing nodes. As stated above, this is not feasible using AR models. For Patient A and 20 minutes of prediction horizon, using models $M3$, $M4$ and $M11$, only EDA (u_2) and HR (u_3) inputs are needed to predict migraines. For Patient B, using models $M2$, $M5$ and $M6$, EDA, HR, and SpO2 (u_4) are needed to predict migraines with 20 minutes in advance. None of the selected models requires the FFT functions, which contributes to generate easily implementable and low power consuming models.

For the purpose of comparison, the average fits achieved with the selected GE models are shown with those of the ZOH Table 4.12. The ZOH model supposes that, within h minutes, the current output value will remain the same for h minutes more, with $h = 10, 20, \dots, 60$. The ZOH model presents higher fit values for Patient A because his/her migraines are, on average, longer than those from Patient B. Longer decay curves lead to a lower error in the ZOH models. In view of the results, the experimental models using GE for the prediction of migraine are, so far, limited to 20 minutes of prediction. Therefore, our test experiments will be limited to this prediction horizon.

Complete-signal repair Strategic Study (CSS)

In the following lines, the benefits of repairing the predicted signals are shown. Several repairing techniques can be applied to the output generated by the prediction models. These techniques are the already mentioned. We remember them briefly, in the following order: (i) removal of spurious events using both time and level thresholds, (ii) single event detection using a linear

Table 4.12: Comparison of the results using GE and the ZOH predictive model.

Horizon (min)	Patient A		Patient B	
	GE	ZOH	GE	ZOH
10	86.1	69.9	88.2	69.2
20	68.7	45.7	75.7	41.5
30	52.7	28.9	68.2	18.1

decider; and, (iii) recovering of the original desired shape of the migraine event by Gaussian fitting of the detected event.

Figure 4.13 represents this situation, where in addition to the real pain curve (dark solid line), a high level event has been detected at the beginning (dotted line). This event is not considered a symptomatic curve and is removed. The fit improves from $f = -365.0\%$ to the $f = 81.1\%$ achieved with the repaired prediction (clear solid line).

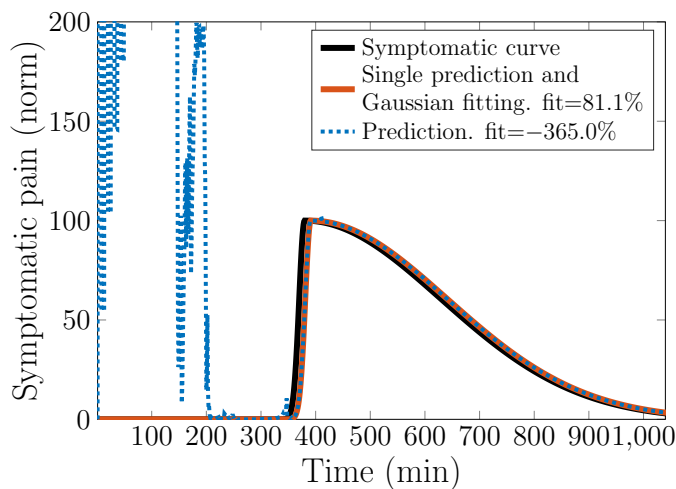


Figure 4.13: CCS. Example of the benefits of using improvement techniques. Here, over the original prediction, spurious and high values are removed and a single event detection is performed before applying a Gaussian fit.

As aforementioned, this improvement techniques are applied for each single prediction before the calculus of the average prediction.

4.1.6.3 Test results

In this Section the selected models are applied over the test dataset. The test dataset for Patient A is composed of 5 migraines and 5 asymptomatic periods of time (time series without migraine events). For Patient B, there are also 5

Table 4.13: CCS. Test results for Patients A and Patient B at 20 minutes of prediction horizon at 68.5% and 70% of fit respectively.

	Patient A	Patient B
TPR (%)	60.0	100
TNR (%)	80.0	100
PPV (%)	75.0	100
F (%)	66.7	100

migraines but 6 asymptomatic periods.

For each patient, the three models for 20 minutes are applied and then the average prediction is computed. A detection is considered as a migraine event when the fit is higher than 70% (except for Patient A, where this threshold has been downgraded to 68.5%, due to this is the average fit achieved by the selected models in training). The metric to evaluate the goodness of the models is the F-value, and the results are shown in Table 4.13.

Models for Patient B detect all the events in the test set, and no alarm has been produced in asymptomatic periods of time. For Patient A, one false positive has occurred due to an non-reparable prediction in one of the models, and precision (PPV) falls to 75%. The sensitivity (TPR) is low due to two misclassifications. Although the F-value is not as good as expected for Patient A, test results show that the migraine prediction using GE is feasible.

Despite the maximum prediction horizon to predict migraines using GE algorithms is 20 minutes, and despite this value is far from the almost 40 minutes achieved using state-space algorithms in Section 4.1.5, the achieved results are considered enough and sufficient to advance the drug intake and avoid the symptomatic crisis. GE algorithms are simple non-linear equations easily programmable in low power monitoring devices for real time prediction. In addition, the GE performs feature engineering and feature selection opposed to the exhaustive work that needs to be done in classic methods. Since the complete evolution of GE is known, the last population can act as a starting point in the new optimization process that happens when a sensor fails. Results obtained are excellent as shown in Figure 4.14.

It can be considered a real Mobile Cloud Computing scenario [42] where the GE models are performed in a cloud server. The selected prediction models run in the monitoring devices, where any of the sensors can fail. In that case, the prediction model of the failing device is not valid while the sensor is out. However, upon this situation, in the remote server, the GE algorithm starts the training from the obsolete model expression avoiding the use of the immeasurable

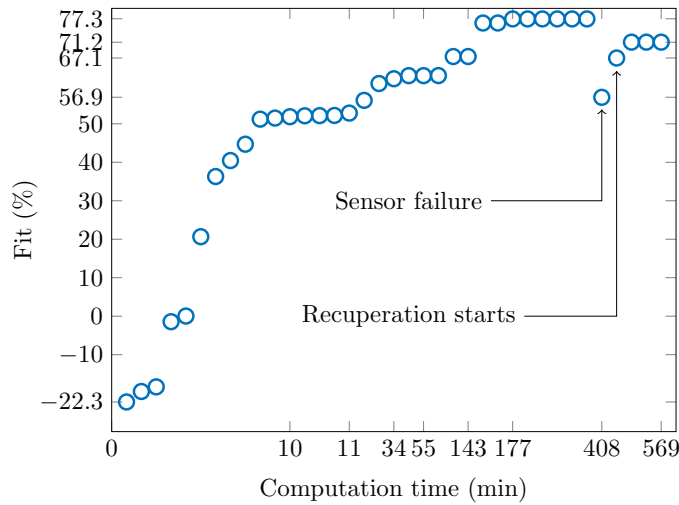


Figure 4.14: Simulation of the recovering capabilities of the GE system against sensor failures. After a sensor failure, the GE algorithm recovers a new model with an accuracy near to the one of the original and deprecated model.

biometric variable. In this case, GE quickly finds another expression with a similar accuracy to be used while the sensor is broken. Therefore, the approach here discussed, based on GE, is suitable for a large-scale monitoring scenario with real-life application.

The aforementioned scenario is simulated in Figure 4.14, where it is presented the evolution of the fit in a training experiment (notice that time axis is not linear). In the 80% of the trained generations (120000 generations), one of the inputs is set to 0, and suddenly the fit falls. From that point, the GE starts looking for new expressions and in 59 minutes achieves a new one with 86.8% of accuracy of the original (77.3%). 34 minutes later the fit is 71.2%, 92.1% of the original. This experiment shows the robustness of the GE algorithm for migraine prediction against sensor failures in a real scenario; the GE automatically finds a new model. This is another advantage over the classic models, where the selection of a new model is an exhaustive and manual off-line work.

The utility of GE for predicting migraines has been demonstrated. Better expressions would be found training the models with a higher population with higher computing units, and this work must be done in a future work. Also, as a future work, a penalty will be introduced in the GE training to avoid autoregressive expressions; in this way better models will be selected.

4.2 Advanced migraine prediction simulation system

4 Results

4.1 Fine-grained migraine predictive modeling

4.2 Advanced migraine prediction simulation system

4.3 Sub-systems II and III. Prediction support

4.4 Energy aware prediction system

In the following lines it is going to be presented the result of the simulation of the migraine predictive system using hemodynamic variables. It is worth to highlight the power of this simulation. The advanced migraine prediction simulation has the power of emulation of real time, which implies the feasibility and reliability of a predictive monitoring device—whose physical implementation is a next step of this research thesis.

The simulation of Sub-system I has been developed in incremental phases. The first step did not include the **Sensor Status Detector SSD**, which gives the robustness to the system, so it was not wither developed the **SDMS2** in this first step. This first version was tested with the GE models trained in Section 4.1.6. Later, it was simulated the complete system as shown in Section 3.3; this time using the state-space models trained in Section 4.1.2.

The evaluation of both implementations are shown in the next paragraphs. Both represent, for the first time in the literature, that prediction models are used to predict migraine events in real time. The current implementation of this simulation raises alarms directly from the average predictions. This may lead to spurious alarms. One possible solution to reduce spurious events could be the recursive Gaussian fitting of the prediction in real time. This should be done in small stretches, and its implementation might be tough. This challenge could be tackled in the future.

Previously developed GE prediction models have been used to validate the simulator's scheme. These models are used in real time where the off-line reparation of the signal does not take place.

Data from the two migraine sufferers have been used. The GE prediction models were trained to predict the incoming migraine events at 10 and 20 minutes in advance. The prediction models (hosted in the atomic models **Model_1**, **Model_2** and **Model_3** in the **Predictor** coupled model) are fed with the four hemodynamic variables: skin temperature, EDA, HR and SpO2.

Figures 4.15a and 4.15b (Equations 4.4 and 4.6) are two examples of real

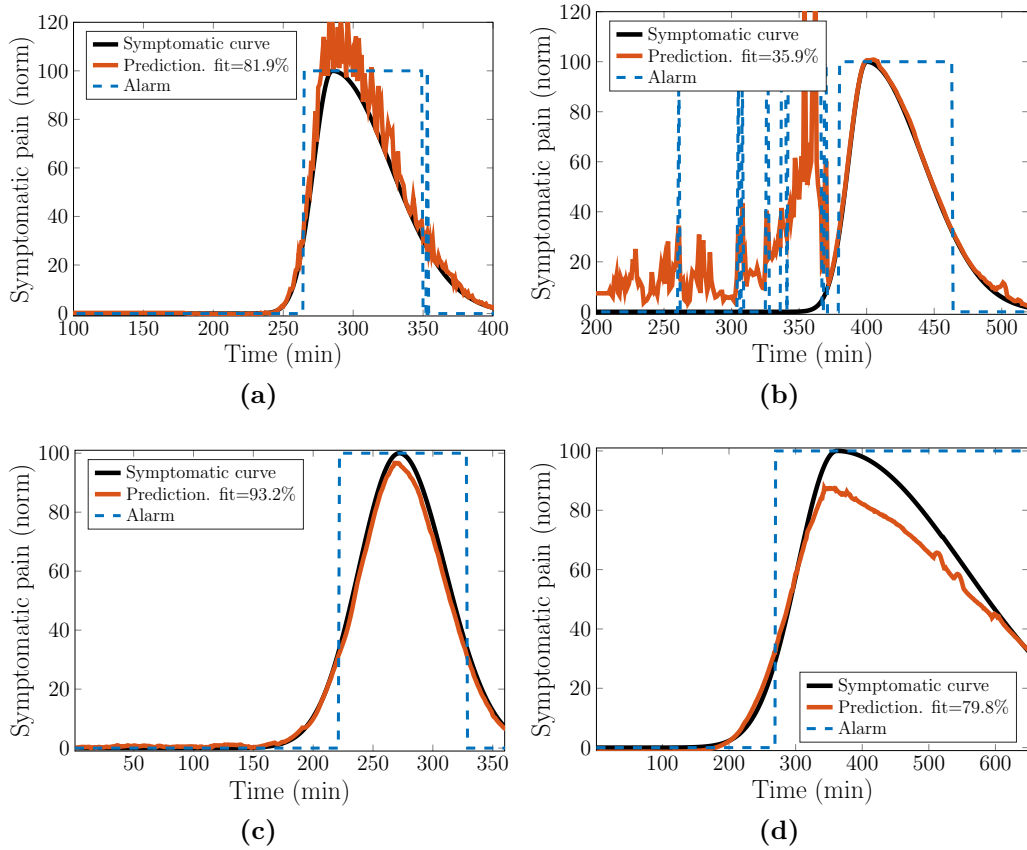
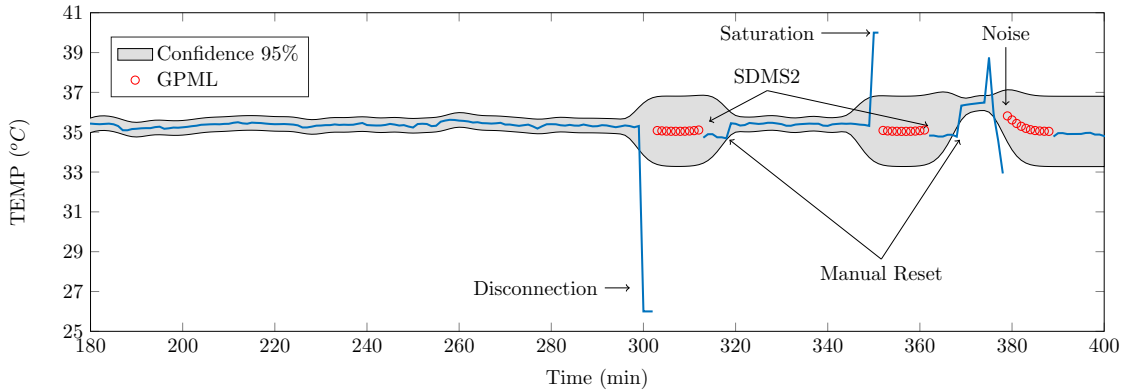


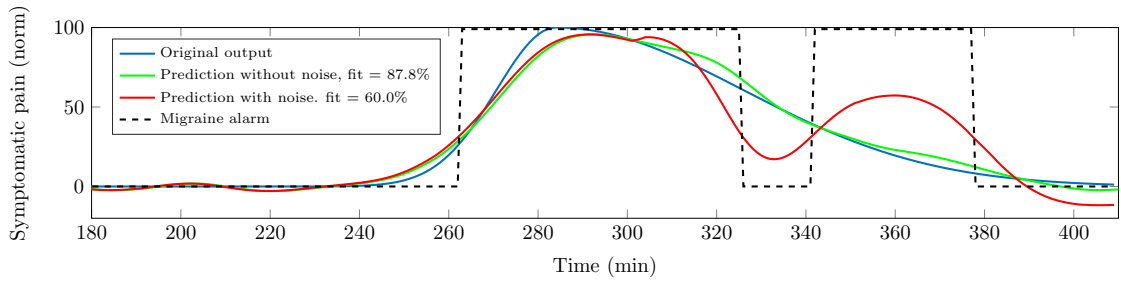
Figure 4.15: Results of the validation of the basic simulator using GE algorithms. (a) Patient A's migraine, 10 minutes forward; (b) Patient A's migraine, 20 minutes forward; (c) Patient B's migraine 10 minutes forward; (d) Patient B's migraine, 20 minutes forward.

time migraine predictions for Patient A at 10 and 20 minutes, respectively. Figures 4.15c and 4.15d (Equations 4.7 and 4.9) are the corresponding ones for Patient B. Using the simple threshold implemented in this version, 7 false positives occur in the examples of Figures 4.15a and 4.15b. This problem will be tackled in the future, to reduce the number of false positive alarms in Figures 4.15b and 4.15d using a more intelligent `Decider` model.

Now the behavior of the complete simulated system against sensor failures is described. Actual implemented migraine prediction models only support one sensor failure at a time. Migraine prediction using less than three sensors leads to low accurate predictions, which makes the system not useful. When all sensors fail, the system is not able to compute predictions and alarms will not be generated. To demonstrate the fundamental concept, errors have been induced in only one sensor (temperature sensor).



(a) TEMP sensor and system events.



(b) System's output and alarm event.

Figure 4.16: Error induction in the temperature sensor and behavior of the system against the failures.

In Figure 4.16a, three different types of errors have been randomly induced to the TEMP signal: disconnection, saturation and noise. In normal conditions, the `Predictors_TEMP-EDA-HR-SpO2` model (Figure 3.21c on 77) computes migraine predictions. When the TEMP SSD model detects an error, it activates the signal recovery through the GPML (red circles), and after a pre-specified duration (defined as 10 minutes in this example) the SDMS2 switches the migraine predictor model to `Predictors_EDA-HR-SpO2`, which does not use the information from the TEMP sensor. The gray bands represent the 95% confidence level of the GPML, which is tight (better) when data are not missing and it is spread (worse) when the GPML algorithm recovers data. Despite being represented all along the signal, the gray bands really exist only when the GPML model works (red circles). In a real HIL implementation, when the TEMP sensor has been replaced or repaired, the `Predictors_TEMP-EDA-HR-SpO2` model will be used again. In the example shown in Figure 4.16a, two manual resets were simulated.

In Figure 4.16b, it can be seen how these errors affect the prediction (red curve). Recovering values through the prediction simulation system avoids destabilization of the migraine system despite an apparent drop in accuracy. In

this example, the normalized root mean squared error (NRMSE) between the prediction with noise (red curve) and the prediction without noise (green curve) is only 12.2%.

With this simple example it has been demonstrated how intermittent sensor failures can be replaced by predicted values to stabilize the entire system and keep it within an acceptable operating range. Certainly, a deeper evaluation is required in real-life conditions for ambulatory monitoring (not having to stay in bed). This would also required advanced behavior in the **Decider** model. These experiments have been computed using state-space algorithms. There are not shown more examples, but it can be seen that real time predictions using state-space algorithms are, in general, smoother than those for GE, as can be seen comparing Figure 4.16b and Figures 4.15a through 4.15d.

4.3 Sub-systems II and III. Methodologies for migraine prediction support

4 Results

4.1 Fine-grained migraine predictive modeling

4.2 Advanced migraine prediction simulation system

4.3 Sub-systems II and III. Prediction support

4.3.1 Sub-system II. Prediction support using environmental data

4.3.2 Sub-system III. Prediction support using qualitative data

4.4 Energy aware prediction system

As it was mentioned, the research in this thesis focused mainly in the Sub-system I. Sub-systems II and III were defined in the methodology as entities for migraine prediction support. Several experiments were performed. They showed promising preliminary results that foresee a wide research area in migraine prediction. In this chapter these results are commented briefly. For further details of the experiments and detailed results, please, refer to the original theses in [141] and [179].

4.3.1 Sub-system II: Migraine predictive modeling from environmental data

For the purpose of creating coarse-grained predictive models to help the **Expert Decision System**, the pain was simplified to a binary sequence of 0/1 (yes/no). There were used slow global weather variables, and faster local environmental

variables (to create detection and predictive models). Due to the low number of migraine episodes with weather information, the results of these studies are not conclusive, however they present a hope on future experiment which will need of more data and and the specific study of patients whose migraines are attributed to weather conditions.

Both global weather variables and local environmental conditions depict similar results. Both generate individual models whose validations against other migraines show poor results. However, the model selection and the average predictive modeling improve the results. This is seen in as a moderate low number of false positives which indicates that environmental data actually do help to improve fine-grained predictions through coarse-grained predictive models.

Table 4.14: Accuracy of average models in validation phase for migraine detection and prediction at 20 minutes ahead using local environmental data for one migraineur.

	Accuracy in detection (%)		Accuracy in prediction at 20 min. (%)	
	Other migraines	Basal	Other migraines	Basal
Random Forest	45.4	77.8	35.3	77.8
IBK	44.0	34.7	33.8	34.7
MLP	25.6	15.7	43.7	15.7
ZeroR	34.6	-	33.7	-

For local environmental variables (see Table 4.14), in general, the behavior of the models has been similar in both the detection and the prediction phases. The *Random Forest* algorithm seems to be the most stable solution, leading to models with average accuracy rate of 45.4%, and individual results up up to 62%. Results are higher than the mode (ZeroR algorithm) and robust in the test phase. When the average model is applied over basal data (without pain) the accuracy is high enough (low rate of false positives).

As a conclusion to go for further, the study has drawn satisfactory results. The methodology has been developed from scratch and it can be used in later research as starting point to serve of help in decision making systems.

4.3.2 Sub-system III. Prediction support: clustering classification from qualitative variables

In the architecture described in this thesis, it was contemplated the usage of qualitative data for prediction support. Qualitative data are variables that surround the problem and might have, *a priori*, a different nature and origin. In

this research it was suggested the usage of premonitory symptoms as qualitative data to help the fine-grain migraine prediction. It is a matter of future work the study of the predictive effect of the premonitory symptoms to predict the beginning of the pain. However, preliminary proof of concepts have been performed to sketch out if premonitory symptoms are susceptible of indicating that a migraine it is going to occur or not.

It was carried out a study using a *Random Forest* algorithm boosted with *AdaBoost* [170], and performing 10-fold cross validation. Variables were selected using the *Information Gain* attribute evaluator with the *Ranker* search method. This study were carried out using 229 migraines from 34 patients, of whom 85.3% experienced at least one premonitory symptom during the lapse of the experiment, which lasted two months. The mean number of premonitory symptoms recorded was 4.3 ± 3.3 per event. The mean number of premonitory symptoms recorded prior to pain onset was 3.4 ± 2.0 , and 5.1 ± 4.9 for symptoms recorded after onset (non- predictive symptoms).

Table 4.15: Detailed accuracy for the classification of migraines with predictive premonitory symptoms, no predictive symptoms, and migraines without symptoms.

Class	TP (%)	FP (%)	TPR (%)	PPV (%)	F-value (%)	ROC
PS pred.	61.9	12.0	66.1	61.9	63.9	82.3
PS no pred.	75.0	15.8	77.4	75.0	76.2	86.8
No PS	100	4.4	90.9	100	95.2	99.1
Weighted average	79.0	11.3	78.4	79.0	78.6	89.3

Results in Table 4.15 show that premonitory symptoms are able to correctly classify 79.0% of events (classes are: predicted migraine, non-predicted migraine and migraine without symptoms). The resulting quality parameters are very promising: $F - value = 78.6\%$, $ROC = 89.3\%$, $TPR = 78.4\%$ and $PPV = 79.0\%$. The premonitory symptoms allowing the highest level of accuracy in event classification (as predictive or not predictive) were photophobia, fatigue, sadness, drowsiness, neck stiffness, yawning, phonophobia, and difficulty concentrating.

These results encourage the researcher to keep ahead with these studies to further complete the **Expert Decision System** fed with heterogeneous data to help the fine-grained migraine predictive system to achieve better results.

4.4 Energy aware prediction system

4 Results

4.1 Fine-grained migraine predictive modeling

4.2 Advanced migraine prediction simulation system

4.3 Sub-systems II and III. Prediction support

4.4 Energy aware prediction system

4.4.1 Compression techniques for energy saving

4.4.1.1 Discrete Wavelet Transform (DWT)

4.4.1.2 Discrete Fourier Transform (DFT)

4.4.1.3 Compressed Sensing

4.4.2 Energy savings

4.4.3 Energy aware predictive models

4.4.3.1 The concern of the energy consumption

4.4.4 Workload balancing in an MCC system

4.4.4.1 Energy results in WBSN

4.4.4.2 Workload off-loading policies

4.4.4.3 Economic benefits

This Section bases on the MCC scheme like the one presented in Figure 3.31 (page 95). As it was mentioned in Section 3.7.2, data transmission is the most power hungry task in a monitoring node. In this Section the results of the aforementioned lossy-compression techniques are shown: (i) Digital Wavelet Transform (DWT), (ii) Compressed Sensing (CS), and (iii) Fast Fourier Transform (FFT). The results presented here were obtained as a result of the Bachelor thesis of Álvaro Martín [113]. The algorithms were implemented and tested in a ATmega328 microcontroller and using a Bluetooth Low Energy device for data transmission.

In the following, the results for each one of the techniques are studied in detail. At the end of the Section, we select solutions of the optimization curves and the energy savings are computed. These solutions reach energy savings of up to 96.5%.

4.4.1 Compression techniques for energy saving in wireless transmissions

Figures 4.17a and 4.17b represent both ECG and PPG respectively. Remember the reader that the compression techniques applied in this study are only focused on ECG and PPG whose sampling frequency are 250 *Hz* and 75 *Hz* respectively—

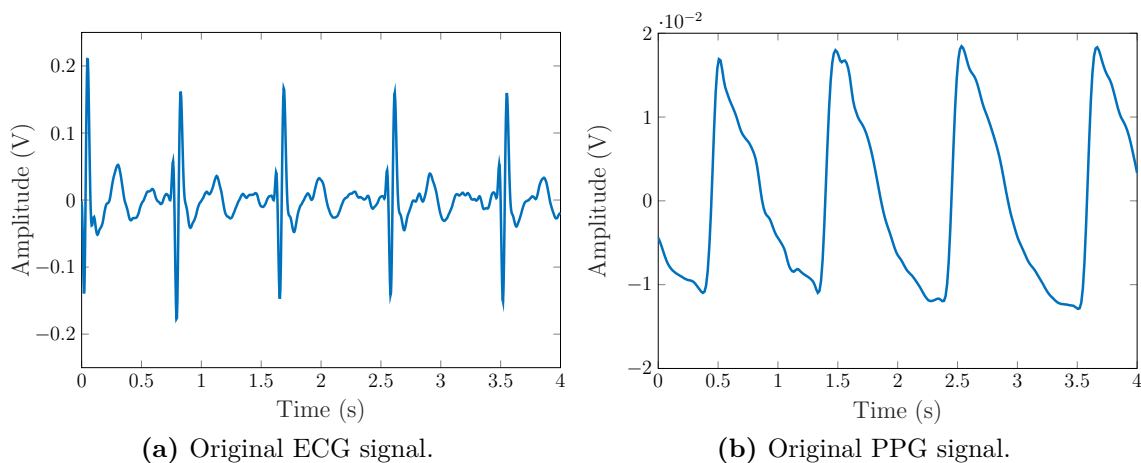


Figure 4.17: ECG and PPG signals before applying transformations.

faster than the 0.2 Hz of the skin temperature and EDA.

To compare the different solutions of the optimization process two metrics are used: the amount of data (relative number of samples (%)) and the fit between the original HR computed from raw ECG and PPG, and the reconstructed one \widehat{HR} obtained from the uncompressed data.

As it was shown in Figure 3.8 (in page 53), R peaks and the systolic points are those with higher energy. HR is the averaged time distance between peaks of the ECG or PPG signal (HR can be measured from the diastolic points as well). As it can be seen in Figures 3.8b and 3.8d only a range of the spectra contains relevant information for the calculation of the HR. Based on all this information, in the following paragraphs there are discussed the optimization results of the different transformations applied to reduce the amount of data to be wireless transmitted.

4.4.1.1 Discrete Wavelet Transform (DWT)

This section shows the results of the energy optimization using the DWT. Remember that, in our optimization process, once the DWT is applied, detail coefficients (D) are removed, and a threshold is set up over the approximation coefficients (A). All values below the threshold are removed. The threshold filters the data that have a lower energy level, minimizing though the number of A coefficients feasible to detect the R peaks. The threshold varies from -200 mV to 200 mV with in steps of 5 mV for the ECG, and from 0 mV to 15 mV with increments of 1 mV for the PPG. These values were selected in a heuristic way using the available datasets from patients.

Figure 4.18 compares the original DWT coefficients and the filtered DWT coefficients after the optimization loop presented in Figure 3.32b in page 96. It

can be seen that only positive values remain. Only those values those coefficients over the threshold and their timestamps are sent. The thresholds obtained are 95 and 6 for ECG and PPG respectively, as shown in Figure 4.19.

Figures 4.19a and 4.19b represent the decision curves resulting of the optimization loops. Setting one of the metrics of the optimization we can get the others. These figures are useful as decision charts. In a real time problem, it can be established different policies to automatically select the threshold for coefficients removal. As an example, with low battery levels it can be chosen higher thresholds—coarser fits—to save energy. A deeper study of the benefits of these decision techniques will be carried out as future work.

Remember the reader that it is not desired to recover the original ECG or PPG, but the HR, and the fit in Figures 4.19a and 4.19b refers to the latter. In Figure 4.19a it can be seen how the saving (in % of samples to be transmitted) keeps high till $fit = 87\%$ (rounded point), and then it drops drastically if more fit is needed. The same situation occurs in Figure 4.19b; the lower the fit, the higher the saving, and thus the higher the threshold (more coefficients are removed).

Once data are transmitted, in the receiver, the inverse DWT (IDWT) is performed over all non-zero coefficients (suppressed by the threshold). The signals representing the peaks are recovered, and then the HR is computed. The results in the receiver side are shown in Figures 4.18e and 4.18f, and correspond to the re-marked solutions in Figures 4.19a through 4.19b. Both solutions perform the optimization with an accuracy of 87%, achieving transmission savings of 93% and 81% for ECG and PPG signals respectively. It is noticeable that the compression is higher for ECG than for PPG (observe that the threshold in PPG leads to more coefficients to be transmitted). This is due to the variability of the PPG used is higher, and more severe thresholds lead to a dramatic reduction of the fit (see this in the few number solutions with positive fit in Figure 4.19b).

4.4.1.2 Discrete Fourier Transform (DFT)

This section shows the results of the energy optimization using the DFT. In this optimization process only some coefficient in the frequency domain are sent, the hypothesis behind this suggest that most of the HR information resides in a band of the ECG signal. Varying the width and position of a rectangular filter there are obtained a set of decisions points (solutions of the curve) that serve as decision making policy.

Figures 4.20a and 4.20b show the single-sided frequency spectrum of the ECG and PPG signals respectively. Most of the information of the ECG signal is found between 3 Hz and 25 Hz. The ECG signal is filtered to remove high frequencies

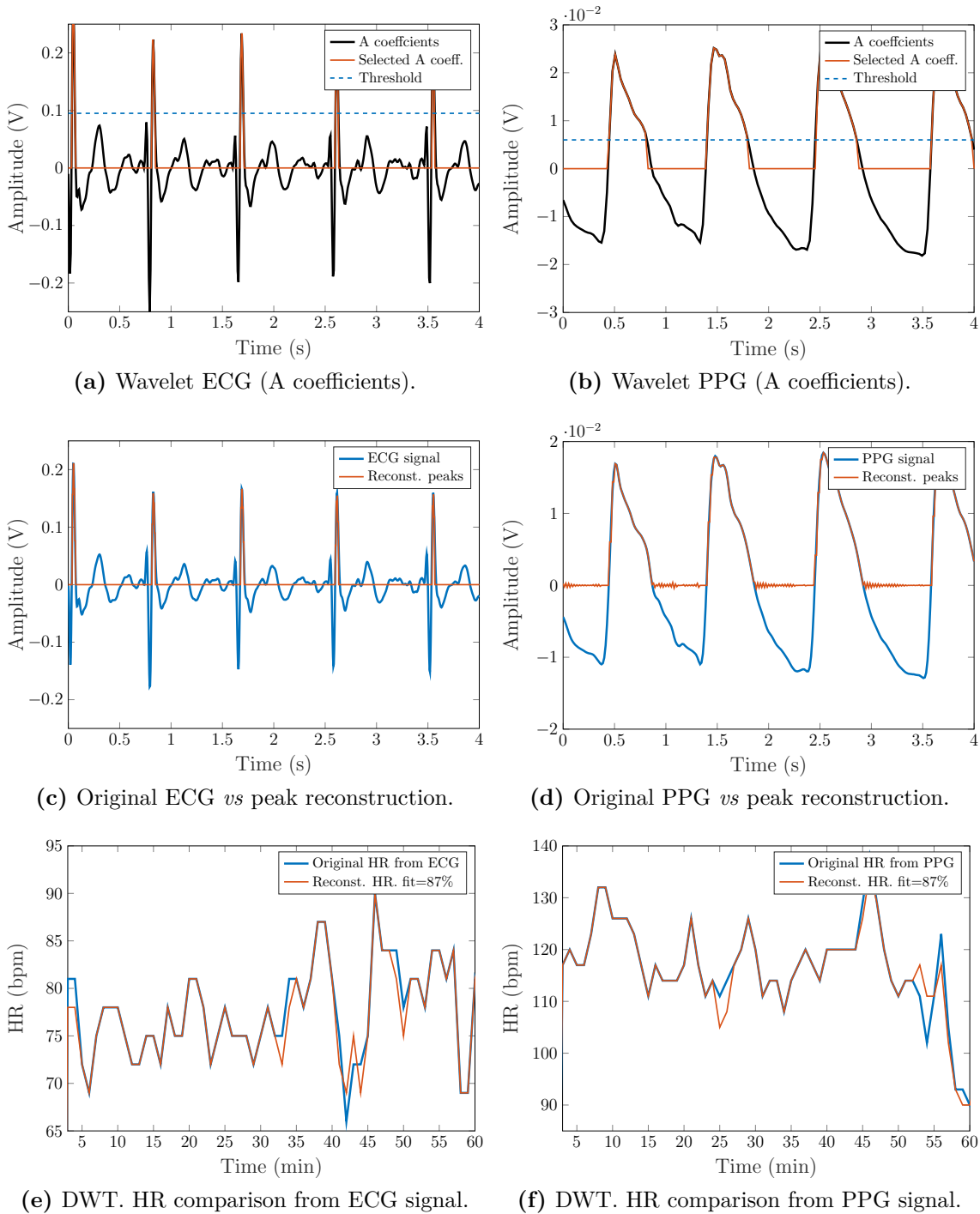


Figure 4.18: Results of the optimization process to achieve energy saving through data compression using the DWT. The results are depicted for the computation of HR obtained from ECG and PPG signals.

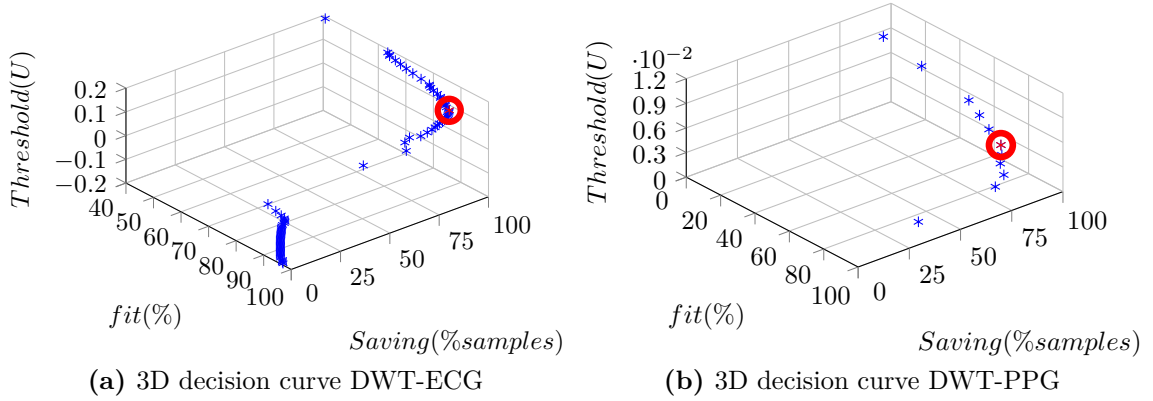


Figure 4.19: 3D decision curves for the compression of the DWT. Setting one of the metrics of the optimization we can get the others. Fit is computed over the HR.

coupled to the signal, such as the 50 Hz of the power grid. The PPG is an optical signal and does not have electromagnetic coupling; the information groups below 10 Hz , and the lateral lobes in its spectrum are caused by the signal clipping that acts as an ideal rectangular filter. Due to the reduced bandwidth of the PPG, the lower frequency k_l fixes to 1 Hz .

For each bandwidth and position of the ideal filter, in the receiver, the signal is recovered by the inverse DFT (IDFT), and then the HR is computed. The decision curves are drawn in Figures 4.21a and 4.21b. The highlighted solutions define bandwidths and savings of 18 Hz ($BW = 3 - 21 Hz$) and 71.2% for ECG, and 4 Hz ($BW = 0 - 4 Hz$) and 80.5% for PPG. Notice the reader that savings duplicate because only the positive side of the spectrum is sent. As a result, the fit achieved in the reconstruction of the HR for the selected solutions is 79.8% and 83.3% for ECG and PPG respectively. These solutions are depicted in `ecgreconsfftppgreconsfft`.

4.4.1.3 Compressed Sensing

This section shows the results of the energy optimization using the CS. The compression ratio (cr) is optimized in this loop. cr represents the savings in % of samples to be transmitted. Setting the cr we get the fit, as shown in Figure 4.23.

The ECG and the PPG are reconstructed in each one of the iterations of the optimization process. The HR is computed and the fit achieved is calculated, leading to the optimization curves in Figure 4.23. Compressed sensing over PPG achieves better results, and it is easy to reach higher energy savings in PPG than in ECG for a given cr . It can be noticed comparing the span of the curves in Figures 4.23a and 4.23b.

The highlighted solution in Figure 4.23a achieves for a $cr = 34\%$ a fit of

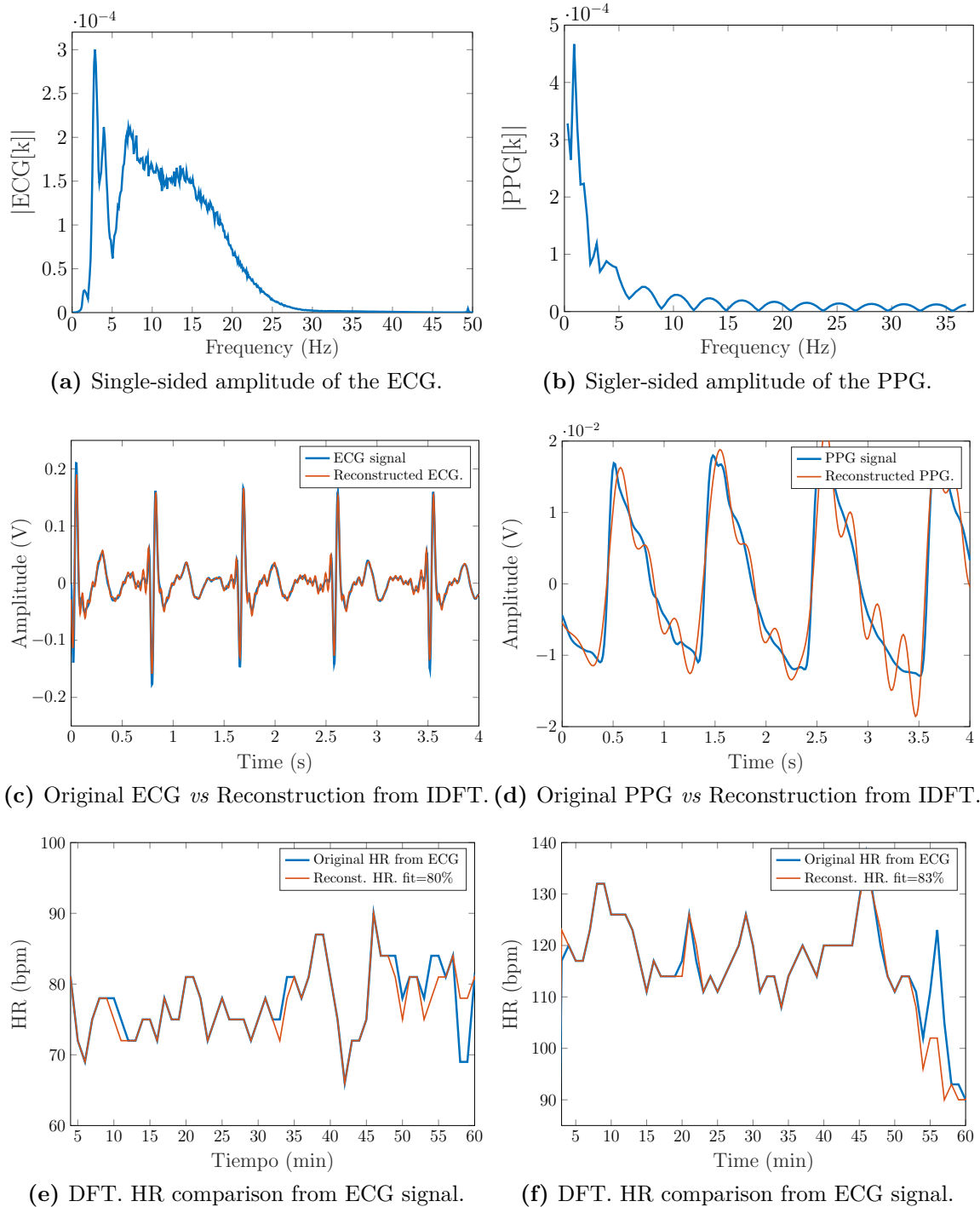


Figure 4.20: Results of the optimization process to achieve energy saving through data compression using the DFT. The results are depicted for the computation of HR obtained from ECG and PPG signals.

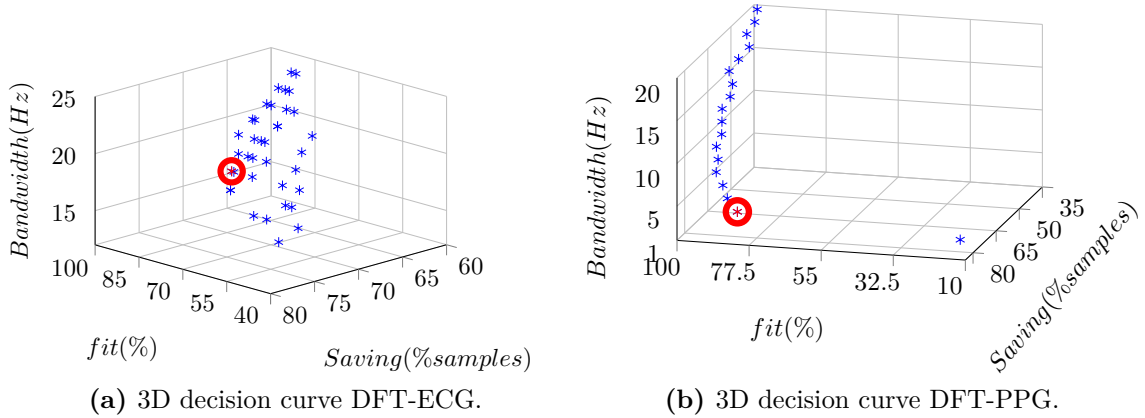


Figure 4.21: 3D decision curves for the compression of the DFT. Setting one of the metrics of the optimization we can get the others. Fit is computed over the HR.

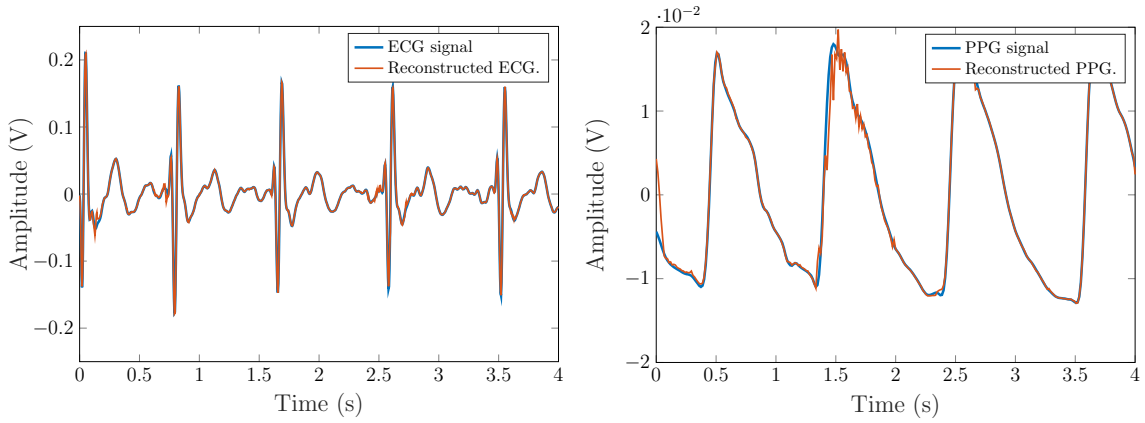
81%. For PPG, in Figure 4.23b, the selected solution as an example, achieves 82% accuracy for a compression ratio of 60%. These results are shown in the reconstructed signals in Figures 4.22a and 4.22b, and the resulting HR in Figures 4.22c and 4.22d.

4.4.2 Energy savings

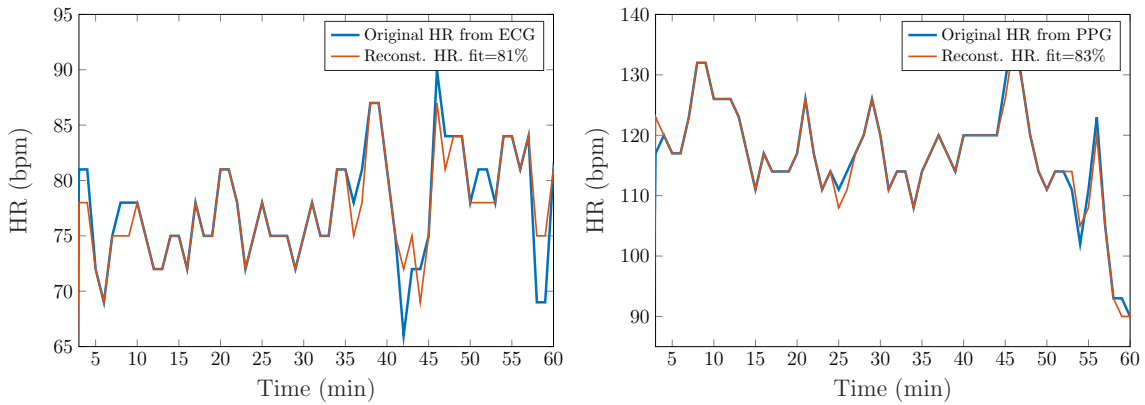
In this Section the results obtained after measuring the energy consumption of each of the optimization methods seen are presented. The initial operation of the application consists of data acquisition for one minute—migraine prediction techniques are carried out every one minute—and storing them in a buffer so that, once this time has elapsed, they can be processed and sent via Bluetooth.

This study compares the transformations and compression techniques with a *baseline* case. The solutions highlighted in the previous section have been implemented for comparison. The results of the relative energy savings and transmission time are drawn in the bar graphs in Figures 4.24a and 4.24b. For the specific details of each one of experiments, please, refer to the original work in [113].

In Figures 4.24a and 4.24b it can be seen that for the ECG signal the compression method based on DWT has offered better results than the rest—saving up to 93.2% and 96.5% in the time and consumption of data transmission. For the PPG instead, the compression methods that have given the best results have been the DFT and DWT, which are very similar. In this case, the solution chosen in Figure 4.21b for the DFT, achieves saving of up to 82.2% and 92.6% in time and consumption of data transmission.



(a) Original ECG *vs* Reconstruction from CS data. (b) Original PPG *vs* Reconstruction from CS data.



(c) CS. HR comparison from ECG signal.

(d) CS. HR comparison from PPG signal.

Figure 4.22: Results of the optimization process to achieve energy saving through data compressed sensing. The results are depicted for the computation of HR obtained from ECG and PPG signals.

The differences between the results obtained for ECG and PPG are due to the properties of these signals (bandwidth of the spectrum where the information of the HR is, the repetitiveness, the morphology...). Therefore, the results obtained with each of the compression methods used will depend on the properties of the signal used; thus, if in the study of migraines it is desired to apply these techniques on other signals (encephalogram, accelerometry...), the conclusions obtained will be different. In this study, the shipment has been made once the parameters of the algorithms have been calculated. These results are specific to the chosen values and will be deferred according to the parameters chosen: sacrificing consumption, or sacrificing quality.

The consumption and time for processing is negligible with respect to the

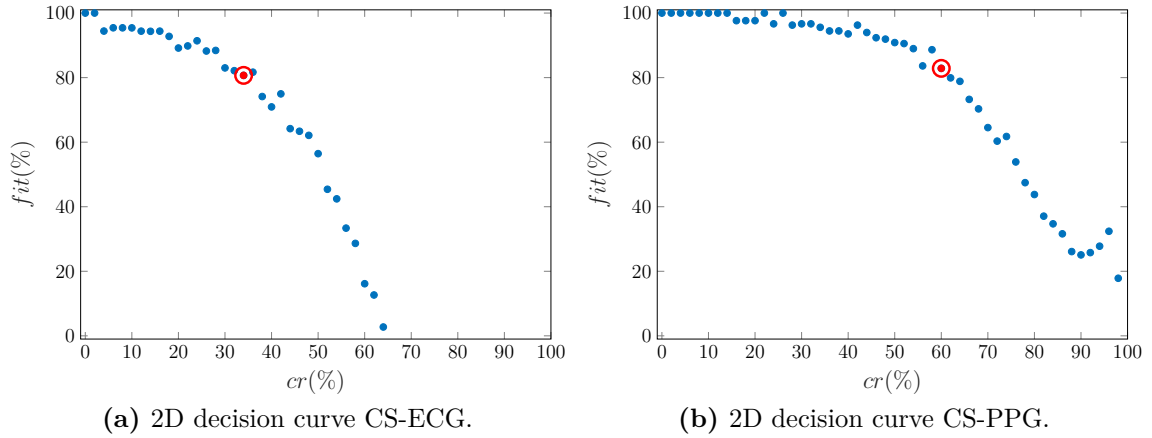


Figure 4.23: 2D decision curves for the CS. Setting the compression ration cr we can get the fit. Fit is computed over the HR.

time and consumption of the data transmission, since the processing is approximately 1% of the total consumption of each optimization algorithm, as shown in Figure 4.24.

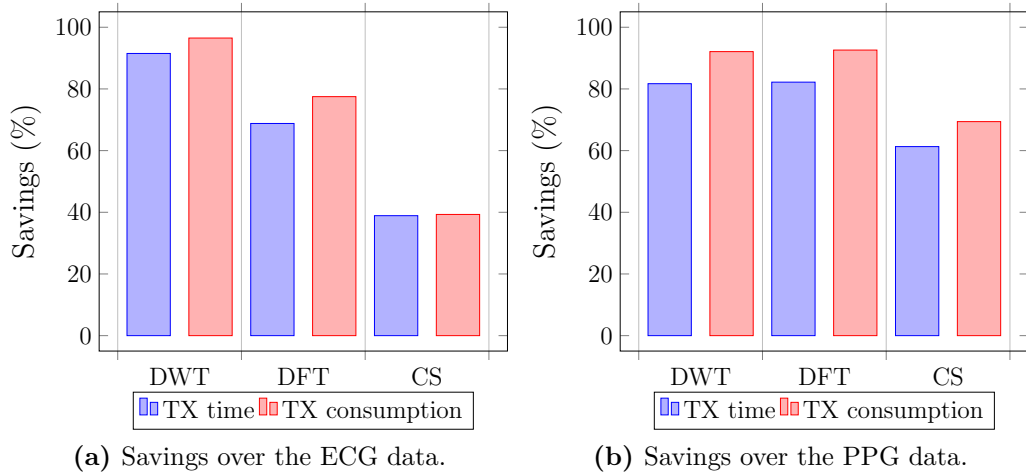


Figure 4.24: Comparison of the percentage of consumption that the sending and processing of the data implies in each of the algorithms. Results obtained for the specific solutions chosen, and compared with the baseline approach where data are acquired and sent via Bluetooth without prior processing.

Based on the results obtained, it could be feasible for a real implementation to eliminate the ECG sensor—getting the HR from the PPG signal. This will entail more comfort, a reduction of the data generated, a lower cost and greater consumption savings.

In this study it has been demonstrated that using any of the optimization

techniques, a reduction in transmission time and consumption has been achieved. With this reduction, a better experience will be obtained for the users in the study, since the charging time of the monitoring devices reduces and battery lasts more time allowing more time of continuous monitoring. This implies a reduction of data loss caused by cuts in the device due to battery depletion.

4.4.3 Energy aware predictive models

In this Section the energy optimization methodology proposed in Section 3.7.1 is evaluated. The results of the optimization problem are shown. As previously mentioned, the main goal of this study is to optimize the energy consumption of monitoring devices as an additional objective of the cost function. For this evaluation we use GE predictive models whose expressions will not be analyzed in detail in this Section. However, we focus on optimization curves and the solutions from the point of view of the optimized parameters: accuracy and energy consumption.

To create the models, 10 and 5 random migraines have been used. They belong to the already known patients A and B, respectively. All models have been trained for predictions of 10 and 20 minutes ahead. As a result of each training experiment a tri-dimensional Pareto front has been obtained. The Pareto front ideally contains 250 different solutions (the size of the population). In most of the cases, repeated solutions appear; thus, a Pareto front reduction is carried out to remove them.

The goodness of all the solutions in the Pareto front are, objectively, the same, because they are non-dominated solutions. We must choose subjectively one criterion to select the models to be embedded in the monitoring device. To maintain the maximum benefit for the patients, there have been selected those models with maximum fit. Figures 4.25a through 4.25c show the 3D Pareto front and one of its 2D projections for models trained with 10 minutes of prediction horizon for both of the patients. The solutions chosen are rounded by a red circle. All Pareto fronts obtained are globally convex: the solutions tend to minimize the error, the number of clock cycles and the energy due to the sampling of sensors. To improve the visualization and reading, the $\#clk$ is shown as thousands of cycles and the energy of sampling is shown in dBm .

Table 4.16 and Table 4.17 show all the models for Patient A and Patient B at 10 and 20 minutes of prediction horizon. In general, the average fit is higher at 10 minutes. Despite the average fit is still high at 20 minutes, the average number of clock cycles is higher. This leads to a longer computation time to get the prediction and thus a higher energy consumption. Regarding the sampling,

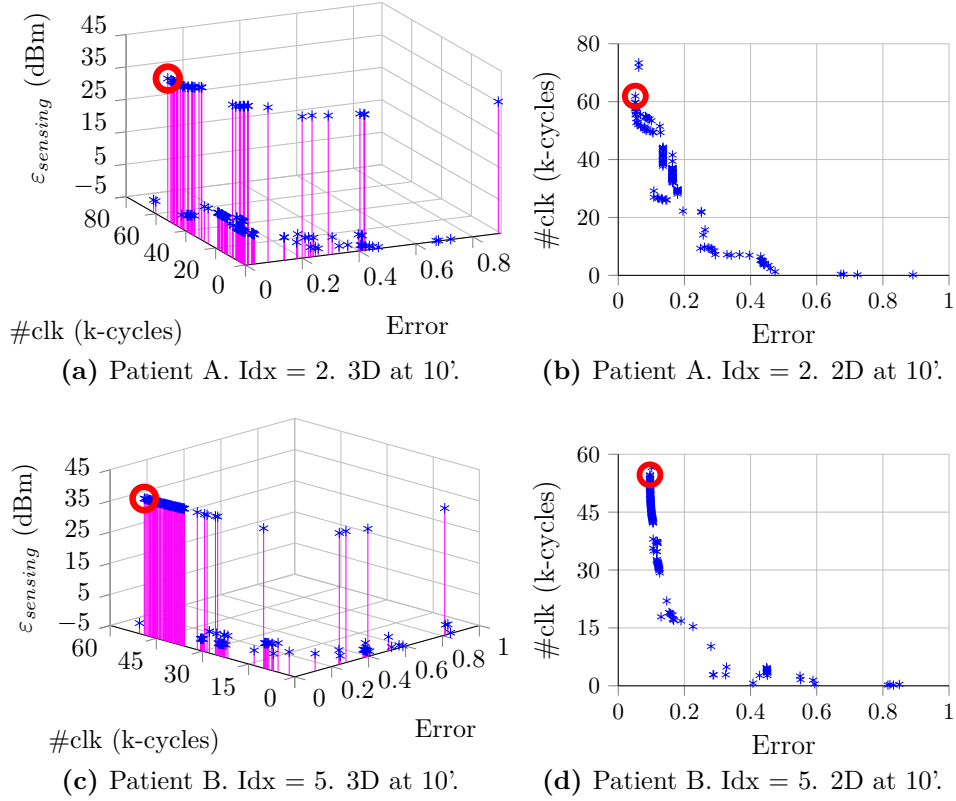


Figure 4.25: 3D and 2D views of the Pareto Fronts, result of the optimization process.

Table 4.16: Best fit solutions in the Pareto fronts for Patient A.

idx	fit (%)		#clk (k-cycles)		$\varepsilon_{sensing}(dBm)$		\bar{U}	
	10'	20'	10'	20'	10'	20'	10'	20'
1	79.2	77.4	74.3	24.8	35.6	-1.9	0.56	0.51
2	94.8	73.9	61.9	17.4	35.6	-1.9	0.52	0.00
3	91.6	92.4	11.6	106.1	35.6	-4.9	0.32	0.55
5	79.3	64.6	1.9	28.6	-4.9	35.6	0.64	0.52
6	75.2	76.7	21.8	21.7	-4.9	35.6	0.00	0.00
7	75.7	75.7	23.0	16.6	-1.9	35.6	0.34	0.41
10	77.1	76.3	21.7	75.7	35.6	-1.9	0.00	0.51
11	81.4	75.8	21.7	51.7	35.6	35.6	0.3	0.51
12	72.9	69.5	129.6	24.3	-1.9	35.6	0.36	0.53
15	76.2	84.2	38.2	86.3	-4.9	35.6	0.61	0.44

Table 4.17: Best fit solutions in the Pareto fronts for Patient B.

idx	fit (%)		#clk (k-cycles)		$\varepsilon_{sensing}(dBm)$		\bar{U}	
	10'	20'	10'	20'	10'	20'	10'	20'
2	81.6	83.6	16.6	52.1	-1.9	35.6	0.00	0.22
3	82.0	77.5	4898.7	4697.3	26	36.1	0.49	0.48
4	80.3	81.1	31.6	26.4	35.6	-4.9	0.61	0.62
5	90.4	81.4	54.7	36.2	35.6	-1.9	0.57	0.73
7	89.4	81.4	4773.1	48.1	26	-4.9	0.63	0.54

the presence or not of the SpO2 sensor—consuming $35.6dBm$ —makes the energy results vary quite much. For Patient A, it seems that the SpO2 sensor is very important for the prediction of her migraine. For Patient B it seems that the HR is an important variable, which makes increase rapidly the number of clock cycles.

From these best-fit solutions, it is computed the average symmetric uncertainty \bar{U} . With this, following the ideas that Ghasemzadeh *et al.* pose in [61], we can split the GE expressions into a combination of features, if possible, and thus estimate whether the features that compose the GE expressions are representative, or not. The symmetric uncertainty $U(X, Y)$ is a normalization of the mutual information $I(X, Y)$ between two discrete random variables X and Y , defined as in Eq. 4.10. U is bounded such that $U \in [0, 1]$, and the lower the U , the better. The idea can be introduced for efficient feature selection and can be adapted to any problem that the reader desires. For the sake of simplicity, the average symmetric uncertainty is computed between all pairs of features—as a worst case of overlapping of the individual entropy—instead of the complex computation of the multivariate symmetric uncertainty.

$$U(X, Y) = \frac{2I(X, Y)}{H(X) + H(Y)} \quad (4.10)$$

In Eq. 4.10 $H(X)$ and $H(Y)$ represent the entropy of the random variables X and Y , and $I(X, Y)$ is the mutual information defined as:

$$I(X, Y) = H(X) - H(X|Y) \quad (4.11)$$

It can be seen in Table 4.16 and Table 4.17 how \bar{U} has medium values, that in most of the cases is lower than 0.5. With this, it can be said that the features that compose the GE expressions are relevant enough and they do not overlap each other. This means that the GE expressions do not compute unnecessary features and the energy consumption of the microcontroller is thus optimized.

4.4.3.1 The concern of the energy consumption

Peculiarities of a GE predictive modeling box in the optimization framework is that the selected models are a result of the casuistic of the metaheuristic process that it is not deterministic. Nevertheless, with any other modeling solution the reader can also follow the underneath study.

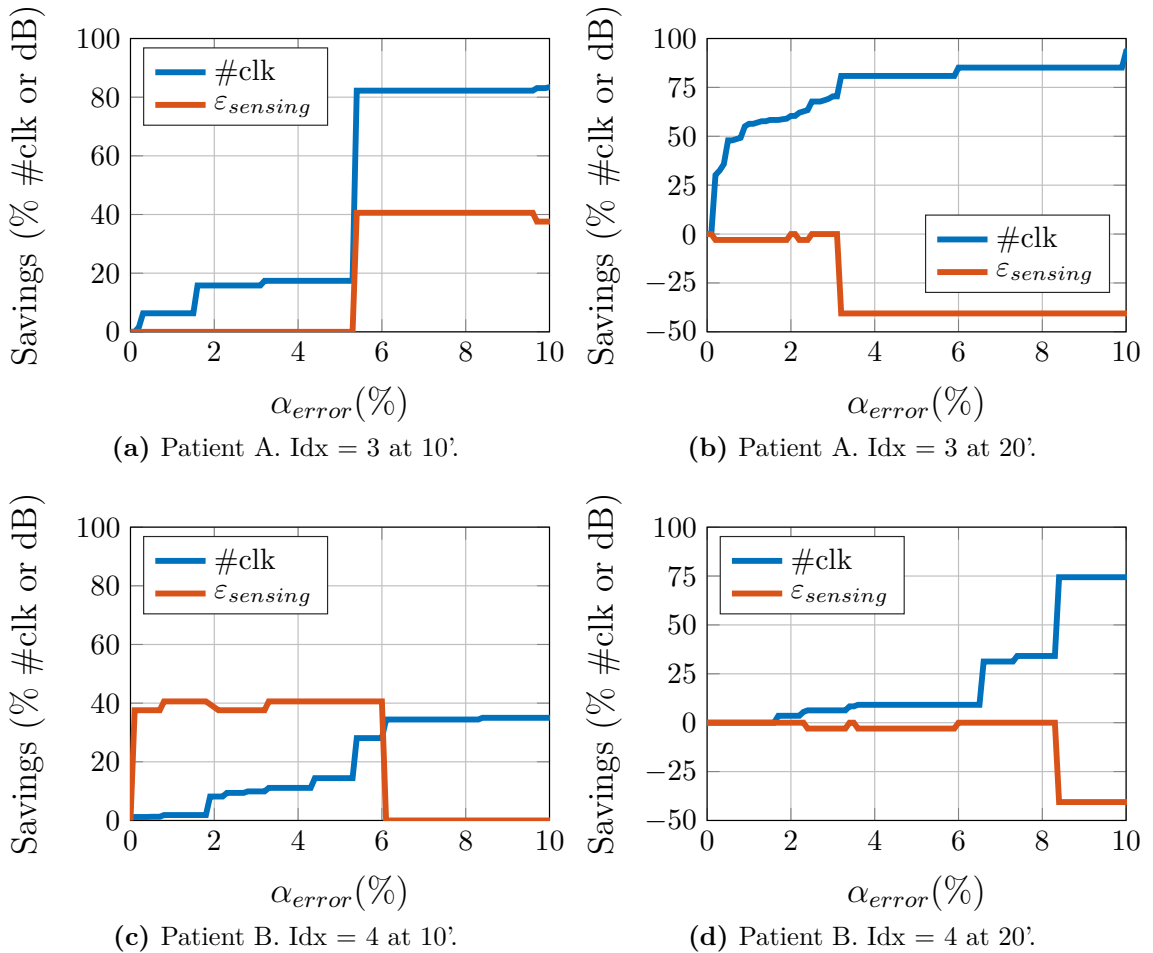


Figure 4.26: Savings in energy consumption from the perspective of number of clock cycles in the model execution and the energy consumed by the sensors in the sensing process.

Depicted Figures 4.26a through 4.26d show examples of the solutions of a Pareto front that satisfy a maximum error difference of α_{error} with the best-fit solution. These figures represent the savings of clock cycles %—proportional to the energy consumption of the microcontroller—and the energy savings (in dB) due to the selection of another set of sensors. The abscissa represents α_{error} , or the degree of error when we select a different solution of the Pareto front.

In Figure 4.26a we can see solutions for migraine $idx = 3$ of Patient A at 10 minutes. It can be seen that clock cycles can be saved if more error is tolerated, and in addition, the selected sensors consume as much as the ones selected in the best-fit solution. At a certain point, the savings in clock cycles get over the 90% with the good news that the set of selected sensors does not require the SpO2 sensor anymore, and the savings due to sensing reaches 40 dB. The opposite occurs in Figure 4.26d—4th migraine from Patient B at 20 minutes. Any of the remaining solutions in the Pareto front saves energy from sampling; furthermore, when the saving in the number of clock cycles gets almost the 80%, to maintain that needs from the SpO2 sensor and the consumption due to sampling increases (negative savings in dB). The sacrifice of error tolerance at 20 minute does not have such reward but a penalty of higher consumption. With this, now it can be applied the methodology for model selection presented in Section 3.2.2.2.

4.4.4 Workload balancing in a Mobile Cloud Computing System

In this Section it is going to be presented the results of the strategies applied at the node, coordinator and Data Center level. In addition, a study of the economic impact of the energy savings is performed.

Results have been extrapolated from real data acquired in the clinic and in the laboratory. The data used correspond to the migraines of the patients A and B again. Power and performance characterization for the coordinators and Data Center corresponds to the description in Section 3.7.3 (page 97).

4.4.4.1 Energy results in WBSN

The energy consumption of the microcontroller for node S1 in both working modes has been obtained from [142]. The consumption in node S2 has been measured with a HAMEG HM8012 digital multimeter. Radio consumption has been calculated in simulation as described in Section 3.7.3 for both of the nodes.

Simulation results have been calculated as the average of the results of 10 simulations of 60 seconds each. For the CC2420 radio chip, the default values provided by the simulator have been chosen.

Table 4.18 shows the energy consumption of the Shimmer nodes—microcontroller and radio for node S1, and microcontroller, 8000R oximetry sensor and OEM-III module and radio for node S2—for both working modes: streaming and processing. Energy has been calculated for an execution time of 1 minute and only for 1 patient.

Table 4.18: Energy consumption for nodes without intelligence in transmission and after applying reactive transmission policies.

	Computing		Streaming mode		Total streaming (mJ)		Processing mode		Total processing (mJ)	
	uC (mJ)	ϵ_{total} (mJ)	ϵ_{TX} (mJ)	Radio	RSSI (dB)	ϵ_{total} (mJ)	ϵ_{TX} (mJ)	Radio	RSSI (dB)	
Node S1	396	13526	61	-59	13922	430	9.0	0.3	-57	439
		13371	38	-68	13767		-	-	-	-
Node S2	3635	3600	6	-63	7235	3635	1.8	0.2	-70	432
		2388	4	-75	6023		10.5	0.3	-63	3646
							2.2	0.2	-75	3637

*uC consumption includes the consumption of NONIN 8000R SpO2 sensor and the consumption of OEM-III module (predominant).

When working in streaming mode, the nodes collect data and transmit them immediately without saving energy. These situations are shown in Table 4.18 for 0 dBm of radio power transmission in the streaming mode.

Radio switches on and switches off in every transmission. This explains the high consumption of node S1 in streaming mode. In processing mode, node S1 switches on just one time per minute, while in streaming mode it does for 217 times (see Table 3.9). A similar situation occurs with node S2; where in streaming mode radio switches on 60 times more than in processing mode.

Energy savings when computing HR in node S1, applying reactive techniques and transmitting once per minute, reach 96.9% (from 13767 mJ to 432 mJ). Energy savings in node S2 after the extraction of the SpO2 data, applying reactive techniques and transmitting once per minute, reach 50.3% in node S2 (from 7235 mJ to 3637 mJ). Notice that reactive policies in the radio lead to 1.1% energy savings in streaming mode for node S1 (from 13526 mJ to 13371 mJ); almost negligible. This is because most energy (99.5%) is used during the initialization process, and barely goes to the energy transmission ϵ_{TX} . These policies reach 80% of energy saving (from 9 mJ to 1.8 mJ) in streaming mode for the same node. For node S2, energy savings in the radio link lead to 33.7% and 79.0% for streaming and processing modes respectively.

Even though, the energy reduction in the radio link is significant. The absolute savings are masked due to the high power consumption of node computation. To increase the impact of radio energy savings, the computation of nodes S1 and S2 should be reduced. In this sense, more efficient HR calculation techniques—or sampling rate reduction techniques—would reduce the power consumption. For node S2 the strategy would be changing the sensor node and its proprietary processing module OEM-III.

According to the results, on-node data processing is more convenient than streaming, reaching a power consumption of 432 mJ for node S1 and 3637 mJ for node S2. Thus, for the remaining of the paper, we use this setup, and apply our workload balancing policies in scenarios SC4 and SC5 in Table 3.11, to further reduce the energy consumption.

4.4.4.2 Workload off-loading policies

In this section we present the results for the workload off-loading policies, corresponding to scenarios SC4 and SC5. Moreover, in this section we show the benefits of combining workload off-loading techniques (*i.e.* moving workload from the Data Center to the coordinators) together with energy optimization strategies in the HPC and Cloud Data Centers.

The energy optimization strategies are called: (i) server turn-off policies,

(ii) workload consolidation, and (iii) cooling power reduction. This yields the following scenarios:

- SC4 (baseline): the original scenario considers no workload off-loading. The coordinator nodes simply forward computation to the Data Center. The off-line phase is performed in the HPC cluster, and the online phase in the virtualized cluster.
- SC4 (optimized): this scenario applies Data Center energy minimization techniques, but without any workload off-loading policy.
- SC5, 100% prediction: coordinator nodes perform data preprocessing (*i.e.* GPML), but the virtualized Cloud takes care of 100% of the prediction. The off-line phase is performed in the HPC cluster.
- SC5, 30% prediction: in this case, coordinator nodes execute both GPML and 70% of the predictions (*i.e.* when battery power allows it). The remaining 30% are computed in the virtualized cluster.

All scenarios except for the first SC4 (baseline) apply the three energy minimization strategies aforementioned. The SC4 (baseline) is used to calculate the energy savings of the three other experiments.

Server turn-off policies are applied in both the HPC and Cloud clusters when the Data Center is not fully utilized. In this sense, workload is packed into the minimum number of servers possible and the remaining servers are turned-off. In order to guarantee the absorbency of the sudden workload peaks, 20% of unused servers are always left on. Workload consolidation is applied only in the virtualized Cloud cluster, and consists on packing together as many instances of either GPML or prediction as possible, until the server reaches a per-core utilization of 100%. As shown in Appendix C and explained in Section 3.7.3, up to 230 and 250 instances of GPML and prediction can be packed together, respectively. These two techniques are particularly useful during the initial deployment of the application. As both the HPC and the Cloud cluster have been sized according to the maximum workload, during the first weeks of the migraine prediction deployment, the clusters are heavily underutilized.

Figure 4.27 shows the evolution of the HPC cluster utilization for a period of 10 weeks as new models that need to be trained arrive to the system, due to the gradual inclusion of patients (blue dots). The utilization of the HPC cluster raises to absorb the peaks. Observe how model re-training increase (red line) as more migraines are being predicted. Finally, see the reader how utilization stabilizes around 80%, the design point of the HPC cluster.

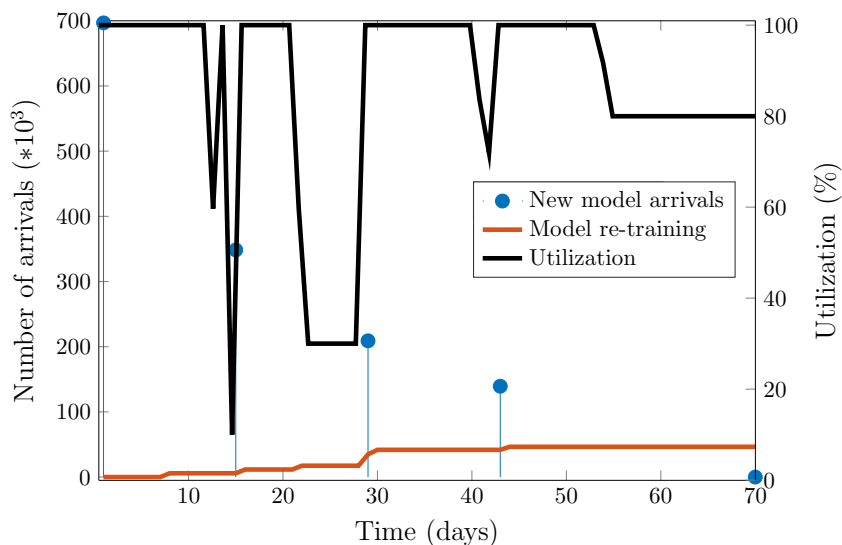


Figure 4.27: Utilization of the HPC Data Center. Blue dots correspond to gradual inclusion of patients (50%, 25%, 15% and 10%). Red line corresponds to models that need to be re-trained.

On the other hand, Figure 4.28 shows how the utilization of the virtualized cluster decreases as we off-load computation to the coordinators. This, by itself, does not decrease energy. However, when combined with consolidation and server turn-off policies, drastically minimizes power consumption.

Table 4.19: Energy consumption breakdown in HPC and Cloud Data Center for 10 weeks and various policies (MWh).

	HPC IT	HPC IT+Cool.	Cloud IT	Cloud IT+Cool.	Total
SC4 (base.)	643.5	1061.2	282.4	466.1	1527.3 (-)
SC4 (opt.)	584.7	907.5	219.1	340.0	1247.5(18.3%)
SC5 (100%)	584.7	907.5	159.1	247.0	1154.5 (24.4%)
SC5 (30%)	584.7	907.5	78.3	121.5	1029 (32.6%)

This behavior can be observed in Table 4.19, which shows the power consumption breakdown for the HPC and Cloud Data Centers for the whole period of 10 weeks. Cooling power reduction is achieved by increasing the room temperature of both clusters. To do so, experiments are run at increased inlet temperature for both the SandyBridge and the Intel Xeon servers, and heuristically observed that room temperature can be raised up to 26 °C without thermal redlining. Assuming an initial PUE equal to the world average PUE of 1.65, and a reduction of 4% [159] in cooling power for every degree of increase in ambient temperature, it has been computed the savings in cooling power. As can be seen, the proposed techniques achieve significant energy reductions, of up to

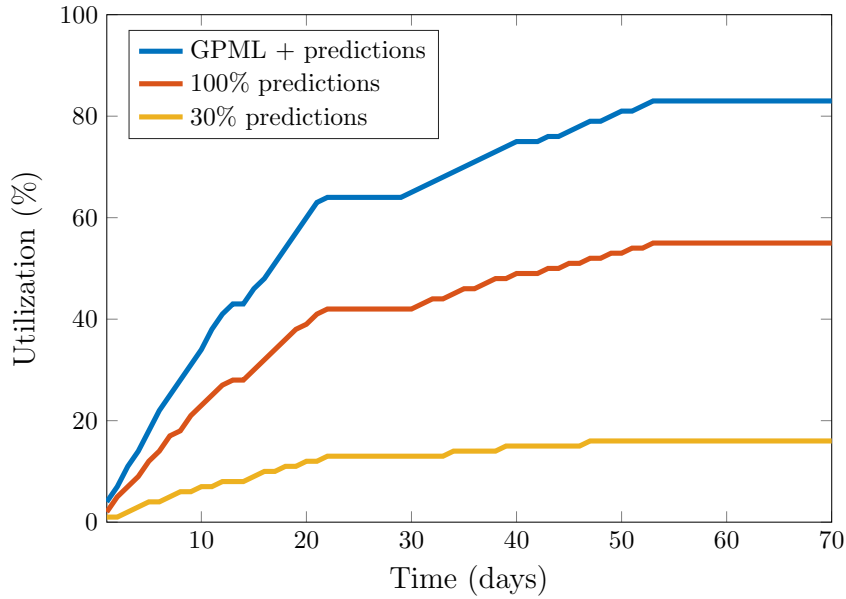


Figure 4.28: Utilization of the Cloud cluster.

32.6% in overall Data Center power when compared with the non-optimized scenario SC4 (baseline).

Table 4.20: Energy consumption for various workload off-loading scenarios for 1 week in the stationary state (MWh).

	Coord.	HPC DC	Cloud DC	Total	Savings
SC4 (baseline)	170.05	104.9	49.9	324.8	-
SC4 (optimized)	170.05	85.4	42.4	297.8	8.3%
SC5 (100% pred.)	170.12	85.4	32.2	287.7	11.4%
SC5 (30% pred.)	170.32	85.4	14.7	270.4	16.7%

Table 4.20 shows the energy consumption for the overall application framework of migraine monitoring and prediction for all scenarios. Results are aggregated per device, *i.e.* each column shows the aggregated energy value for all 1,393,649 patients under test. As can be seen, in a scenario with so large number of sensors and coordinators, the impact of these devices on the overall energy consumption of the application is large. In this sense, reducing the energy consumed by the sensor devices implies significant savings. Data Centers, however, are still the most important contributors to power consumption, and off-loading data to coordinators always yields important benefits, reducing overall energy consumption in the steady-state by up to 16.7%.

Finally, Table 4.20 also summarizes the maximum energy savings achieved for sensor nodes and the offloading scenario between the coordinator and the Data

Center. This results are used further below to compute the economic benefits of the proposed approach.

Table 4.21: Total energy savings in nodes and Data Centers and expenses in coordinator nodes for 1 minute simulation.

Saving in nodes S1* (kWh)	Saving in nodes S2* (kWh)	Daily saving in Data Centers (kWh)	Daily expenses in coordinators (kWh)
$3.7 * 10^{-6}$	$1.0 * 10^{-6}$	7819.3	51.2

4.4.4.3 Economic benefits

To compute the economic benefits expected by the proposed approach, it has been computed in Appendix D the total amount of economic savings that can be achieved by migraine prediction in Europe. These savings are of € 1271.8 million with an efficiency in prediction of 76%. Given the amount of energy consumed by the nodes, coordinators, and Data Centers, together with the price of household and industrial electricity, we can provide some insight on the economic impact of our approach. The goal of this result is simply to provide a general overview of the dimension of this problem in particular, and of Mobile Cloud Computing applications in general.

Table 4.21 shows the energy savings for each element of the monitorization system. After applying processing techniques and low-power radio policies in node S1 for HR measurement, we achieve a daily saving of $5.32 * 10^{-3}$ kWh; $1.44 * 10^{-3}$ kWh doing the appropriate in node S2. These values are shown in Table 4.21 for 1 minute of simulation of the radio channel in Castalia. Bigger than these values are the energy savings in Data Centers. Energy saving in these facilities brings an issue: part of the execution is done in coordinators leading to daily expenses of 51.2 kWh. As shown in Table 4.22, it worth to balance the workload. On average, economic savings in a year of execution in Data Centers lead to € 287.7 ± 104.1 million. This average calculation understands the HPC and Cloud clusters as a federation of Data Centers located in Europe (see Appendix C), and only the variable part of the industrial and household energy prices is taken into account. Economic savings in nodes in Europe lead to € 692,400. This calculation is the sum of all the savings in Europe taking into account the household energy prices in each country. Finally, the total amount of saving is € 288.4 ± 104.1 million.

Table 4.22: Average saving for a federation of Data Centers in Europe. Savings calculated for the variable electricity prices. Savings in nodes and Data Centers have repercussion in consumption and electricity costs in coordinator nodes.

Country	Target population	Household [50] energy price (€/kWh)	Industrial [50] energy price (€/kWh)	Saving in nodes (k€)	Expenses in coordinators (k€)	Saving in Data Center (k€)	Total energy saving (M€)
Turkey	305,138	0.122	0.070	93.0	0.50	200.4	478.9
Germany	184,150	0.295	0.149	135.2	0.73	426.1	594.3
UK	183,922	0.218	0.152	100.1	0.54	433.8	601.8
France	147,472	0.243	0.160	89.3	0.48	455.8	590.5
Italy	182,348	0.168	0.095	76.1	0.41	270.8	437.3
Spain	117,211	0.237	0.113	69.2	0.37	323.4	430.4
Netherlands	78,088	0.183	0.084	35.7	0.19	238.3	309.6
Netherland [122]	40,047	0.192	0.142	19.2	0.10	405.3	441.8
Sweden	25,462	0.187	0.059	11.9	0.06	168.4	191.6
Norway	23,497	0.143	0.069	8.4	0.05	195.5	217.0
Denmark	21,496	0.304	0.091	16.3	0.09	258.6	278.2
Hungary	18,965	0.115	0.087	5.4	0.03	248.3	265.6
Portugal	18,352	0.229	0.115	10.5	0.06	329.4	346.1
Austria	17,354	0.198	0.105	8.6	0.05	298.8	314.7
Croatia	16,138	0.131	0.093	5.3	0.03	264.9	279.6
Georgia [87]	14,010	0.050	0.030	1.7	0.01	85.6	98.4
Total	1,393,649			685.8			287.7±104.1

Chapter 5

Conclusions and future work

This thesis presents a robust methodology for predictive modeling and optimization applied to critical scenarios presenting symptomatic crises. In an IoT context, this work focuses on a real scenario of ambulatory monitoring of people suffering from chronic migraines. Migraine is one of the most disabling neurological diseases. It affects around 15% of the population and leads to costs of € 1222 per patient per year to private and public health services in Europe. Currently there is no cure for the migraine, and migraine sufferers can only take their medication once they feel the pain. According to the medical literature, the intake of medicines in advance can abort the pain and patients can improve their quality of life. In this research we pose and answer the following questions: is it possible to model and predict a migraine? Can this prediction be launched in a specific time? To answer these questions, it has been developed a WBSN in an *eHealth* scenario where biomedical data are acquired in non-intrusive way. Data are processed, and to do so, there have been defined and implemented different MCC architectures. Due to WBSNs are prone to data loss, the methodology developed performs autonomous, and intelligent data repair and adaptive predictive modeling according to the sensors status. It has been demonstrated for the first time in the literature that it is possible to model and predict subjective pain of migraine attacks. In average, the system developed is able to detect 76% of crises, achieving average prediction horizons of 25 minutes with high accuracy and low rate of false positives. The developed methodology has been optimized to energy efficient, and in an hypothetical deployment of the network in Europe, it would lead to savings in the electricity bill of up to € 287.7 ± 104.1 million and € 1271.8 million for the health entities.

This research marks a starting point, and very promising future to the prediction of critical events in chronic diseases. However, further studies will require of a higher database to give statistical rigor to the technical work done in

this research. A broader study will allow, for example, the localization of clusters of migraine sufferers according to the behavior of their ANS that regulates changes in their hemodynamic variables.

There have been studied and implemented different architectures for the implementation of the predictive modeling system. In a general overview, there have been proposed a cascade of three major predictive sub-systems working together with an energy efficiency framework that support the implemented IoT network.

Sub-system I is the main and most important subsystem and it is the one that has more relevance and interest in this thesis. Sub-system I bases on fine-grained predictive models through information of hemodynamic variables. This is the most accurate sub-system that can provide alarms based only the information of an ambulatory and non-intrusive WBSN that patients wear. To create migraine predictive models, there have been used classic state-space algorithms (N4SID), and heuristic approaches such as Grammatical Evolutionary algorithms. However, other alternatives should be further studied. During this research there have been implemented some test using time series equations, which have shown hopeful results, but a more profound study is needed. Other interesting alternative that has been left out of the scope of this thesis, are the Kalman filters. These state-space models could lead to insightful results so they are able to deal with broken data as many research studies have shown, for example for tracking problems with path-information loss.

The fine-grained predictive algorithms, have been fed with raw data from four hemodynamic variables. Grammatical Evolutionary Algorithms performs automatic feature creation, however, this has not been fully studied and a deeper search is makes necessary. As a future work it would be interesting to create features as traditional time series studies do, such as mean, variance, energy, spectral information in different time windows. These features could fed the already implemented algorithms, or other like traditional machine learning algorithms (k-means, Support Vector Machine, *etc.*). To delimit the search space of all possibilities in the feature creation it could be used Grammatical Evolutionary as well. It would be interesting to see how windowing affects the prediction horizon, and a trade-off would become necessary. This will need to define different semi-overlapped and non-overlapped windows, which lead to more coarse predictions. The research group is already focusing further steps in this matter and previous test show promising results.

The study of more predictive solutions leads to less objective decisions about the goodness of each implementation. The simpler the models, the lower the energy consumption, but the lower the accuracy as well. Often, small increments

of accuracy require more complex models. Thus, this objective model selection would require of more sophisticated metrics as the Akaike Information Criterion (AIC) [2], which takes into account the complexity of the models and its quality metric. As a first step, in this thesis, it has been desired to improve the fit regardless the complexity of the model. AIC estimator would be an interesting metric when modeling energy efficient models as well.

Sub-system I includes the Sensor Dependent Model Selection System, a module that allows the system maintain the accuracy and prediction horizon when a sensor of the WBSN fails temporary. Sub-system I has been simulated using DEVS, and it has been demonstrated for the first time that it is possible to raise alarms in real time to predict migraines. There have been implemented two alternatives to perform signal repair: Gaussian Process Machine Learning techniques, and time series algorithms. There have been tested sensor disconnections, saturation and noisy signals; however, a complete sensor failure and the statistical study of the accuracy and improvement as a result of that failure is out of the scope of this research and will require future work. There is still room for improvement and comparison of with other alternatives, such as the Kalman filters. Kalman filters could lead to accurate results so that it provides prediction and deals with data loss in the same process. In addition, it would be necessary an study the energetic performance of these alternatives to achieve better energetic profiles of the implementation.

This research has focused only in the predictive modeling of the pain curve using hemodynamic variables. However, it might not be the best solution, neither the most suitable for a real implementation in a real device. Other solutions may consider the pain as a binary variable (yes/no pain) and, as aforementioned, using other features it could lead to clustering and machine learning algorithms for predictive detection. Nevertheless, the simplification of the pain curve to a binary variable misses a lot of details about the evolution of the pain. Despite it still requires from a decision module to raise the alarm, it would be an interesting research to perform the Gaussian fitting in real time, as well as the removal of spurious events. This should be done in small stretches, and its implementation might be tough. Prior to a real implementation, the simulation of these tests will raise objective comparison of both techniques.

Sub-system II is conceived for the prediction support based on a coarse-grained prediction obtained from environmental data. The environmental variables such as room and outdoor temperature, pressure or humidity are considered as trigger or precursor variables of the migraine disease. This system has a lower relevance in the system, and helps the Expert Decision System, **EDS**, in the decision making process. This sub-system has a marginal contribution to the current research results, but

it has been partially implemented as part of the whole system architecture. Due to the low number of migraine episodes with weather information, the results of these studies are not conclusive, however they present a hope on future experiment which will need of more data and the specific study of patients whose migraines are attributed to weather conditions.

On the other hand, there is plenty of information that can be gathered from premonitory symptoms and activity related to the migraine crises, such as triggers or daily activities, as an example. The number of commercial apps focused on these data collection reflects an interest of the population to register this information. Likewise, the results obtained encourage the research to keep ahead with these studies to further complete the Expert Decision System. During this research it has been noticed that there are few research studies considering temporal heterogeneous data to help decision making systems. A lot of decision making problems contemplate a set of events of the same type prior to the considered crisis. For example, several measurement of the blood pressure and information regarding nutrition before a heart failure—these events are probably measured at inconsistent periods of time or with unbalanced number of registrations. However, one of the major concerns about this in the migraine prediction problem using premonitory information is that, each one of the symptoms occurs only once before a migraine attack. In addition to this, in our case there is about thirty symptoms, and only few of them occur—not always the same ones—in each crisis. This would require a similar approach to the one presented by Zhao *et al.* in [190]. In this study the authors piece-wise and symbolic aggregate approximations for time series heterogeneous data. Our research has already detected this lack, and pursues an implementation to improve the Expert Decision System.

In this thesis there have been proposed different energy saving techniques at different abstraction levels: creation of energy efficient predictive models, efficient transmission of only relevant data and workload balancing policies for energy efficiency in a large deployment of the IoT solution. Regarding one of the most consuming task of a wearable device, the wireless transmission, in this research it was provided a solution where, after applying transformation algorithms, only a part of the transformation coefficients are transmitted. In this way, the energy savings are achieved at expense of the quality of the signal compressed, as it happens with the heart rate, computed from the ECG. If prediction methods are carried out in the back-end, or if compressed data are not needed for visualization (or other purposes which require signal quality), higher compression levels can be applied. It has been guessed that this could be possible if, in the optimization problem, we consider the quality level of the final migraine

prediction instead of the quality of the heart rate, for example. This assumption states that higher compression levels in the ECG affects more to the calculation of the heart rate than to the migraine prediction curve. A deeper study of the benefits of these decision techniques would be needed to be carried out in the future.

There might be situations where it is needed high quality signals in the back-end. A trade-off solution could be applied if it is still desired to maintain the autonomy of the wearable device. Wearable devices are designed to be worn during the whole day. It is desired that they operate effectively in risky situations, at least, when the patient is unsafe out of its home. A trade-off solution could be based on the control of the battery level using energetic policies. As an example, with high battery levels it could be transmitted high quality signals to the back-end, and as the day goes by, the battery level decreases, and the patient is closer to come back home. In this situations, later in the day, it could be applied higher compression levels—coarser predictions—to save energy. These situations need to be study, as they can be interesting from the point of view of research and real implementation in commercial devices.

As it can be seen, it has been achieved a functional migraine predictive methodology optimized and with energy constraints. The proof of concept of the methodology of the architectures proposed has been completed with high satisfaction degree. Many issues have been covered in this research, and obviously they have not been deeply studied. This opens the research to many other areas, which the author consider that would lead to further relevant innovative solutions.

Bibliography

- [1] ADNANE, M., JIANG, Z., CHOI, S., AND JANG, H. Detecting specific health-related events using an integrated sensor system for vital sign monitoring. *Sensors* 9, 9 (2009), 6897–6912.
- [2] AKAIKE, H. Information theory and an extension of the maximum likelihood principle. In *Selected papers of hirotugu akaike*. Springer, 1998, pp. 199–213.
- [3] ALEMDAR, H., AND ERSOY, C. Wireless sensor networks for healthcare: A survey. *Computer Networks* 54, 15 (2010), 2688–2710.
- [4] ALFONSECA, M., AND GIL, S. Evolving a predator–prey ecosystem of mathematical expressions with grammatical evolution. *Complexity* 20, 3 (2015), 66–83.
- [5] AMINIAN, M., AND NAJI, H. A hospital healthcare monitoring system using wireless sensor networks. *J. Health Med. Inform* 4 (2013), 121.
- [6] AQUINO-SANTOS, R., MARTINEZ-CASTRO, D., EDWARDS-BLOCK, A., AND MURILLO-PIEDRAHITA, A. F. Wireless sensor networks for ambient assisted living. *Sensors* 13, 12 (2013), 16384–16405.
- [7] ARROBA, P., RISCO-MARTÍN, J. L., ZAPATER, M., MOYA, J. M., AND AYALA, J. L. Enhancing regression models for complex systems using evolutionary techniques for feature engineering. *Journal of Grid Computing* 13, 3 (2015), 409–423.
- [8] ARROBA, P., RISCO-MARTÍN, J. L., ZAPATER, M., MOYA, J. M., AYALA, J. L., AND OLCOZ, K. Server power modeling for run-time energy optimization of cloud computing facilities. *Energy Procedia* 62 (2014), 401–410.
- [9] ATZORI, L., IERA, A., AND MORABITO, G. The internet of things: A survey. *Computer networks* 54, 15 (2010), 2787–2805.
- [10] AUGUSTINE, R. *Electromagnetic modelling of human tissues and its application on the interaction between antenna and human body in the BAN context*. PhD thesis, Université Paris-Est, 2009.

- [11] AULERY, A., DIGUET, J.-P., ROLAND, C., AND SENTIEYS, O. Low-complexity energy proportional posture/gesture recognition based on wbsn. In *Wearable and Implantable Body Sensor Networks (BSN), 2015 IEEE 12th International Conference on* (2015), IEEE, pp. 1–6.
- [12] BABUŠIAK, B., AND MOHYLOVA, J. The eeg signal prediction bz using neural network. *Advances in Electrical and Electronic Engineering (AEEE)* (2008).
- [13] BAHETI, P. K., AND GARUDADRI, H. An ultra low power pulse oximeter sensor based on compressed sensing. In *Wearable and Implantable Body Sensor Networks, 2009. BSN 2009. Sixth International Workshop on* (2009), IEEE, pp. 144–148.
- [14] BANAEE, H., AHMED, M. U., AND LOUTFI, A. Data mining for wearable sensors in health monitoring systems: a review of recent trends and challenges. *Sensors* 13, 12 (2013), 17472–17500.
- [15] BANK, J., AND MARTON, S. Hungarian migraine epidemiology. *Headache: The Journal of Head and Face Pain* 40, 2 (2000), 164–169.
- [16] BARHAK, J., ISAMAN, D. J., YE, W., AND LEE, D. Chronic disease modeling and simulation software. *Journal of biomedical informatics* 43, 5 (2010), 791–799.
- [17] BARSHAN, B., AND YÜKSEK, M. C. Recognizing daily and sports activities in two open source machine learning environments using body-worn sensor units. *The Computer Journal* 57, 11 (2014), 1649–1667.
- [18] BERGENTHAL, J. Final report model based engineering (mbe) subcommittee. February 2011 Web. < http://www.ndia.org/Divisions/Divisions/SystemsEngineering/Documents/Committees/M_S%20Committee/2011/February/NDIA-SE-MS_2011-02-15_Bergenthal.pdf (2011).
- [19] BOICHAT, N., KHALED, N., RINCON, F., AND ATIENZA, D. Wavelet-based ecg delineation on a wearable embedded sensor platform. In *2009 Sixth International Workshop on Wearable and Implantable Body Sensor Networks* (2009), IEEE, pp. 256–261.
- [20] BOX, G. Box and jenkins: time series analysis, forecasting and control. *A Very British Affair, ser. Palgrave Advanced Texts in Econometrics. Palgrave Macmillan UK* (2013), 161–215.
- [21] BOX, G. E., AND COX, D. R. An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)* (1964), 211–252.
- [22] BRAOJOS LOPEZ, R., MAMAGHANIAN, H., VALLEJOS, R., JAVIER, F., DIAS JUNIOR, A., ANSALONI, G., MURALI, S., AND ATIENZA ALONSO, D. Ultra-low power design of wearable cardiac monitoring systems. In *Proceedings of the IEEE/ACM 2014 Design Automation Conference (DAC)* (2014), vol. 1, IEEE/ACM Press, pp. 140–145.

- [23] BUCK, J. T., HA, S., LEE, E. A., AND MESSERSCHMITT, D. G. Ptolemy: A framework for simulating and prototyping heterogeneous systems.
- [24] BURNS, A., GREENE, B. R., MCGRATH, M. J., O'SHEA, T. J., KURIS, B., AYER, S. M., STROIESCU, F., AND CIONCA, V. ShimmerTM—a wireless sensor platform for noninvasive biomedical research. *Sensors Journal, IEEE* 10, 9 (2010), 1527–1534.
- [25] BURSTEIN, R., NOSEDA, R., AND BORSOOK, D. Migraine: multiple processes, complex pathophysiology. *Journal of Neuroscience* 35, 17 (2015), 6619–6629.
- [26] CASINO, F., LOPEZ-ITURRI, P., AGUIRRE, E., AZPILICUETA, L., SOLANAS, A., AND FALCONE, F. Dense wireless sensor network design for the implementation of smart health environments. In *Electromagnetics in Advanced Applications (ICEAA), 2015 International Conference on* (2015), IEEE, pp. 752–754.
- [27] CELIK, Y., EKUKLU, G., TOKUÇ, B., AND UTKU, U. Migraine prevalence and some related factors in turkey. *Headache: The Journal of Head and Face Pain* 45, 1 (2005), 32–36.
- [28] CESCÓN, M. *Modeling and prediction in diabetes physiology*. Department of Automatic Control, Lund University, 2013.
- [29] CHATTERJEE, S., DUTTA, K., XIE, H., BYUN, J., POTTATHIL, A., AND MOORE, M. Persuasive and pervasive sensing: A new frontier to monitor, track and assist older adults suffering from type-2 diabetes. In *System Sciences (HICSS), 2013 46th Hawaii International Conference on* (2013), IEEE, pp. 2636–2645.
- [30] CHEN, S.-W., AND CHAO, S.-C. Compressed sensing technology-based spectral estimation of heart rate variability using the integral pulse frequency modulation model. *IEEE journal of biomedical and health informatics* 18, 3 (2014), 1081–1090.
- [31] CHUNG, W.-Y., LEE, Y.-D., AND JUNG, S.-J. A wireless sensor network compatible wearable u-healthcare monitoring system using integrated ecg, accelerometer and spo 2. In *Engineering in Medicine and Biology Society, 2008. EMBS 2008. 30th Annual International Conference of the IEEE* (2008), IEEE, pp. 1529–1532.
- [32] CLIFTON, L., CLIFTON, D. A., PIMENTEL, M. A., WATKINSON, P. J., AND TARASSENKO, L. Gaussian processes for personalized e-health monitoring with wearable sensors. *IEEE Transactions on Biomedical Engineering* 60, 1 (2013), 193–197.
- [33] COLMENAR, J. M., RISCO-MARTIN, J. L., ATIENZA, D., AND HIDALGO, J. I. Multi-objective optimization of dynamic memory managers using grammatical evolution. In *Proceedings of the 13th GECCO* (2011), ACM, pp. 1819–1826.

- [34] DAHLÖF, C., AND LINDE, M. One-year prevalence of migraine in sweden: a population-based study in adults. *Cephalalgia* 21, 6 (2001), 664–671.
- [35] DAIKIN AC (AMERICAS), INC. Engineering Data SPLIT, FTXS-L Series.
- [36] DAMEN, J. A., HOOFT, L., SCHUIT, E., DEBRAY, T. P., COLLINS, G. S., TZOULAKI, I., LASSALE, C. M., SIONTIS, G. C., CHIOCCHIA, V., ROBERTS, C., ET AL. Prediction models for cardiovascular disease risk in the general population: systematic review. *bmj* 353 (2016), i2416.
- [37] DEAKIN, M., AND AL WAER, H. From intelligent to smart cities. *Intelligent Buildings International* 3, 3 (2011), 140–152.
- [38] DEB, K., PRATAP, A., AGARWAL, S., AND MEYARIVAN, T. A fast and elitist multiobjective genetic algorithm: Nsga-ii. *IEEE transactions on evolutionary computation* 6, 2 (2002), 182–197.
- [39] DHAMDHERE, A., SIVARAMAN, V., MATHUR, V., AND XIAO, S. Algorithms for transmission power control in biomedical wireless sensor networks. In *Asia-Pacific Services Computing Conference, 2008. APSCC'08. IEEE* (2008), IEEE, pp. 1114–1119.
- [40] DIENER, H.-C., DODICK, D. W., GOADSBY, P. J., LIPTON, R. B., OLESEN, J., AND SILBERSTEIN, S. D. Chronic migraine—classification, characteristics and treatment. *Nature Reviews Neurology* 8, 3 (2012), 162–171.
- [41] DILMAGHANI, R. S., BOBARSHAD, H., GHAVAMI, M., CHOGBKAR, S., AND WOLFE, C. Wireless sensor networks for monitoring physiological signals of multiple patients. *IEEE transactions on biomedical circuits and systems* 5, 4 (2011), 347–356.
- [42] DINH, H. T., LEE, C., NIYATO, D., AND WANG, P. A survey of mobile cloud computing: architecture, applications, and approaches. *Wireless communications and mobile computing* 13, 18 (2013), 1587–1611.
- [43] DJITOG, I., ALIYU, H. O., AND TRAORÉ, M. K. A model-driven framework for multi-paradigm modeling and holistic simulation of healthcare systems. *SIMULATION* (2017), 0037549717744888.
- [44] DOBKIN, B. H., AND DORSCH, A. The promise of mhealth: daily activity monitoring and outcome assessments by wearable sensors. *Neurorehabilitation and neural repair* 25, 9 (2011), 788–798.
- [45] DORSCHKY, E., SCHULDHAUS, D., KOERGER, H., AND ESKOFIER, B. A Framework for Early Event Detection for Wearable Systems. In *Proceedings of the 2015 ACM International Symposium on Wearable Computers* (2015), ACM, Ed., pp. 109–112.
- [46] EDDY, D. M., AND SCHLESSINGER, L. Archimedes a trial-validated model of diabetes. *Diabetes care* 26, 11 (2003), 3093–3101.

- [47] ESPINA, J., FALCK, T., MUEHLSTEFF, J., AND AUBERT, X. Wireless body sensor network for continuous cuff-less blood pressure monitoring. In *Medical Devices and Biosensors, 2006. 3rd IEEE/EMBS International Summer School on* (2006), IEEE, pp. 11–15.
- [48] ESTEFAN, J. A., ET AL. Survey of model-based systems engineering (mbse) methodologies. *IncoSE MBSE Focus Group* 25, 8 (2007).
- [49] ESWARI, T., SAMPATH, P., LAVANYA, S., ET AL. Predictive methodology for diabetic data analysis in big data. *Procedia Computer Science* 50 (2015), 203–208.
- [50] EUROSTAT. Electricity price statistics. http://ec.europa.eu/eurostat/statistics-explained/index.php/Electricity_price_statistics.
- [51] EUROSTAT. Population on 1 january by age and sex. http://ec.europa.eu/eurostat/en/web/products-datasets/-/DEMO_PJAN. Accessed: 2016-06-01.
- [52] EYSENBACH, G. What is e-health? *Journal of medical Internet research* 3, 2 (2001), e20.
- [53] FALLAHZADEH, R., ORTIZ, J. P., AND GHASEMZADEH, H. Adaptive compressed sensing at the fingertip of internet-of-things sensors: An ultra-low power activity recognition. In *2017 Design, Automation & Test in Europe Conference & Exhibition (DATE)* (2017), IEEE, pp. 996–1001.
- [54] FERNANDEZ, F., AND PALLIS, G. C. Opportunities and challenges of the internet of things for healthcare: Systems engineering perspective. In *Wireless Mobile Communication and Healthcare (Mobihealth), 2014 EAI 4th International Conference on* (2014), IEEE, pp. 263–266.
- [55] FINKELSTEIN, J., ET AL. Machine learning approaches to personalize early prediction of asthma exacerbations. *Annals of the New York Academy of Sciences* 1387, 1 (2017), 153–165.
- [56] FRIEDMAN, D. I., AND DE VER DYE, T. Migraine and the environment. *Headache: The Journal of Head and Face Pain* 49, 6 (2009), 941–952.
- [57] GAGO, A., SOBRADO, M., MORA, J. V., PAGÁN, J., ORBE, D., IRENE, M., AND AYALA, J. L. Método para determinar el nivel de activación del sistema trigémino-vascular, 02 2016.
- [58] GAGO, A., SOBRADO, M., MORA, J. V., PAGÁN, J., ORBE, D., IRENE, M., AND AYALA, J. L. Method for determining the degree of activation of the trigeminovascular system, 01 2017.
- [59] GAGO-VEIGA, A. B., DE ORBE-IZQUIERDO, M. I., SOBRADO, M., PAGÁN, J., CARRERAS, M., AYALA, J. L., AND VIVANCOS, J. Monitorización ambulatoria no invasiva de variable biométricas en pacientes con migraña. ¿es posible predecir una crisis? In *XLVI Reunión Anual de la Sociedad Española de Neurología* (2014), Sociedad Española de Neurología, pp. 94–103.

- [60] GEORGOULAS, G., GAVRILIS, D., TSOULOS, I. G., STYLIOU, C., BERNARDES, J., AND GROUMPOS, P. P. Novel approach for fetal heart rate classification introducing grammatical evolution. *Biomedical Signal Processing and Control* 2, 2 (2007), 69–79.
- [61] GHASEMZADEH, H., AMINI, N., AND ET AL. Power-aware computing in wearable sensor networks: An optimal feature selection. *IEEE Transactions on Mobile Computing* 14, 4 (2015), 800–812.
- [62] GIFFIN, N., RUGGIERO, L., LIPTON, R., SILBERSTEIN, S., TVEDSKOV, J., OLESEN, J., ALTMAN, J., GOADSBY, P. J., AND MACRAE, A. Premonitory symptoms in migraine an electronic diary study. *Neurology* 60, 6 (2003), 935–940.
- [63] GOADSBY, P., ZANCHIN, G., GERAUD, G. A., DE KLIPPEL, N., DIAZ-INSA, S., GOBEL, H., CUNHA, L., IVANOFF, N., FALQUES, M., AND FORTEA, J. Early vs. non-early intervention in acute migraine—‘act when mild (awm)’: a double-blind, placebo-controlled trial of almotriptan. *Cephalalgia* 28, 4 (2008), 383–391.
- [64] GRADL, S., KUGLER, P., LOHMÜLLER, C., AND ESKOFIER, B. Real-time eeg monitoring and arrhythmia detection using android-based mobile devices. In *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE* (2012), IEEE, pp. 2452–2455.
- [65] GROH, B., REINFELDER, S., STREICHER, M., TARABEN, A., AND ESKOFIER, B. Movement prediction in rowing using a Dynamic Time Warping based stroke detection. In *IEEE Ninth International Conference on Intelligent Sensors, Sensor Networks and Information Processing (ISSNIP 2014)* (2014), IEEE, Ed., pp. 1–6.
- [66] GUBBI, J., BUYYA, R., MARUSIC, S., AND PALANISWAMI, M. Internet of things (iot): A vision, architectural elements, and future directions. *Future generation computer systems* 29, 7 (2013), 1645–1660.
- [67] GUGLIELMO, R., MARTINOTTI, G., DI GIANNANTONIO, M., AND JANIRI, L. A possible new option for migraine management: Agomelatine. *Clinical neuropharmacology* 36, 2 (2013), 65–67.
- [68] GÜLER, I., AND ÜBEYLI, E. D. Adaptive neuro-fuzzy inference system for classification of eeg signals using wavelet coefficients. *Journal of neuroscience methods* 148, 2 (2005), 113–121.
- [69] HAGHI, M., THUROW, K., AND STOLL, R. Wearable devices in medical internet of things: scientific research and commercially available devices. *Healthcare informatics research* 23, 1 (2017), 4–15.
- [70] HALL, M., FRANK, E., HOLMES, G., PFAHRINGER, B., REUTEMANN, P., AND WITTEN, I. H. The weka data mining software: an update. *ACM SIGKDD explorations newsletter* 11, 1 (2009), 10–18.

- [71] HAMILTON, J. D. State-space models. *Handbook of econometrics 4* (1994), 3039–3080.
- [72] HAO, Y., AND FOSTER, R. Wireless body sensor networks for health-monitoring applications. *Physiological measurement 29*, 11 (2008), R27.
- [73] HASSINGER, H. J., SEMENCHUK, E. M., AND O’BRIEN, W. H. Cardiovascular responses to pain and stress in migraine. *Headache: The Journal of Head and Face Pain 39*, 9 (1999), 605–615.
- [74] HELDBERG, B. E., KAUTZ, T., LEUTHEUSER, H., HOPFENGARTNER, R., KASPER, B. S., AND ESKOFIER, B. M. Using wearable sensors for semiology-independent seizure detection-towards ambulatory monitoring of epilepsy. In *Engineering in Medicine and Biology Society (EMBC), 2015 37th Annual International Conference of the IEEE* (2015), IEEE, pp. 5593–5596.
- [75] HENARES, K., PAGÁN, J., AND RISCO-MARTÍN, J. L. *Sistema avanzado de predicción de crisis migrañosas*. 2018.
- [76] HERO. Heuristic optimization library. <https://github.com/jlrisco/hero>, 2016.
- [77] HERSHEY, A. D. Current approaches to the diagnosis and management of paediatric migraine. *The Lancet Neurology 9*, 2 (2010), 190–204.
- [78] HIDALGO, J. I., COLMENAR, J. M., RISCO-MARTIN, J. L., CUESTA-INFANTE, A., MAQUEDA, E., BOTELLA, M., AND RUBIO, J. A. Modeling glycemia in humans by means of grammatical evolution. *Applied Soft Computing 20* (2014), 40–53.
- [79] HOFFMANN, J., LO, H., NEEB, L., MARTUS, P., AND REUTER, U. Weather sensitivity in migraineurs. *Journal of neurology 258*, 4 (2011), 596–602.
- [80] HOULE, T. T., TURNER, D. P., AND PENZIEN, D. B. How does the migraine attack stop? it is not the trigger: common headache triggers do not predict cessation of pain. *Headache 52*, 1 (2012), 189–190.
- [81] HOUSING, U., AND OTB, M. S. Smart cities ranking of european medium-sized cities.
- [82] HU, X., HU, X., ZEIGLER, B. P., AND MITTAL, S. Variable structure in devs component-based modeling and simulation. *Simulation 81*, 2 (2005), 91–102.
- [83] HU, X. H., RASKIN, N. H., COWAN, R., MARKSON, L. E., AND BERGER, M. L. Treatment of migraine with rizatriptan: when to take the medication. *Headache: The Journal of Head and Face Pain 42*, 1 (2002), 16–20.
- [84] HUANG, G., ZHANG, Y., CAO, J., STEYN, M., AND TARAPOREWALLA, K. Online mining abnormal period patterns from multiple medical sensor data streams. *World Wide Web 17*, 4 (2014), 569–587.

- [85] INC., T. M. Matlab and econometrics toolbox. version 7.14.0.739 (r2012a), 2012.
- [86] INC., T. M. Matlab and system identification toolbox. version 7.14.0.739 (r2012a), 2012.
- [87] INOGATE. A review of energy tariffs in inogate partner countries 2015. <http://www.inogate.org/>. Accessed: 2016-06-01.
- [88] JANG, J. R. *MATLAB: Fuzzy logic toolbox user's guide: Version 1*. Math Works, 1997.
- [89] KATSARAVA, Z., DZAGNIDZE, A., KUKAVA, M., MIRVELASHVILI, E., DJIBUTI, M., JANELIDZE, M., JENSEN, R., STOVNER, L., STEINER, T., ET AL. Primary headache disorders in the republic of georgia prevalence and risk factors. *Neurology* 73, 21 (2009), 1796–1803.
- [90] KEWMAN, D., AND ROBERTS, A. H. Skin temperature biofeedback and migraine headaches. *Biofeedback and Self-Regulation* 5, 3 (1980), 327–345.
- [91] KHALEGHI, B., KHAMIS, A., KARRAY, F. O., AND RAZAVI, S. N. Multisensor data fusion: A review of the state-of-the-art. *Information Fusion* 14, 1 (2013), 28–44.
- [92] KHAN, A. N., KIAH, M. M., KHAN, S. U., AND MADANI, S. A. Towards secure mobile cloud computing: A survey. *Future Generation Computer Systems* 29, 5 (2013), 1278–1299.
- [93] KLUCKEN, J., BARTH, J., AND ET AL. Unbiased and mobile gait analysis detects motor impairment in parkinson's disease. *PloS one* 8, 2 (2013), e56956.
- [94] KONG, L., XIA, M., LIU, X.-Y., CHEN, G., GU, Y., WU, M.-Y., AND LIU, X. Data loss and reconstruction in wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems* 25, 11 (2014), 2818–2828.
- [95] KOOMEY, J. Growth in data center electricity use 2005 to 2010. Tech. rep., Analytics Press, Oakland, CA, 2011.
- [96] KUGLER, P., SCHULDHAUS, D., JENSEN, U., AND ESKOFIER, B. Mobile recording system for sport applications. In *Proceedings of the 8th international symposium on computer science in sport (IACSS 2011)*, Liverpool (2011), pp. 67–70.
- [97] KUTUR, C. S., RAVI KANTH, K., AND SREE KANTH, K. Improved algorithm for prediction of heart disease using case based reasoning technique on non-binary datasets. *IJRCCT* 1, 7 (2012), 420–424.
- [98] LAMPL, C., BUZATH, A., BAUMHACKL, U., AND KLINGLER, D. One-year prevalence of migraine in austria: a nation-wide survey. *Cephalalgia* 23, 4 (2003), 280–286.

- [99] LAN, M., SAMY, L., ALSHURAF, N., SUH, M.-K., GHASEMZADEH, H., MACABASCO-O'CONNELL, A., AND SARRAFZADEH, M. Wanda: An end-to-end remote health monitoring and analytics system for heart failure patients. In *Proceedings of the conference on Wireless Health* (2012), ACM, p. 9.
- [100] LANTÉRI-MINET, M., VALADE, D., GERAUD, G., CHAUTARD, M., AND LUCAS, C. Migraine and probable migraine—results of framig 3, a french nationwide survey carried out according to the 2004 ihs classification. *Cephalalgia* 25, 12 (2005), 1146–1158.
- [101] LAUNER, L. J., TERWINDT, G. M., AND FERRARI, M. D. The prevalence and characteristics of migraine in a population-based cohort the gem study. *Neurology* 53, 3 (1999), 537–537.
- [102] LEE, K. H., KUNG, S.-Y., AND VERMA, N. Low-energy formulations of support vector machine kernel functions for biomedical sensor applications. *Journal of Signal Processing Systems* 69, 3 (2012), 339–349.
- [103] LEE, S. Y., HONG, J. H., HSIEH, C. H., LIANG, M. C., CHIEN, S. Y. C., AND LIN, K. H. Low-power wireless ecg acquisition and classification system for body sensor networks. *IEEE Journal of Biomedical and Health Informatics* 19, 1 (Jan 2015), 236–246.
- [104] LEUTHEUSER, H., GRADL, S., KUGLER, P., ANNEKEN, L., ARNOLD, M., ACHENBACH, S., AND ESKOFIER, B. M. Comparison of real-time classification systems for arrhythmia detection on android-based mobile devices. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE* (2014), IEEE, pp. 2690–2693.
- [105] LINDE, M., GUSTAVSSON, A., STOVNER, L., STEINER, T., BARRÉ, J., KATSARAVA, Z., LAINEZ, J., LAMPL, C., LANTÉRI-MINET, M., RASTENYTE, D., ET AL. The cost of headache disorders in europe: the eurolight project. *European journal of neurology* 19, 5 (2012), 703–711.
- [106] LIPTON, R. B., STEWART, W. F., AND SCHER, A. I. Epidemiology and economic impact of migraine. *Current medical research and opinion* 17, sup1 (2001), s4–12.
- [107] LIU, Y., DE VOS, M., AND VAN HUFFEL, S. Compressed sensing of multichannel eeg signals: the simultaneous cosparsity and low-rank optimization. *IEEE Transactions on Biomedical Engineering* 62, 8 (2015), 2055–2061.
- [108] LOPES, M., DUNN, J. D., CALHOUN, A., AND RAPOPORT, A. M. Concepts in acute migraine management: Clinical and managed care perspectives. *Am. J. Pharm* 4 (2012), 201–206.
- [109] LOVATI, C., ET AL. *Breathing Sleep Disturbances and Migraine: A Dangerous Synergy or a Favorable Antagonism?* 2012.

- [110] LUCERO, A. Simulation of batch scheduling using real production-ready software tools. *Proceedings of the 5th IBERGRID* (2011).
- [111] LUCIANI, R., CARTER, D., MANNIX, L., HEMPHILL, M., DIAMOND, M., AND CADY, R. Prevention of migraine during prodrome with naratriptan. *Cephalalgia* 20, 2 (2000), 122–126.
- [112] MAMAGHANIAN, H., KHALED, N., ATIENZA, D., AND VANDERGHEYNST, P. Compressed sensing for real-time energy-efficient ecg compression on wireless body sensor nodes. *Biomedical Engineering, IEEE Transactions on* 58, 9 (2011), 2456–2466.
- [113] MARTÍN, Á., PAGÁN, J., AND AYALA, J. L. *Implementación de técnicas de bajo consumo en WBSN*. 2017.
- [114] MASON, J. W., RAMSETH, D. J., CHANTER, D. O., MOON, T. E., GOODMAN, D. B., AND MENDZELEVSKI, B. Electrocardiographic reference ranges derived from 79,743 ambulatory subjects. *Journal of electrocardiology* 40, 3 (2007), 228–234.
- [115] MATÍAS-GUIU, J., PORTA-ETESSAM, J., MATEOS, V., DÍAZ-INSA, S., LOPEZ-GIL, A., AND FERNÁNDEZ, C. One-year prevalence of migraine in spain: a nationwide population-based survey. *Cephalalgia* 31, 4 (2011), 463–470.
- [116] MATT STANSBERRY, J. K. Uptime institute 2013 data center industry survey. Tech. rep., Uptime Institute, 2013.
- [117] MERIKANGAS, K. R., WHITAKER, A. E., ISLER, H., AND ANGST, J. The zurich study: Xxiii. epidemiology of headache syndromes in the zurich cohort study of young adults. *European archives of psychiatry and clinical neuroscience* 244, 3 (1994), 145–152.
- [118] MILENKOVIĆ, A., OTTO, C., AND JOVANOVIĆ, E. Wireless sensor networks for personal health monitoring: Issues and an implementation. *Computer communications* 29, 13 (2006), 2521–2533.
- [119] MILOSEVIC, J., DITTRICH, A., FERRANTE, A., MALEK, M., QUIROS, C. R., BRAOJOS, R., ANSALONI, G., AND ATIENZA, D. Risk assessment of atrial fibrillation: a failure prediction approach. In *Computing in Cardiology Conference (CinC), 2014* (2014), IEEE, pp. 801–804.
- [120] MONTEIRO, J. M. P. Cefaleias: estudo epidemiológico e clínico de uma população urbana.
- [121] MORMANN, F., KREUZ, T., RIEKE, C., ANDRZEJAK, R. G., KRASKOV, A., DAVID, P., ELGER, C. E., AND LEHNERTZ, K. On the predictability of epileptic seizures. *Clinical neurophysiology* 116, 3 (2005), 569–587.
- [122] OF SWISS ELECTRICITY COMPANIES VSE-AES, A. Swiss electricity prices- development 1991-2015. <http://www.strom.ch/de/metanavigation/download.html>. Accessed: 2016-06-01.

- [123] ORDÁS, C. M., CUADRADO, M. L., RODRÍGUEZ-CAMBRÓN, A. B., CASAS-LIMÓN, J., DEL PRADO, N., AND PORTA-ETESSAM, J. Increase in body temperature during migraine attacks. *Pain medicine* 14, 8 (2013), 1260–1264.
- [124] PAGÁN, J., FALLAHZADEH, R., GHASEMZADEH, H., MOYA, J. M., RISCO-MARTÍN, J. L., AND AYALA, J. L. An optimal approach for low-power migraine prediction models in the state-of-the-art wireless monitoring devices. In *Proceedings of the Conference on Design, Automation & Test in Europe* (2017), European Design and Automation Association, pp. 1297–1302.
- [125] PAGÁN, J., MOYA, J. M., RISCO-MARTÍN, J. L., AND AYALA, J. L. Advanced migraine prediction simulation system. In *Proceedings of the Summer Simulation Multi-Conference* (San Diego, CA, USA, 2017), SummerSim '17, Society for Computer Simulation International, pp. 24:1–24:12.
- [126] PAGÁN, J., ORBE, D., IRENE, M., GAGO, A., SOBRADO, M., RISCO-MARTÍN, J. L., MORA, J. V., MOYA, J. M., AND AYALA, J. L. Robust and accurate modeling approaches for migraine per-patient prediction from ambulatory data. *Sensors* 15, 7 (2015), 15419–15442.
- [127] PAGÁN, J., RISCO-MARTÍN, J. L., MOYA, J. M., AND AYALA, J. L. Grammatical evolutionary techniques for prompt migraine prediction. In *Proceedings of the Genetic and Evolutionary Computation Conference 2016* (2016), ACM, pp. 973–980.
- [128] PAGÁN, J., RISCO-MARTÍN, J. L., MOYA, J. M., AND AYALA, J. L. Modeling methodology for the accurate and prompt prediction of symptomatic events in chronic diseases. *Journal of biomedical informatics* 62 (2016), 136–147.
- [129] PAGÁN, J., RISCO-MARTÍN, J. L., MOYA, J. M., AND AYALA, J. L. A real-time framework for a devs-based migraine prediction simulator system. *MAEB 2016* (2016).
- [130] PAGÁN, J., ZAPATER, M., AND AYALA, J. L. Power transmission and workload balancing policies in ehealth mobile cloud computing scenarios. *Future Generation Computer Systems* 78 (2018), 587–601.
- [131] PAGE, A., HIJAZI, S., ASKAN, D., KANTARCI, B., AND SOYATA, T. Research directions in cloud-based decision support systems for health monitoring using internet-of-things driven data acquisition. *International Journal of Services Computing (IJSC)* 4, 4 (2016), 18–34.
- [132] PASSCHIER, J., GOUDSWAARD, P., AND ORLEBEKE, J. F. Abnormal extracranial vasomotor response in migraine sufferers to real-life stress. *Journal of psychosomatic research* 37, 4 (1993), 405–414.

- [133] PATEL, S., PARK, H., BONATO, P., CHAN, L., AND RODGERS, M. A review of wearable sensors and systems with application in rehabilitation. *Journal of neuroengineering and rehabilitation* 9, 1 (2012), 21.
- [134] PFAFFENRATH, V., FENDRICH, K., VENNEMANN, M., MEISINGER, C., LADWIG, K.-H., EVERS, S., STRAUBE, A., HOFFMANN, W., AND BERGER, K. Regional variations in the prevalence of migraine and tension-type headache applying the new ihs criteria: the german dmkg headache study. *Cephalalgia* 29, 1 (2009), 48–57.
- [135] POLI, R., LANGDON, W. B., MCPHEE, N. F., AND KOZA, J. R. *A field guide to genetic programming*. Lulu. com, 2008.
- [136] PORTA-ETESSAM, J., CUADRADO, M. L., RODRÍGUEZ-GÓMEZ, O., VALENCIA, C., AND GARCÍA-PTACEK, S. Hypothermia during migraine attacks. *Cephalalgia* 30, 11 (2010), 1406–1407.
- [137] RASHIDI, P., AND MIHAILIDIS, A. A survey on ambient-assisted living tools for older adults. *IEEE journal of biomedical and health informatics* 17, 3 (2013), 579–590.
- [138] RASMUSSEN, C. E., AND WILLIAMS, C. K. *Gaussian processes for machine learning*, vol. 1. MIT press Cambridge, 2006.
- [139] REINHOLD JR, H. E., AND AUERBACH, A. A. Ambulatory monitoring system with real time analysis and telephone transmission, July 30 1985. US Patent 4,531,527.
- [140] RENZO, M. D., BUEHRER, R. M., AND TORRES, J. Pulse shape distortion and ranging accuracy in uwb-based body area networks for full-body motion capture and gait analysis. In *GLOBECOM (2007)*, IEEE, pp. 3775–3780.
- [141] RICO, A., PAGÁN, J., AND MALAGÓN, P. *Diseño e implementación de algoritmos de aprendizaje automático para la ayuda a la predicción de crisis sintomáticas de migrañas a partir señales ambientales y biométricas*. 2017.
- [142] RINCÓN, F., RECAS, J., KHALED, N., AND ATIENZA, D. Development and evaluation of multilead wavelet-based ecg delineation algorithms for embedded wireless sensor nodes. *Information Technology in Biomedicine, IEEE Transactions on* 15, 6 (2011), 854–863.
- [143] RISCO-MARTÍN, J. L., JESÚS, M., MITTAL, S., AND ZEIGLER, B. P. eudevts: Executable uml with devts theory of modeling and simulation. *Simulation* 85, 11-12 (2009), 750–777.
- [144] RISCO-MARTÍN, J. L., MITTAL, S., FABERO, J. C., MALAGON, P., AND AYALA, J. L. Real-time hardware/software co-design using devts-based transparent m&s framework. In *SummerSim-SCSC, 2016* (2016).
- [145] ROBERTS, N. E., OH, S., AND WENTZLOFF, D. D. Exploiting channel periodicity in body sensor networks. *Emerging and Selected Topics in Circuits and Systems, IEEE Journal on* 2, 1 (2012), 4–13.

- [146] RUDOLF, G., FERTNER, C., KRAMAR, H., KALASEK, R., PICHLER-MILANOVIC, N., AND MEIJERS, E. Smart cities-ranking of european medium-sized cities. *Rapport technique, Vienna Centre of Regional Science* (2007).
- [147] RUSSELL, M. B., LEVI, N., ŠALTYTĖ-BENTH, J., AND FENGER, K. Tension-type headache in adolescents and adults: a population based study of 33,764 twins. *European journal of epidemiology* 21, 2 (2006), 153–160.
- [148] RYAN, C., AND O’NEILL, M. Grammatical evolution: A steady state approach. *Late Breaking Papers, Genetic Programming 1998* (1998), 180–185.
- [149] SALINAS-HILBURG, J. C., ZAPATER, M., RISCO-MARTIN, J. L., MOYA, J. M., AND AYALA, J. L. Using grammatical evolution techniques to model the dynamic power consumption of enterprise servers. In *Complex, Intelligent, and Software Intensive Systems (CISIS), 2015 Ninth International Conference on* (2015), IEEE, pp. 110–117.
- [150] SANTAMBROGIO, M. D., AYALA, J. L., CAMPANONI, S., CATTANEO, R., DURELLI, G. C., FERRONI, M., NACCI, A., PAGÁN, J., ZAPATER, M., AND VALLEJO, M. Power-awareness and smart-resource management in embedded computing systems. In *Hardware/Software Codesign and System Synthesis (CODES+ ISSS), 2015 International Conference on* (2015), IEEE, pp. 94–103.
- [151] SCHWIEBERT, L., GUPTA, S. K., AND WEINMANN, J. Research challenges in wireless networks of biomedical sensors. In *Proceedings of the 7th annual international conference on Mobile computing and networking* (2001), ACM, pp. 151–165.
- [152] SEEPERS, R. M., STRYDIS, C., PERIS-LOPEZ, P., SOURDIS, I., AND DE ZEEUW, C. I. Peak misdetection in heart-beat-based security: Characterization and tolerance. In *Engineering in Medicine and Biology Society (EMBC), 2014 36th Annual International Conference of the IEEE* (2014), IEEE, pp. 5401–5405.
- [153] SHEHABI, A., SMITH, S. J., SARTOR, D. A., BROWN, R. E., HERRLIN, M., KOOMEY, J. G., MASANET, E. R., HORNER, N., AZEVEDO, I. L., AND LINTNER, W. United states data center energy usage report. Tech. rep., 06/2016 2016.
- [154] SILBERSTEIN, S. D., DODICK, D. W., SAPER, J., HUH, B., SLAVIN, K. V., SHARAN, A., REED, K., NAROUZE, S., MOGILNER, A., GOLDSTEIN, J., ET AL. Safety and efficacy of peripheral nerve stimulation of the occipital nerves for the management of chronic migraine: results from a randomized, multicenter, double-blinded, controlled study. *Cephalalgia* 32, 16 (2012), 1165–1179.

- [155] SINGH, R. R., CONJETI, S., AND BANERJEE, R. An approach for real-time stress-trend detection using physiological signals in wearable computing systems for automotive drivers. In *Intelligent Transportation Systems (ITSC), 2011 14th International IEEE Conference on* (2011), IEEE, pp. 1477–1482.
- [156] SJAASTAD, O., AND BAKKETEIG, L. Migraine without aura: comparison with cervicogenic headache. vågå study of headache epidemiology. *Acta Neurologica Scandinavica* 117, 6 (2008), 377–383.
- [157] SMITH, D. B., MINIUTTI, D., AND HANLEN, L. W. Characterization of the body-area propagation channel for monitoring a subject sleeping. *Antennas and Propagation, IEEE Transactions on* 59, 11 (2011), 4388–4392.
- [158] SOL, H. G., CEES, A. T., DE VRIES ROBBÉ, P. F., ET AL. *Expert systems and artificial intelligence in decision support systems: proceedings of the Second Mini Euroconference, Lunteren, The Netherlands, 17–20 November 1985*. Springer Science & Business Media, 2013.
- [159] STANSBERRY, M., AND KUDRITZKI, J. Uptime institute 2012 data center industry survey. *Uptime Institute Survey* (2012).
- [160] STATPOINT TECHNOLOGIES, I. Statgraphics centurion xvii user manual, 2014.
- [161] STEELE, B. G., HOLT, L., BELZA, B., FERRIS, S., LAKSHMINARYAN, S., AND BUCHNER, D. M. Quantitating physical activity in copd using a triaxial accelerometer. *CHEST Journal* 117, 5 (2000), 1359–1367.
- [162] STEINER, T., SCHER, A., STEWART, W., KOLODNER, K., LIBERMAN, J., AND LIPTON, R. The prevalence and disability burden of adult migraine in england and their relationships to age, gender and ethnicity. *Cephalalgia* 23, 7 (2003), 519–527.
- [163] STOVNER, L. J., AND ANDREE, C. Prevalence of headache in europe: a review for the eurolight project. *The journal of headache and pain* 11, 4 (2010), 289–299.
- [164] SUH, M.-K., EVANGELISTA, L. S., CHEN, V., HONG, W.-S., MACBETH, J., NAHAPETIAN, A., FIGUERAS, F.-J., AND SARRAFZADEH, M. Wanda b.: Weight and activity with blood pressure monitoring system for heart failure patients. In *World of Wireless Mobile and Multimedia Networks (WoWMoM), 2010 IEEE International Symposium on a* (2010), IEEE, pp. 1–6.
- [165] SUN, F.-T., KUO, C., CHENG, H.-T., BUTHPITIYA, S., COLLINS, P., AND GRISS, M. Activity-aware mental stress detection using physiological sensors. In *International Conference on Mobile Computing, Applications, and Services* (2010), Springer, pp. 211–230.

- [166] SURREL, G., RINCON, F., MURALI, S., AND ATIENZA, D. Design of ultra-low-power smart wearable systems. In *Test Symposium (LATS), 2015 16th Latin-American* (2015), IEEE, pp. 1–2.
- [167] TANAKA, H., MONAHAN, K. D., AND SEALS, D. R. Age-predicted maximal heart rate revisited. *Journal of the American College of Cardiology* 37, 1 (2001), 153–156.
- [168] TANG, Q., TUMMALA, N., GUPTA, S. K. S., AND SCHWIEBERT, L. Communication scheduling to minimize thermal effects of implanted biosensor networks in homogeneous tissue. *Biomedical Engineering, IEEE Transactions on* 52, 7 (2005), 1285–1294.
- [169] THAKOR, N. V., WEBSTER, J. G., AND TOMPKINS, W. J. Estimation of qrs complex power spectra for design of a qrs filter. *IEEE Transactions on biomedical engineering*, 11 (1984), 702–706.
- [170] THONGKAM, J., XU, G., AND ZHANG, Y. Adaboost algorithm with random forests for predicting breast cancer survivability. In *Neural Networks, 2008. IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Conference on* (2008), IEEE, pp. 3062–3069.
- [171] TOBOLA, A., KORPOK, O., LEUTHEUSER, H., SCHMITZ, B., HOFMANN, C., STRUCK, M., WEIGAND, C., ESKOFIER, B., HEUBERGER, A., AND FISCHER, G. System Design Impacts on Battery Runtime of Wearable Medical Sensors. In *Proceedings of International Conference on Mobile and Information Technologies in Medicine and Health* (2014), D. Novák, Ed., pp. 1–4.
- [172] TOBOLA, A., STREIT, F. J., ESPIG, C., KORPOK, O., SAUTER, C., LANG, N., SCHMITZ, B., HOFMANN, C., STRUCK, M., WEIGAND, C., ET AL. Sampling rate impact on energy consumption of biomedical signal processing systems. In *Wearable and Implantable Body Sensor Networks (BSN), 2015 IEEE 12th International Conference on* (2015), IEEE, pp. 1–6.
- [173] VALLEJO, M., RECAS, J., AND AYALA, J. L. Proactive and reactive transmission power control for energy-efficient on-body communications. *Sensors* 15, 3 (2015), 5914–5934.
- [174] VALLEJO, M., RECAS, J., DEL VALLE, P. G., AND AYALA, J. L. Accurate human tissue characterization for energy-efficient wireless on-body communications. *Sensors* 13, 6 (2013), 7546–7569.
- [175] VALLEJO, M., RECAS PIORNO, J., AND AYALA RODRIGO, J. L. A link quality estimator for power-efficient communication over on-body channels. In *Embedded and Ubiquitous Computing (EUC), 2014 12th IEEE International Conference on* (2014), IEEE, pp. 250–257.
- [176] VAN BREDA, W., HOOGENDOORN, M., EIBEN, A., AND BERKING, M. Assessment of temporal predictive models for health care using a formal method. *Computers in biology and medicine* 87 (2017), 347–357.

- [177] VAN OVERSCHEE, P., AND DE MOOR, B. N4sid: Subspace algorithms for the identification of combined deterministic-stochastic systems. *Automatica* 30, 1 (1994), 75–93.
- [178] VENKATRAMAN, A. Global census shows datacentre power demand grew 63% in 2012.
- [179] VIÑAS, F., PAGÁN, J., AND RISCO-MARTÍN, J. L. *Definición, desarrollo e implementación de un sistema experto de decisión para entornos de modelado e-health*. 2017.
- [180] VOLLONO, C., GNONI, V., TESTANI, E., DITTONI, S., LOSURDO, A., COLICCHIO, S., DI BLASI, C., MAZZA, S., FARINA, B., AND DELLA MARCA, G. Heart rate variability in sleep-related migraine without aura. *Journal of clinical sleep medicine: JCSM: official publication of the American Academy of Sleep Medicine* 9, 7 (2013), 707.
- [181] VON MACKENSEN, S., HOEPPE, P., MAAROUF, A., TOURIGNY, P., AND NOWAK, D. Prevalence of weather sensitivity in germany and canada. *International journal of biometeorology* 49, 3 (2005), 156–166.
- [182] WAELKENS, J. Dopamine blockade with domperidone: bridge between prophylactic and abortive treatment of migraine? a dose-finding study. *Cephalalgia* 4, 2 (1984), 85–90.
- [183] WAINER, G. A., MADHOUN, R., AND AL-ZOUBI, K. Distributed simulation of devs and cell-devs models in cd++ using web-services. *Simulation Modelling Practice and Theory* 16, 9 (2008), 1266–1292.
- [184] WHO. Headache disorders, fact sheet n°277. <http://www.who.int/mediacentre/factsheets/fs277/en/>, 2012. Accessed: 2018-03-01.
- [185] WOLD, H. *A study in the analysis of stationary time series*. PhD thesis, Almqvist & Wiksell, 1938.
- [186] xDEVS. Devs m&s framework. <https://github.com/jlrisco/xdevs>, 2016.
- [187] XIAO, S., DHAMDHERE, A., SIVARAMAN, V., AND BURDETT, A. Transmission power control in body area sensor networks for healthcare monitoring. *Selected Areas in Communications, IEEE Journal on* 27, 1 (2009), 37–48.
- [188] ZAPATER, M., ARROBA, P., AYALA, J. L., MOYA, J. M., AND OLCOZ, K. A novel energy-driven computing paradigm for e-health scenarios. *Future Generation Computer Systems* 34 (2014), 138–154.
- [189] ZEIGLER, B. P., PRAEHOFER, H., AND KIM, T. G. *Theory of modeling and simulation: integrating discrete event and continuous complex dynamic systems*. Academic press, 2000.

- [190] ZHAO, J., PAPAPETROU, P., ASKER, L., AND BOSTRÖM, H. Learning from heterogeneous temporal data in electronic health records. *Journal of biomedical informatics* 65 (2017), 105–119.
- [191] ZHENG, Y.-L., DING, X.-R., POON, C. C. Y., LO, B. P. L., ZHANG, H., ZHOU, X.-L., YANG, G.-Z., ZHAO, N., AND ZHANG, Y.-T. Unobtrusive sensing and wearable devices for health informatics. *Biomedical Engineering, IEEE Transactions on* 61, 5 (2014), 1538–1554.
- [192] ZHU, Y. Automatic detection of anomalies in blood glucose using a machine learning approach. *Journal of Communications and Networks* 13, 2 (2011), 125–131.
- [193] ZIVADINOV, R., WILLHEIM, K., JURJEVIC, A., SEPIC-GRAHOVAC, D., BUCUK, M., AND ZORZON, M. Prevalence of migraine in croatia: A population-based survey. *Headache: The Journal of Head and Face Pain* 41, 8 (2001), 805–812.

Appendix A

Modeling and simulation background

In this thesis, we create predictive models in order to anticipate a critical events. Models trained are obtained by supervised and semi-supervised algorithms. For further theoretical details refer to the original works and authors of the algorithms.

A.1 State-space models

State-space models represent the behavior of a signal through the relation with system's states and other variables [71]. In this thesis we restrict our model class as a Linear and time-invariant (LTI) state-space model of discrete sequences. State-space models can be written as shown in Equations A.1 and A.2, that represent the state equation and the observation equation respectively.

$$x_{k+1} = Ax_k + Bu_k + w_k \quad (\text{A.1})$$

$$y_k = Cx_k + Du_k + v_k \quad (\text{A.2})$$

State equation is easily resolvable by recursive substitution or more advanced algorithms as the N4SID algorithm proposed by Overshee and De Moor in [177].

State-space models define immeasurable states to describe difference equations that calculate the current and future outputs from current inputs, as shown in Equations A.1 and A.2. Both equations constitute a linear state-space representation of the dynamic behavior of y . u_k are, in our case, U exogenous inputs and y_k are the Y outputs at time k . A is the state transition matrix, B relates the states at time k (x_k) with the inputs, C is the state to output matrix, and D relates current outputs with current inputs; because the systems to be

modeled does not use to have an instantaneous response, D in the most of the cases equals zero. A , B , C and D are the matrices of the deterministic subsystem which describes the influence of the deterministic input u_k on the deterministic part y_k^d of the output $y_k = y_k^d + y_k^s$. v_k and w_k are immeasurable white noises that describe the influence of noise sequences on the stochastic output y_k^s .

The N4SID order nx (size of the square matrix A) has to be chosen as the best one in terms of fit. We assume coefficient matrices are time invariant despite the system might be retrained once in a while due to accuracy loss. Models are trained for stability, what requires that the eigenvalues of A lie inside the unit circle. Notice the reader that samples included in the past horizon are needed only to compute parameters of the state-space equations, but they will never be used for real time prediction.

Specifications to the implementation of the algorithm

There are many parameters in (Equations A.1 and A.2) that must be calculated. In our case, u_k are the 4 hemodynamic inputs and y_k is 1 output (symptomatic pain), both at time step k , and D equals zero. To calculate matrices A , B and C , the initial status x_{init} , it is necessary to use past data and future data to predict, for an specific order nx . This is a sized limited problem that has been solved through exhaustive search.

In the training process, the N4SID order nx (size of the square matrix A) and the number of samples from the past inputs and the output are chosen by the algorithm as the ones that achieve the best fit. To do this, the process runs in a triple loop looking for the best parameters, as shown in Algorithm 2. The inner loop chooses the order of the models. The one in the middle chooses the backward window to get information from past inputs, and the outer loop only fixes the prediction horizon.

In addition, a parallel study for feature selection has been performed. Algorithm 2 has been applied for the four hemodynamic inputs, but also with the combinations in triads of them; in total, we checked 5 sets of features. From these experiments, the features that better describe a migraine per patient are obtained. After training the models, 200 combinations have been checked to select the best one per future horizon and per migraine event, and per set of features.

At this point, the reader might have taken into account the problem that the definitions of Equations A.1 and A.2 imply for the predictive modeling due to the output $y[k]$ is only defined for the current moment of time k . It is easy to deduce that to predict the output value at time $k + h$, it will be necessary to know the

Data: $\{I = \text{Hemodynamic features}, O = \text{Pain to model}\}$
Variables: $\{\text{fw future windows}, \text{pw past windows}, \text{nx orders}\}$
Result: Best model generated for a migraine event

Initial best fit: $\text{prevFit} = -\infty$;

Exhaustive search:

```

for  $fw = 10 : 10 : 60$  do
  for  $pw = 0 : 5 : 100$  do
    Set options:  $\text{opts} = \text{n4sidOptions}(fw, pw)$ ;
    for  $nx = 1 : 10$  do
      Calculate the model:  $\text{stateSystem} = \text{n4sid}(\text{data}, nx, \text{opts})$ ;
      Calculate the fit:  $\text{fit} = \text{compare}(\text{data}, \text{stateSystem}, \text{futWin})$ ;
      if  $\text{fit} > \text{prevFit}$  then
        Save best model:
         $\text{prevFit} = \text{fit}$ ;
         $\text{bestPast}(fw) = pw$ ;
         $\text{bestOrder}(fw) = nx$ ;
      end
    end
  end
  Retrieve the best model:
   $\text{bestOptions} = \text{n4sidOptions}(fw, \text{bestPast}(fw))$ ;
   $\text{bestModel} = \text{n4sid}(\text{data}, \text{bestOrder}(fw), \text{bestOptions})$ ;
end

```

Algorithm 2: Space-state model training (N4SID algorithm).

status $x[k+h]$, which implies to know the inputs at $u[k+h-1]$. This means that it is necessary to predict the inputs too. What it is done to solve this issue is to push the output left h steps—bring the future values to the current moment—in order to let the models know that current inputs have to be related with future outputs.

A.2 Grammatical Evolutionary algorithms

Grammatical Evolution (GE) algorithms are a heuristic approach to the model discovery problem. GE reduce the model complexity and help on the automatic feature selection automatically by means of symbolic regressions, as opposed to the time costly manual feature selection of classic modeling methods [7].

GE [148] is a grammar-based form of Genetic Programming (GP) [135], used to generate programs in any language, where a Genetic Algorithm (GA) which,

based on biology, evolves a population formed by a set of individuals (the genotypes) as shown in Figure A.2. Each genotype is represented by a chromosome. A chromosome is an array of integer numbers (genes) that select production rules from a group expressed in a Backus Naur Form (BNF) grammar previously defined, which leads to phenotypes. A phenotype is a tree-shape structure which is evaluated in an iterative process. Phenotypes can represent (depending on the problem to solve): (i) mathematical expressions for time varying function modeling by applying Symbolic Regression (SR), or (ii) sets of features for feature selection and feature combination problems.

When a GE model symbolizes a time varying function, the model leads to a mathematical expression, which is a combination of features. This mathematical expression is an optimal solution obtained by applying SR. The features are operations, combinations and transformations over the input signals. They are automatically selected by the GE and this is an advantage over the classic methods such as state-space or autoregressive models. What the model actually provides when a GE model symbolizes a set/array of features to be selected is a weighing or a binary selection of the features.

Data: $\{I = \text{Input features}, O = \text{Function to model}\}$
Variables: $\{M \text{metrics}, P_{\text{crossover}}, P_{\text{mutation}}, G_{\text{max}}\}$
Result: Pareto front of best individuals in the last generation

Genesis: $S_g = S_{\text{genesis}};$

Optimization:

for $(\forall g \in G_{\text{max}})$ *or stabilization* **do**

Evaluate: $\{e_m\} = \text{evaluate}(I, O, S_g, \text{metric}_m);$

Rank and select: $\text{bestParents}_g = \max_{S_g} \{e_m \mid \forall m \in M\};$

Mix: $\text{offsprings}_g = \text{mix}(\text{bestParents}_g, P_{\text{crossover}});$

Mutate: $\text{mut}_g = \text{mutate}(\text{offsprings}_g, P_{\text{mutation}});$

New population (elitist): $S_g = \text{mut}_g \cup \text{replace}(\text{worstParents}_g);$

end

Algorithm 3: Genetic Evolutionary algorithm with elitists reinsertion (NSGA-II algorithm).

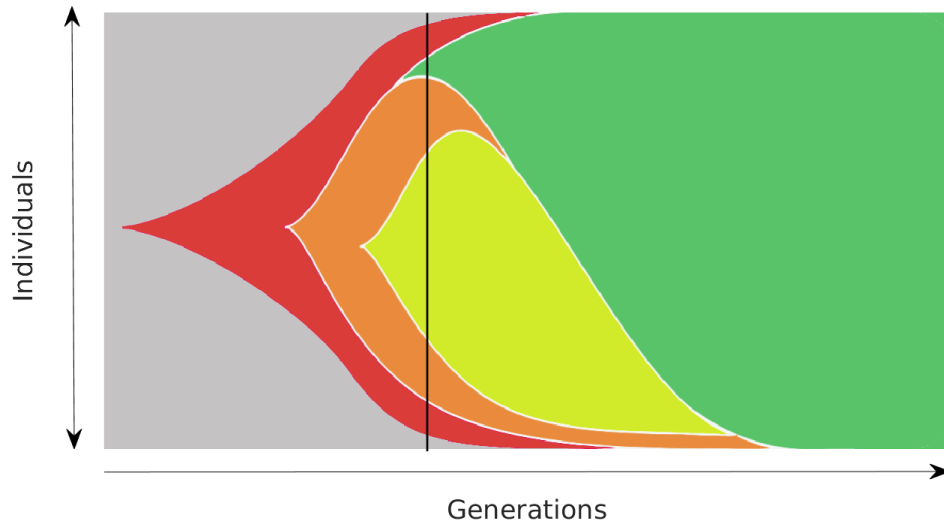


Figure A.1: In this diagram, colors represent equally-ranked solutions. The black vertical line shows how in a population can have different equally-ranked solutions at the same time. This diversity leads to new and better solutions. Author: *Gregory I. Lang, Lewis-Sigler Institute for Integrative Genomics and Department of Molecular Biology, Princeton University.*

The GE algorithm

Henceforth, this thesis is going to be focused on GE models representing time varying functions. As Algorithm 3 describes, at the beginning of the computation, in the *Genesis*, the GE algorithm creates a first random population formed by a set of individuals—the genotypes, also called solutions (Figure A.1). From now, iteratively for each generation (defined by epochs), till the last one, or stabilization of the metric values it is performed an optimization. It is said mono-objective optimization or multi-objective optimization if it is used one or more metrics respectively: (i) solutions are evaluated for each metric; (ii) the best solutions are chosen (*bestParents*) and (iii) these mix each other with the probability of crossover $P_{crossover}$ to create new ones (offsprings). (iv) The offsprings may mutate with a probability of mutation $P_{mutation}$, *i.e.* some genes change. (v) At the end of the epoch, it is created the next generation as the union of the offsprings and the best parents (up to complete the size of the population).

Figure A.1 shows how at the beginning (left side) an individual mutates (red) and along epochs this individual mix with others. Eventually, a new offspring—due to mutation or combination of chromosomes—improves the metric (orange). This happens again when yellow and green (the best) solutions appear. This keeps till the system tends to stabilization and all solutions are equally-ranked, *i.e.* different

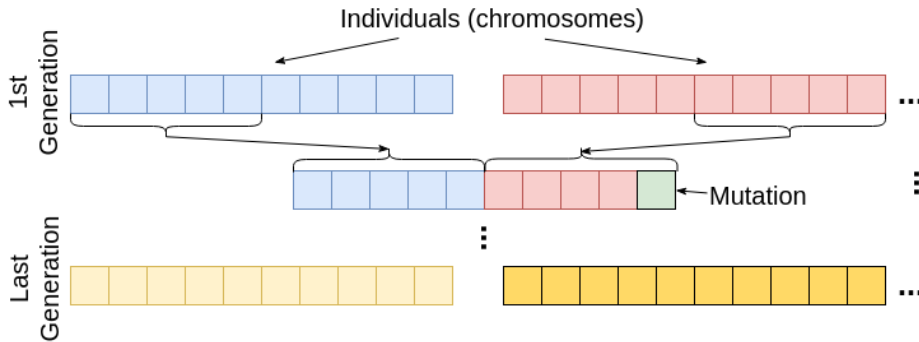


Figure A.2: Example of population. Mix and mutation of individuals (genotypes).

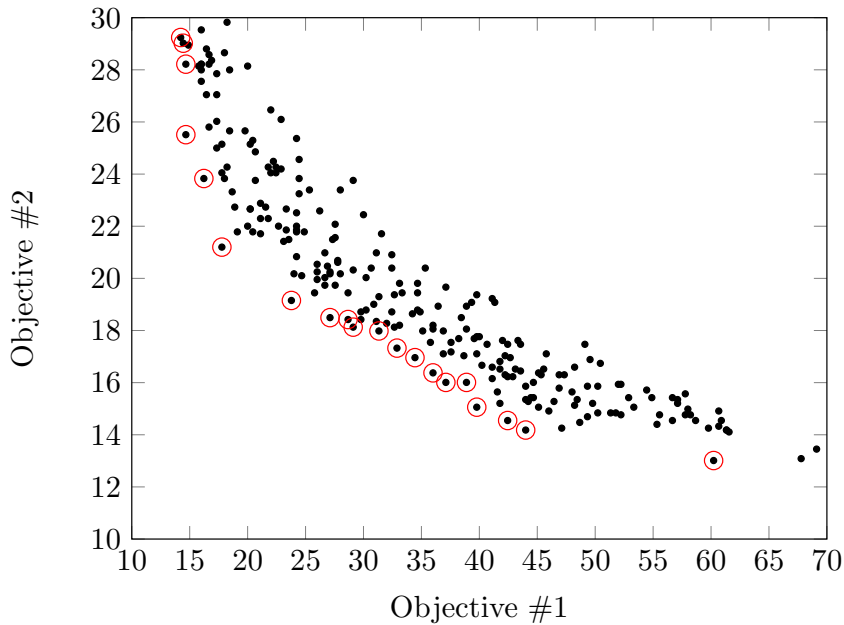


Figure A.3: Example of Pareto front to minimize a two-dimensional objective problem.

mathematical solutions but same quality (metric value).

The computation finishes when all generations G_{max} are evaluated or the metric values stabilize. At the end of the computation, the last generation represents the optimal solutions. If it used only one metric, there will be only one optimal solutions. On the contrary, if it has been used two or more metrics, a decision making problem appears. In this thesis, for multi-objective optimization, we use a multi-objective genetic algorithm inside GE: the Non-dominated Sorting Genetic Algorithm II (NSGA-II) [38]. The main difference between a simple GA and NSGA-II is that at the end, as opposed to a single best solution, a set of non-dominated solutions is obtained (traditionally named as Pareto front). Figure A.3 illustrates an example of a two-dimensional Pareto front. Black points in Figure A.3 are the last individuals or solutions of

the NSGA-II execution. Red circles surround those solutions that lead to the Pareto front. Refer to [33] for details on multi-objective GE implementation.

NSGA-II is an elitist approach, what means that a small part of the best candidates remain unchanged into the next generation—they remain as parents of the next generation. Each individual of the population plots a vector. The NSGA-II algorithm [38] will choose the best ones that are in the non-dominated Pareto front. An individual belongs to the Pareto front and is called non-dominated if any of its objectives can be improved (minimized in our case) without degrading the others. Figure A.3 represents a two-dimensional Pareto front. Black points in Figure A.3 are the last individuals or solutions of the GA problem (the best ones). Red circles surround those solutions that lead to the Pareto front.

For a better comprehension, a BNF grammar decodification example using real data is shown in Section 3.2.2.

Specifications to the implementation of the algorithm

This is the example of the chromosomes—understood as an array of Bytes potentially representing a part of a mathematical function—in Genetic Evolutionary algorithms. As an example, a brute force approach should evaluate, for a chromosome of 100 elements length and base 8-bit, $2^{8^{100}} = 6.67 \times 10^{240}$ different solutions. To avoid this, the Genetic Evolutionary algorithm evolves chromosomes of a population based on the probability of crossover and probability of mutation as explained in Section A.2.

Given that our GE algorithm is generating predictive models, we have designed a grammar that produces phenotypes forming symbolic mathematical expressions for the target model.

The decoding process of a chromosome into a mathematical function, is made by a set of production rules expressed in a Backus Naur Form (BNF) grammar [148]. A BNF grammar is represented by a set of parameters in the form $\{N, T, P, S\}$, where N is the set of non-terminals (coded symbols), T is the set of terminals (decoded expressions), P is the set of production rules to substitute the elements of N into T , and S is a non-terminal element of N used as starting symbol.

The grammar in Figure A.4 is the adaptation for 10 minutes of prediction horizon in BNF format. This is the version of the grammar that has been designed for the generation of predictive models using GE. This grammar is constituted by 12 production rules that are used to compose functions, variables and constants. The starting symbol, S , is the most general definition of an expression, the `<Model>` production rule. The others are non-terminals N production rules. In addition to

the basic mathematical pre-operators $\langle \text{PreOp} \rangle$, other basic mathematical functions are in $\langle \text{Fcn} \rangle$. In these functions it has been defined complex functions based on the Fast Fourier Transform (FFT) over the signals. In addition, the BNF considers to take past samples of the input variables as described in the Var production rule.

```

N = { <Model>, <Op>, <PreOp>, <Input>, <Fcn>, <Const>,
      <Base>, <Exponent>, <Sign>, <Var>, <Pw>, <Fw> }
T = { +, -, *, /, Exp, Sin, Cos, Log, Avg, Sum, Drv,
      Max, Min, AvgFFT, DrvFFT, MaxFFT, MinFFT,
      0, 1, 2, 3, ..., 130, TEMP, EDA, HR, SPO2, YP }
S = <Model>

P = {
  I   <Model>      ::= (<Model><Op><Model>)
                       | <PreOp>(<Model>)
                       | <Fcn>(k-<Pw>,k-10,<Input>)
                       | <Var> | <Const>

  II  <Op>         ::= +|-|*|/
  III <PreOp>     ::= Exp|Sin|Cos|Log
  IV  <Input>     ::= TEMP|EDA|HR|SPO2
  V   <Fcn>       ::= Avg|Sum|Drv|Max|Min
                       |AvgFFT|DrvFFT|MaxFFT|MinFFT
  VI  <Const>     ::= <Base>*Pow(10,<Sign><Exponent>)
  VII <Base>      ::= 1|2|3...99
  VIII <Exponent> ::= 1|2|3...9
  IX  <Sign>      ::= +|-
  X   <Var>       ::= TEMP(k-<Pw>)
                       | EDA(k-<Pw>)
                       | HR(k-<Pw>)
                       | SPO2(k-<Pw>)
                       | YP(k-<Fw>)

  XI  <Pw>        ::= 10|11|12...130
  XII <Fw>        ::= 0|1|2...9
}

```

Figure A.4: BNF grammar used for migraine prediction with 10 minutes in advance. The inputs of the grammar are the hemodynamic variables: surface skin temperature (TEMP), electrodermal activity (EDA), heart rate (HR) and oxygen saturation (SPO2), as well as the past predicted pain level (YP).

As seen in the figure, a predictive model (labeled as Model) is a combination of functions of five variables (terminals, T): the set of inputs (the hemodynamic variables: TEMP, EDA, HR and SPO2), and the predicted signal (YP). There exist a future horizon, labeled as Fw and a past horizon, labeled as Pw . To avoid autoregressive expressions using information of the real pain reported by the migraine sufferer YR, this variable has not been introduced in the grammar. The maximum past horizon is 120 minutes ($\langle \text{Pw} \rangle$), enough for our experiments. The output can also be based on the past predicted pain levels in a 10 minutes window backwards ($\langle \text{Fw} \rangle$).

One of the main benefits of GE is the capacity to create complex phenotypes from a genotype formed by integer values. In this regard, it is shown an example of the mapping process for the problem at hand using the grammar in



Figure A.5: Decoding process in GE that, starting from the genotype above, obtains the phenotype by mapping through the proposed grammar.

Figure A.4. Let's consider the genotype shown in the upper part of Figure A.5 as a candidate solution. This individual is a genotype that represents a predictive model. However, it has to be decoded in order to obtain the symbolic expression that will define such model.

The mapping or decodification process is performed by applying the modulus operation between the current codon value—group of bits mapped to an integer—and the number of options of each production rule corresponding to the non-terminal symbol being processed. Then, considering the example individual, the decoding process is performed as shown in Figure A.5. It begins decoding the starting symbol, <Model>. Here, the first gene of the genotype is selected, 220. Given that the production for <Model> (rule I) has 4 different options, the selected value is option 0 (220 MOD 4 = 0), which corresponds to <Model><Op><Model>. Notice that the first option of the rule is indexed with 0, the second with 1 and so on. Next, the second gene is read and its value, 106, is used to decode the following non-terminal symbol, which is <Model> with production rule I once more. Then, after the modulus operation (106 MOD 4 = 2), the first <Model> is replaced by

Table A.1: Parameters for the GE experiments.

Parameter	Value
Number of generations	150000
Population size	250
Probability of crossover	0.9
Probability of mutation	0.083
Chromosome length	100
Wrapping	No

$\langle \text{Fcn} \rangle(k - \langle \text{Pw} \rangle, k - 10, \langle \text{Input} \rangle)$. Now, the following non-terminal symbol to be decoded is $\langle \text{Fcn} \rangle$, whose production rule, V , has 9 different options. Then, the codon value, 91, is processed. Given that $91 \text{ MOD } 9 = 1$, the selected value is **Sum**. In the same way, the next codon, 123 for $\langle \text{Pw} \rangle$ is turned into 13 using rule **XI**, given that $123 \text{ MOD } 120 = 3$. The procedure continues until the full expression is (i) completely decoded (there might codons left over), or (ii) there are no codons to finish all the non-terminals, considering the chromosome invalid (sometimes, a wrapping process is used, but this is not the case presented). As a result, the genotype of the example returns the phenotype shown in the lower part of Figure A.5, which represents a predictive model with past horizons of 10 and 13 units of time.

The parameters used for training the models are shown in Table A.1. The probability of mutation is the inverse of the number of rules, 12 in our case. The length of the chromosomes is sufficient to avoid wrapping in more than the 80% of the cases (statistical observation of our experiments).

A.3 Time series analysis

Predictive modeling of time series requires from a deep understanding of our data. Time series analysis provides a statistical model which allows (i) to describe the evolution of an observed series (univariate), (ii) the relation between series (multivariate), and (iii) forecast the evolution. Eq. A.3 shows the basic decomposition of a time series z_t using an additive model of deterministic and non-deterministic components. This is the basis for the Wold's Theorem [185] on what the following explanations base.

$$z_t = T_t + C_t + S_t + I_t \tag{A.3}$$

z_t represents the observed values; T_t , the trend, which reflects the long-term

progression of the series; C_t the cyclical component, which shows repeated but non-periodic fluctuations; S_t the seasonal movements in a short period; and I_t random variations (noise) around the previous components. A different way to see a time series is the one shown in Eq. A.4.

Time series models are polynomial functions based on past data of the series. Time series models can be used to predict data by recursive iteration over past data, and the new predicted data—which, of course, increase the error. Classical methods try to find the known part of the time series in Eq. A.4, z_t^* ; while the methodology proposed by Box-Jenkins [20] looks for a_t , the unpredictable part, that does not depend on its past values.

$$z_t = f(z_{t-1}, z_{t-2}, \dots) + a_t = z_t^* + a_t \quad (\text{A.4})$$

Time series models are created for a particular time-finite series of a stochastic process. It is desired the series to be stationary; which means that mean μ and variance σ do not vary over time. Non-stationary series do not have a histogram, thus they do not have stable mean nor variance over time. Stationary series are common in physic variables, but not in social science or econometry. This hypothesis of stationery may be strict or weak. There are several transformations that convert non-stationary data into stationary, such as logarithmic, differentiation (see Figure A.6b) or more complex transformation like the Box-Cox transformation [21] to ensure the assumptions aforementioned for linear modeling.

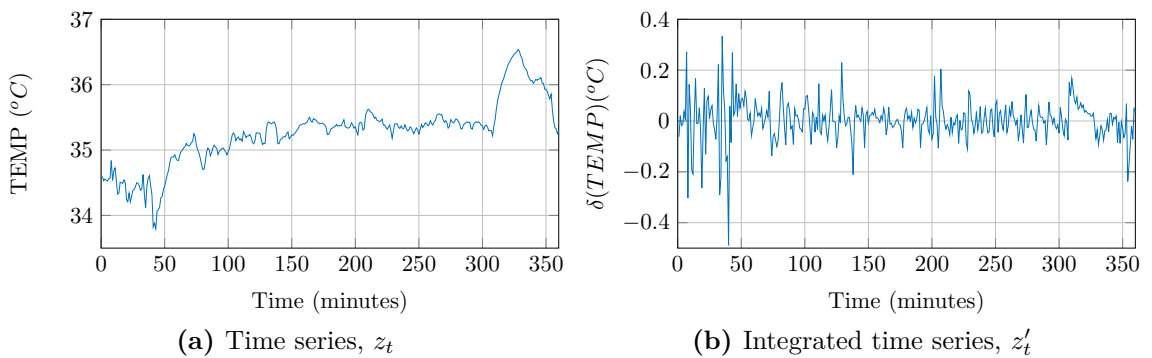


Figure A.6: Body skin temperature (TEMP) and its first derivative. See how the first derivative makes mean stationary. However, the variance does not seem to be stationary after the first derivative and it would require a higher order derivative.

Time series models can be linear (stationary or not), or not. As linear and stationary models there can be found Autoregressive models $AR(p)$, Moving-Average models $MA(q)$, or mixed models $ARMA(p, q)$. As linear and

non-stationary models, there can be found the integrated processes (Autoregressive Integrated Moving-Average, $ARIMA(p, d, q)$, which become stationary after d derivatives.

If it is desired to study the relation of a time series—response, lagged dependent variable y_t —and lagged independent variables—exogenous—it is used an ARIMAX model (in the most general case: autoregressive, integrated and moving average; other forms are ARX, MAX and ARMAX). The Autoregressive Moving-Average model including exogenous covariates, $ARMAX(p, q)$, extends $ARMA(p, q)$ model by including the linear effect that one or more exogenous stationary response series y_t . On the other hand, there can be found multivariate models as VARMAX (Vector Autoregressive Moving-Average model with exogenous variables), where the output variables—also called dependent, response, or endogenous variables—of interest can be influenced by other input variables—are called independent, input, predictor, regressor, or exogenous variables.

This brief background is focused on ARIMAX linear models—in the most general form—which are used to predict critical events based on external time series, and they are also used to repair a signal based on past data from other signals. Without mathematical details lets explain how to find the model that represents a time series.

Time series model identification

Looking at a time series to identify the type of model that better represents the series is not an evident task. The Autocorrelation Function (ACF) and the Partial Autocorrelation Function (PACF) are two tools that help on the identification of pure $AR(p)$ or pure $MA(q)$ models. The ACF represents delay k in the series with its autocorrelation, $\rho_k = \frac{cov(z_t, z_{t-k})}{\sqrt{var(z_t) - var(z_{t-k})}}$. On the other hand the PACF measures the relation between z_t and z_{t-k} removing the effect of $z_{t-1}, z_{t-2}, \dots, z_{t-k+1}$.

The number of not null coefficients of the PACF determines the order of an autoregressive model. Table A.2 represents the main characteristics to identify a model by looking at their autocorrelation coefficients. An example is shown in Figure A.7. Figures A.7a and A.7b represent the ACF and PACF of the time series z_t depicted in Figure A.6a. It can be seen that the series is not stationary, at least its mean value. A basic transformation based on its first derivative (Figure A.6b) makes it more likely a stationary series; the mean is stable while there is still doubts about the stationarity of the variance. Looking at the ACF of z_t and z'_t , we can conclude that z'_t is a Moving-Average process, so the time series TEMP might be a integrative MA process. There might have doubts about the order q ,

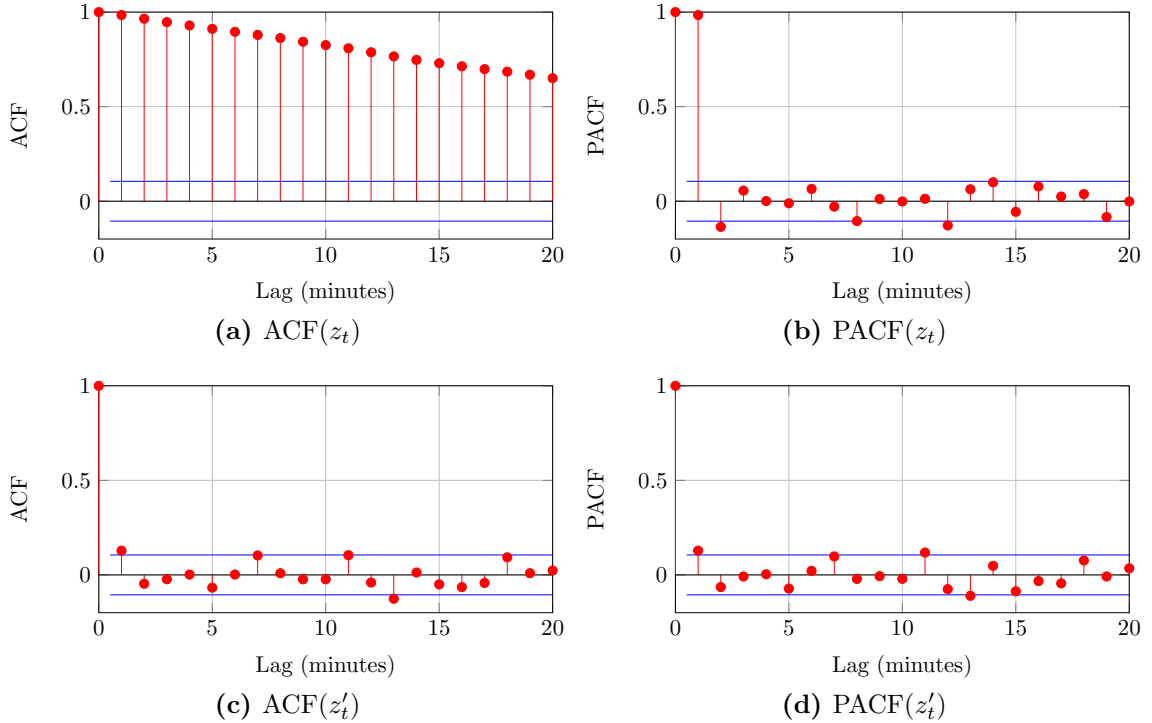


Figure A.7: Simple and partial autocorrelation functions for the TEMP time series and its first derivative. The lag represents the delays z_{t-k} . Observe that for $k = 0$, there is no delay and then the autocorrelation is 1. The blue lines represent the 95% confidence interval assuming z_t is an MA process in (a) and (c), or an AR process in (b) and (d).

but the derivative has changed considerably the shape of the ACF in Figure A.7a and $ACF(z'_t)$ decreases smoothly as expected from Table A.2. The PACF remains the same, and they have significant coefficients, so it seems z_t is a mixed process, ARIMA.

The identification of a mixed process is not an easy task if it desired to be strict with the characteristics described in Table A.2. Maybe because it has an important seasonal series added, the model that represents a process can lead to an excessive number of parameters (large polynomial equations). Seasonal data are considerable cyclical and residual variations around the trend, whose values are grow and diminish with the trend. These are the SARMA models,

Table A.2: Main characteristics to identify pure AR and MA time series models.

	ACF	PACF
$MA(q)$	Cuts-off after lag q	Decreases as an exponential or sinusoidal
$AR(p)$	Decreases as an exponential or sinusoidal	Cuts-off after lag p

multiplicative models that understand a time series as shown in Eq. A.5. They can be seen as more complex models which reduce the number of parameters these $ARIMA(p, d, q)x(P, Q)$ [20]. To deal with them, take logarithms in both sides of Eq. A.5 to transform multiplications into sums, and then the time series is analyzed as aforementioned.

$$z_t = (T_t * C_t * I_t) * S_t \quad (\text{A.5})$$

The study of time series models is a complex task. In these lines there have been just mentioned the basics. To know more, please refer to the bibliography cited.

Specifications to the implementation of the algorithm

If a signal suffer from data loss or disconnections, it might be recovered temporary through information extracted from the remaining variables. In our case, hemodynamic variables can be seen as autoregressive, moving-average, mixed, or multiplicative processes. With this premise, it makes sense to go further to find a univariate model where relation of past data of unaltered signals, explain the behavior of the damaged one. To solve this problem the algorithm has been computed using the Econometrics Toolbox of the MATLAB software [85].

To identify the nature of the process behind these models, it has been conducted an exhaustive search like the one presented for state-space algorithms. Models have been selected to maximize the fit to the symptomatic pain curve. In Algorithm 4 it is necessary to preprocess the exogenous inputs u_t by testing stationarity and differencing if needed. Otherwise, if any non-stationary exogenous variable enters the model, the quality of the system decreases.

In Algorithm 4, due to there is no possibility to force the model to be trained in prediction, *i.e.* compute future outputs, it is necessary to bring forward—to current time $k = 0$ —the output of the system. This lets the model know that it is desired to correlate current and past input data with future output data.

It is desired to set $p = 0$ to avoid AR models. AR models depends heavily on the initial state of the basal system, *i.e.* it is necessary to know and control the basal state of the output (the hemodynamic variable), which seems to be a very complicated task. In addition, $p = 0$ reduces the computation time of the computation of the model. D indicates the order of differentiation of the hemodynamic variables, if needed.

Data: $\{I = \text{Unaltered hemodynamic variables},$
 $O = \text{Damaged signal to model}\}$

Variables: $\{p \text{ past outputs}, r \text{ past inputs}, d \text{ integrative window}\}$

Result: Best model generated for a hemodynamic variable

Initial best fit: $\text{prevFit} = -\infty$;

Exhaustive search:

```

for  $p = 1 : 20$  do
  for  $r = 0 : 20$  do
    for  $d = 0 : 20$  do
      for  $D = 0 : 2$  do
        Set options:  $\text{opts} = \text{arimaxOptions}(p, r, d, D)$ ;
        Calculate the model:  $\text{system} = \text{arimax}(\text{data}, \text{opts})$ ;
        Calculate the fit:  $\text{fit} = \text{compare}(\text{data}, \text{system})$ ;
        if  $\text{fit} > \text{prevFit}$  then
          Save best model:
           $\text{prevFit} = \text{fit}$ ;
           $\text{bestOptions} = \text{opts}$ ;
        end
      end
    end
  end
end

```

Retrieve the best model:

$\text{bestModel} = \text{arimax}(\text{data}, \text{bestOptions})$;

Algorithm 4: Time series model training (ARIMAX algorithm).

A.4 DEVS simulation

DEVS modeling and simulation formalism allows to simulate models under a unified modeling and simulation theory in real time, soft-real time and virtual time.

DEVS is a modular and hierarchical modeling formalism, with all of the advantages and uses of simulation systems, such as: completeness, verifiability, extensibility, and maintainability and allows execution of Monte Carlo simulations, parallel simulation using threads or distributed using webs [183], as an example. It has been used the xDEVS open source JAVA library with the aim of making a future implementation on a hardware device. DEVS is a general formalism for discrete event system modeling based on a mathematical Set Theory [189]. Over the last four decades it has been used to implement a formally described system using an existing software/hardware library in

multiple languages (e.g. Lisp, Scheme, C++, JAVA, Python, *etc.*). There are two types of models in DEVS: atomic and coupled. An atomic model is irreducible and it specifies the behavior for any modeled entity: processes an input event based on its state and condition, and generates an output event and changes its state. DEVS formally represents an atomic model by three sets: input (X), output (Y) and state (S), and five functions: time advance (ta), external transition (δ_{ext}), internal transition (δ_{int}), confluent (δ_{con}) and output (λ). Formally, it is expressed as follows:

$$A = \langle I, O, X, S, Y, \lambda, \delta_{int}, \delta_{ext}, \delta_{con}, ta \rangle \quad (\text{A.6})$$

where:

- I is the set of input ports.
- O is the set of output ports.
- X is the set of inputs described in terms of pairs port-value: $\{p, v\}$.
- S is the state space. It includes the current state of the atomic model and also two special parameters called σ and *phase*. σ is the time until the next event generation, and the *phase* is a description of the current state (usually in natural language).
- Y is the set of outputs, also described in terms of pairs port-value: $\{p, v\}$.
- $\lambda : S \rightarrow Y$ is the output function. When the time elapsed since the last output function is equal to σ , then λ is automatically executed.
- $\delta_{int} : S \rightarrow S$ is the internal transition function. It is used to change the state S , *phase* and σ , and it is executed right after the output function (λ).
- $\delta_{ext} : Q \cdot X^b \rightarrow S$ is the external transition function. It is automatically executed when an external event arrives to one of the input ports, changing the current state if needed.
 - $Q = (s, e), s \in S, 0 \leq e \leq ta(s)$ is the total state set, where e is the time elapsed since the last transition.
 - X^b is a bag of elements of X .
- $\delta_{con} : Q \cdot X^b \rightarrow S$ is the confluent function, subject to $\delta_{con}(s, \emptyset) = \delta_{int}(s)$. This transition is selected if δ_{ext} and δ_{int} must be executed at the same instant.
- $ta(s) : S \rightarrow \mathfrak{R}_0^+ \cup \infty$ is the time advance function.

A coupled model aggregates and interconnects two or more atomic or coupled models. And it is formally described as:

$$M = \langle I, O, X, Y, C_i, EIC, EOC, IC \rangle \quad (\text{A.7})$$

where:

- I, O are the set of external (not coupled) input and output ports.
- X is the set of external input events.
- Y is the set of output events.
- C_i is a set of DEVS component models (atomic or coupled). Note that C_i makes this definition recursive.
- EIC is the external input coupling relation.
- EOC is the external output coupling relation.
- IC is the internal coupling relation.

Due to the definition in Eq. A.7, a coupled model can itself be a part of a component in a larger coupled model system giving rise to a hierarchical DEVS model construction.

In this thesis it has been used DEVS for model verification. It will be shown the simulation of the an advanced monitoring prediction system from a top-down view.

Appendix B

Experimental set-up. On-body channel transmission modeling

In the offline phase, and still in the clinic, the study of the on-body transmission parameters is carried out. Figure B.2 corresponds to the initial phase prior to the experimental procedure described in Figure 3.6. However, in this implementation, the offline phase of the study lasts between 2 weeks and one month. During this time the patient's migraines are recorded and the prediction models are developed. In the online phase, patients are continuously monitored and a runtime prediction of the migraine is performed. As it is shown later, each of these two phases has a different impact on the global consumption of the application.

During the training or offline phase, in the hospital, the sensing nodes are placed on patients' body describing a star topology. The coordinator (a smartphone) is placed in the waist (just over the navel), the sensor node S1 in the right arm (link L1) and the sensor node S2 in the right knee (link L2). The sensors are wirelessly connected to the coordinator. Sensor node S1 senses the ECG signal on the chest using one derivation, EDA in the arm using two electrodes, and TEMP near the armpit using an NTC thermistor. Sensor node S2 senses the SpO2 in the capillarity zone near the groin using the 8000R SpO2 sensor and the OEM-III module.

On-body channel transmission parameters

Still in the hospital, the on-body channel transmission parameters for each patient are calculated. These parameters are required to feed the transmission models and apply the radio techniques in [174] to reduce the energy consumption.

To develop the models some physiological measurements of each patient are required, such as: the arm circumference, body fat mass, bone mass, muscle mass, *etc.* [175]. Based on the research group previous work, a proactive

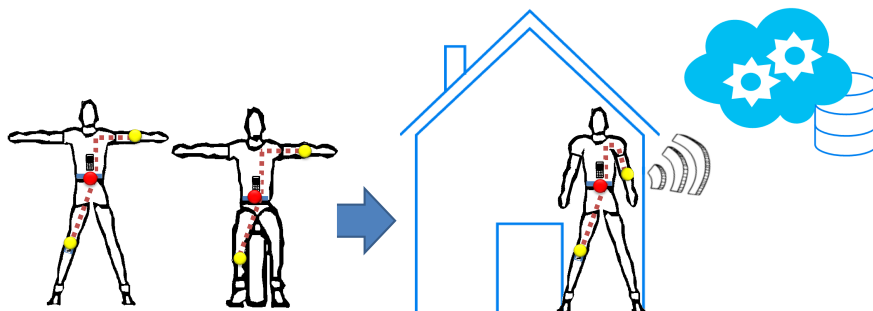


Figure B.1: Sensor placement and monitorization. Two sensors are wirelessly connected in a star topology to a coordinator (a smartphone in the waist). The sensor node S1, placed in the right arm, communicates with the coordinator via the L1 link. The sensor node S2, in the knee, connects via the L2 link. On left side of the figure, the patient performs a sequence of movements to calculate the on-body channel transmission parameters. On right side, data are transmitted wirelessly to the Data Center.

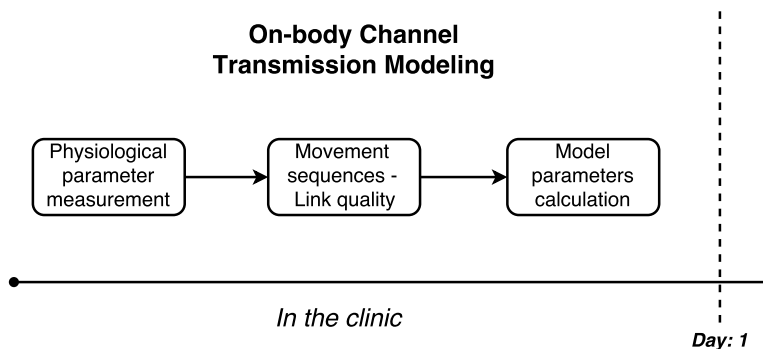


Figure B.2: On-body channel transmission scheme. At the beginning of the study the parameters of the models for the on-body channel transmission are calculated in the hospital. After that, the study continues as stated in Figure 3.6

technique to control the transmission power in the communication between the nodes and the coordinator is carried out [173]. This proactive technique adapts dynamically the transmission power based on the variations in the RSSI index. These models are patient-dependent and use an ANFIS Link Quality Estimator (A-LQE) model—based on ANFIS networks—to estimate the quality of the radio links. RSSI levels and minimum transmission power affordable are related through values stored in a Look Up Table (LUT). To calculate the parameters of the models, the patients have to perform a sequence of movements standing up and sitting down [175].

The exercises are a sequence of movements that simultaneously combine different positions P_i , $i = 1, 2, \dots, N$, of arm and knee. The number of exercises $N = 5$ for sequence 1 of movements, and $N = 4$ for sequence 2. The positions of

arm and knee are named as: L1P1+L2P4, L1/P2+L2/P3, L1/P3+L2/P1, L1/P4+L2/P2 and L1P5+L2P1. For each link, the size of the LUT in the corresponding node matches the number of exercises considered in that link. Each one of the positions is described below:

- Sequence 1: the subject sits on a chair and performs five movements of the arms (Link 1, L1): (i) hands on thighs, denoted as L1/P1; (ii) arms crossed, L1/P2; (iii) arms extended forward, L1/P3; (iv) arms extended up, L1/P4; and (v) arms extended to both sides, L1/P5.
- Sequence 2: the subject sits on a chair performs four movements of the legs (Link 2, L2): (i) leg in 90° angle with the body, L2/P1; (ii) left leg crossed over the right knee, L2/P2; (iii) right leg crossed over left knee, L2/P3; and (iv) leg extended forward, L2/P4.

Appendix C

Sizing of HPC and Cloud clusters

The HPC and Cloud clusters of this thesis can be either understood as either a centralized Data Centers that performs all the computation needed, but also as federation of Data Centers located in Europe. In order to compute the power consumption and performance of these two clusters, the following methodology has been followed:

1. Profiling and characterization of the GPML, training and validation stages in the Intel Xeon servers of the HPC cluster, obtaining the power consumption of each task for one instance of each task and when the server is fully utilized.
2. Profiling of GPML and prediction in the SandyBridge servers and the coordinators nodes, obtaining power and performance.
3. Consolidation analysis at maximum frequency for the virtualized cluster, to discover the maximum amount of instances that can be run in one server until utilization reached 100% without degrading performance. In this sense, we have found that the computational burden of GPML is of 260 ms, allowing up to 230 instances to run per core on the same VM without degrading performance. As for prediction, we can launch simultaneously 250 instances per VM, reaching a per-core utilization of 100% and not degrading performance.
4. Data Center sizing, *i.e.* obtaining the amount of servers of each type for both the HPC and the Cloud cluster.
5. Generation of the incoming workloads, that consists on new models to be trained in the HPC cluster, as well as model re-training. The output of the HPC cluster, *i.e.* the trained models, are the input to the Cloud cluster.

Regarding Data Center sizing, we need to compute models for $P = 1,393,649$ patients (2% of the migraine sufferers in Europe). To this end, the HPC cluster is composed of with $S = 2275$ Intel Xeon servers, with 4 cores each, so that the total number of cores $C = 9100$ is able to tackle modeling. Models need to be re-trained each $R_{re-train} = 30$ days per patient.

As for the Cloud cluster, a similar computation needs to be performed. Again, Data Center has been designed for an 80% utilization. In this sense, considering that we run one VM per core. There is only one final consideration that must be taken into account, and is that even though all models are predicted, the data preprocessing is not run every time, only when data are lost. As it has been experimentally calculated the probability loss of each sensor separately, to compute the workload it is needed to obtain the number of GPML instances that need to be run, on average, for each model. Once the probability is obtained, it can used the same formula than for the HPC cluster.

Appendix D

Costs of the migraine disease in Europe

The migraine affects approximately 15% population in Europe. This value ranges from 8.8% in Portugal [120] to 24.6% in Switzerland [117]. Each migraine patient leads to costs of €1222 per year in Europe, and according to the study in [105], Linde *et al.* distribute these costs in the way shown in Table D.1. Reducing indirect costs would lead to a huge amount of savings for the National Health Services and private health companies.

The migraine is a stable chronic disease whose affection rates in population vary very slow in time; thus, let's suppose the percentages of migraineurs in Table D.2 are still real (despite some of them are studies from more than 20 years ago). Table D.2 shows the number of migraine sufferers in different European countries. In this thesis it has been considered the European population in 2014 [51] (excluding Andorra, Bosnia, Kosovo, Monaco, European area of Russia, San Marino and Armenia).

Second-to-last column in Table D.2 show the amount of patients in the migraine prediction study. It has been designed a Data Center able to manage this population. This population is 2% of the migraine population.

In this thesis it has been proposed a methodology in order to change between

Table D.1: Average direct and indirect costs of migraine disease in Europe per patient per year [105].

Indirect (93%)		Direct* (7%)				
Productivity (€)	Absenteeism (€)	Outpatient care (€)	Diagnostic investigations (€)	Hospitalizations (€)	Acute medications (€)	Prophylactics (€)
(2/3)	(1/3)					
758	379	30	19	16	16	5

*Acute medications and prophylactics costs are not reduced by predictions of migraines. Patients will continue with the medical treatment.

Table D.2: Migraine sufferers in Europe for the target population of 2% of migraineurs for this research study. Total savings in direct and indirect costs due to migraine prediction with 76% of average rate of prediction success.

Country	Population [51]	Age range	Migraineurs (%) [Reference]	Total migraineurs	Target population	Savings (M€)
Turkey	76,667,864	≥14	19.9 [27]	15,256,905	305,138	278.5
UK	64,308,261	16-65	14.3 [162]	9,196,081	184,150	168.0
Germany	80,767,463	≥20	11.4 [134]	9,207,491	183,922	167.8
Italy	60,782,668	43±13	15.0 [163]	9,117,400	182,348	166.4
France	65,835,579	≥18	11.2 [100]	7,373,585	147,472	134.6
Spain	46,512,199	18-65	12.6 [115]	5,860,537	117,211	107.0
Netherlands	16,829,289	20-65	23.2 [101]	3,904,395	78,088	71.3
Switzerland	8,139,631	29-30	24.6 [117]	2,002,349	40,047	36.5
Sweden	9,644,864	18-74	13.2 [34]	1,273,122	25,462	23.2
Norway	5,107,970	18-65	23.0 [156]	1,174,833	23,497	21.4
Denmark	5,627,235	12-41	19.1 [147]	1,074,802	21,496	19.6
Hungary	9,877,365	15-80	9.6 [15]	948,227	18,965	17.3
Portugal	10,427,301	-	8.8 [120]	917,602	18,352	16.7
Austria	8,506,889	≥15	10.2 [98]	867,703	17,354	15.8
Croatia	4,246,809	15-65	19.0 [193]	806,894	16,138	14.7
Georgia	4,490,498	≥16	15.6 [89]	700,518	14,010	12.8
Total	477,771,885		15.7±5.2	69,682,444	1,393,649	1271.8

Table D.3: Availability of models depending on features and sensors' status. Total of detected events on average to compute accuracy of the prediction system.

	Model (M_k)	Availability (%)	Alarms = TPR (%)	Time of usage (%)	Detected events (%)
Patient A	EDA-HR-SpO2	61.9	67.0	61.9	41.4
	TEMP-EDA-SpO2	61.9	60.0	23.6	14.2
	TEMP-EDA-HR-SpO2	58.0	53.0	8.4	4.5
	TEMP-EDA-HR	82.5	53.0	5.0	2.7
	TEMP-HR-SpO2	61.9	47.0	0.7	0.3
	Total				63.1
Patient B	TEMP-EDA-SpO2	61.9	90.0	61.9	55.7
	TEMP-EDA-HR	82.5	90.0	31.5	28.3
	TEMP-EDA-HR-SpO2	58.0	90.0	3.9	3.5
	EDA-HR-SpO2	61.9	60.0	1.7	1.0
	TEMP-HR-SpO2	61.9	50.0	0.7	0.3
	Total				88.9
Average					76.0

prediction models according to the availability of sensors. The probability of failure for each sensor is shown in Table 3.7. With these probabilities, and the prediction results obtained, the percentage of avoided migraines has been calculated. This value (76.0%) appears as an average result in Table D.3 and indicates the saving costs per patient per year.

Table D.3 shows the results for two patients. Availability (%) means the probability of usage of each kind of model depending on the combination of features (sensors). The availability has been calculated with probabilities of failure in Table 3.7. Eq. D.1 shows the most general expression to calculate the probability of usage of a kind of model M_k , $k = 1, 2, \dots, 5$.

$$\begin{aligned}
 P_U(M_k) &= P_A(f_1) * P_A(f_i) * \dots * P_A(f_N) = \\
 &(1 - P_F(f_1)) * (1 - P_F(f_i)) * \dots * (1 - P_F(f_N))
 \end{aligned}
 \tag{D.1}$$

where:

$P_U(M_k)$ is the probability of usage or availability of model M_k . $P_A(f_i)$ is the probability of availability of sensor f_i , and $P_F(f_i)$ is the probability of failure of sensor f_i in Table 3.7, $i = 1, 2, \dots, 4$.

The *Alarms* column in Table D.3 represents the True Positive Rate (TPR) of each model. These models do not report any False Positive (FP) event, thus, TPR equals the percentage of real migraines detected by the system. The more accurate a model is the more time it is used. Time of usage $T_U(M_{k|k>1})$ (%) of model $M_{k|k>1}$ in Table D.3 is calculated as shown in Eq. D.2:

$$T_U(M_{k|k>1}) = (1 - T_U(M_1) - \dots - T_U(M_{k-1})) * P_U(M_k) \quad (\text{D.2})$$

and:

$$T_U(M_1) = P_U(M_1) \quad (\text{D.3})$$

As an example, for both of the patients considered in Chapter 4, the models of features combinations TEMP-HR-SpO2 only detect 0.3% of all the migraines of each patient. Models detect 63.1% of migraines of Patient A and 88.9% of migraines of Patient B when hierarchical change of models is applied taking into account the probability of failure of each sensor. On average it can be extrapolated that this system detects 76.0% of migraines without false alarms.

76.0% are the savings applied for each row in Table D.2. Total savings of migraine costs over the migraineurs to whom one Data Center can provide its prediction benefits leads to € 1271.8 million.

Appendix E

Ethical consent

In the following it is shown a copy of the ethical consent taken in this research study. Despite it is a public document, the access to it is not easy. The following lines translate the document:

"The **Clinical Research Ethics Committee of the Hospital Universitario de La Princesa** in its meeting of 07-08-2014 (minutes 15/14) evaluated the following research project:

TITLE: WiMigraine: Non-invasive outpatient monitoring of biometric, biophysical and electroencephalographic variables as a method for the prediction of a crisis of migraine (Monitorización no invasiva de variables biométricas, biofísicas y electrofisiológicas en pacientes con migraña como método para la predicción de una crisis). **Number of registration:** PI-771 **Main researcher:** Dr. José Vivancos (Neurology Service)

Decision taken: Approval (7-08-14)

This Clinical Research Ethics Committee considers that both the research project and the Patient Information Sheet are **ethically and methodologically acceptable**. Likewise, it considers that the researchers are competent to carry out this project that is framed within the priority research lines of the Hospital Universitario de La Princesa."

COMITÉ ÉTICO DE INVESTIGACIÓN CLÍNICA

Madrid, a 08 de agosto de 2014

El **Comité Ético de Investigación Clínica del Hospital Universitario de la Princesa** en su reunión del día 07-08-2014 (acta 15/14) evaluó el siguiente proyecto de investigación:

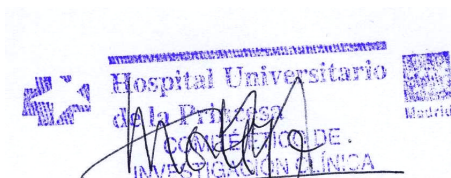
TITULO: WiMigraine: Non-invasive outpatient monitoring of biometric, biophysical and electroencephalographic variables as a method for the prediction of a crisis of migraine (Monitorización no invasiva de variables biométricas, biofísicas y electrofisiológicas en pacientes con migraña como método para la predicción de una crisis).

Nº de Registro: PI-771

Investigador principal: Dr. José Vivancos (Servicio Neurología)

Decisión tomada: Aprobación (7-08-14)

Este Comité Ético de Investigación Clínica considera que tanto el proyecto de investigación como la Hoja de información al paciente son **ética y metodológicamente aceptables**. Asimismo, considera que los investigadores son competentes para llevar a cabo este proyecto que está enmarcado dentro de las líneas de investigación prioritarias del Hospital Universitario de La Princesa.



Fdo: Dra. M^a de Mar Ortega Gómez
Secretaria del C.E.I.C.

