



HAL
open science

Data analytic frameworks for patients monitoring and assessment in connected healthcare applications

Ghina Saad

► **To cite this version:**

Ghina Saad. Data analytic frameworks for patients monitoring and assessment in connected healthcare applications. Human health and pathology. Université de Haute Alsace - Mulhouse, 2022. English. NNT : 2022MULH5986 . tel-04117588

HAL Id: tel-04117588

<https://theses.hal.science/tel-04117588>

Submitted on 5 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Université de Haute Alsace

Ecole Doctorale (269) Mathématiques, sciences de l'information et de l'ingénieur

IRIMAS : Institut de Recherche en Informatique, Mathématiques, Automatique et Signal

Spécialité : Informatique

THÈSE DE DOCTORAT

Pour obtenir le grade de : Docteur de l'Université de Haute-Alsace

Par

Ghina Hassan SAAD

Data Analytic Frameworks for Patients Monitoring and Assessment in Connected Healthcare Applications

Soutenu le 10/ 11/ 2022 devant le jury :

M. Ye-Qiong SONG

Professeur des universités à l'Université de Lorraine.

M. Abdallah MAKHOUL

Professeur des universités à l'Université de Franche-Comté.

Mme. Laetitia JOURDAN

Professeure des universités à l'Université de Lille.

M. Jaafar GABER

McF HDR à l'Université de Technologie de Belfort-Montbéliard.

M. Abdelhafid ABOUAISSA

Professeur des universités à l'Université de Haute-Alsace.

M. Lhassane IDOUMGHAR

Professeur des universités à l'Université de Haute-Alsace.

M. Hassan HARB

McF à l'American University of Culture and Education.

Mme. Nour CHARARA

McF à l'American University of Culture and Education

Rapporteur

Rapporteur

Examineur

Examineur

Directeur de thèse

Directeur de thèse

Co-encadrant

Co-encadrant

Abstract

Today, diseases and illnesses are becoming the most dangerous enemy to humans. The number of patients is increasing day after day accompanied with the emergence of new types of viruses and diseases. Recently, wireless body sensor network (WBSN) has been considered as an efficient technology for real-time health-monitoring applications. It provides a low cost solution for hospitals, performs a relief for staff and allows nurses and doctors to remotely track patients. However, the huge amount of data collected by sensors produce two major challenges for WBSN: the quickly depletion of the available sensor energy and the complex decision making by the doctor. Subsequently, these challenges are highly depended on the enormous amount of redundant collected and transmitted data in such network. Therefore, data reduction and prediction techniques have an important effect in preserving the sensor energy and prolonging the network lifetime. In addition, most hospitals suffer from the deficiency of qualified staff needed to continuously monitor patients and act when an urgent situation is detected. Thus, designing a nurse-patients scheduling is becoming essential to organise and balance the workload of the medical staff. In this thesis, we consider a WBSN architecture consists of a set of sensors, where each sensor monitors a specific vital sign during a period of time and then sends the collected data to a coordinator which, in turns, forwards them to the sink. Then, we propose data reduction, classification, management and nurse scheduling techniques that aim to overcome the mentioned challenges. Fundamentally, the proposed techniques work on sensor and sink levels. At sensor level, we propose an emergency detection and adaptive sensing frequency techniques that aims respectively to reduce the collected and transmitted redundant data according to the patient situation. At sink level, we propose data analysis and management techniques: First, a framework based on distributed systems for ingestion, processing, storage, prediction and visualisation, using several hadoop tools. Where, a prediction technique is implemented based on Prophet method that predicts the future patient's situation in order to provide the necessary treatment. Second, a patient classification model that aims to classify the patients according to their vital signs compared to the patient archive, using the Dynamic Time Warping algorithm. Finally, an efficient nurse-patients scheduling technique which consist of two steps. In first step, we proposed three distribution mechanisms that aims to allocate for each nurse the group of patients who should follow. Where, two clustering-based mechanisms and one is based on a hybrid of genetic algorithm and particle swarm optimization. In second step, a scheduling algorithms based on the patient's priority is implemented, that aims to determine the optimal order of patients for examination. We applied our simulation on real health sensor data, while the obtained results show its accuracy in terms of extending the network life, processing storage speed, regenerating of missing data, and confirm that the proposed scheduling approach provides the best possible organised time for the nurses routing to patients appointments.

KEYWORDS: *Wireless Body Sensor Network, Data Reduction, Patient Classification, Patient-Nurse Scheduling, Emergency Detection, Energy Saving.*

Shorts Contents

<i>Abstract</i>	I
<i>Contents</i>	III
<i>List of Figures</i>	VIII
<i>List of Tables</i>	XI
<i>List of Abbreviations</i>	XIII
<i>Notations</i>	XV
<i>Dedication</i>	XVIII
<i>Acknowledgements</i>	XX
Intorduction	XXI
1 General Introduction	XXI
2 Main Contributions of this Thesis	XXII
3 Thesis Structure	XXIII
1 An Overview About Wireless Body Sensor Networks	1
1.1 Introduction	1
1.2 Various Types of Biomedical Sensors	1
1.3 WBSN in Internet of Things (IoT)	2
1.4 WBSN 3-Tier Architecture	3
1.5 Intra-WBSN Communication Types	3
1.6 WBSN Challenges	3
1.7 Patient Monitoring and Risk Assessment: A Background	5
1.8 Conclusion	21
2 An Adaptive Intra-WBSN Communication Approach	23
2.1 Introduction	23
2.2 Comparative study of recent advances in data transmission in IoT	24
2.3 Optimised Intra-WBSN Communication Approach	28
2.4 Simulation Results	31
2.5 Conclusion	36
3 P2D: An Efficient Patient-to-Doctor Framework for Real-Time Health Monitoring and Decision Making	39
3.1 Introduction	39

3.2	WBSN Framework	40
3.3	Patient Monitoring Model	41
3.4	Medical Team Decision-Making Model	47
3.5	Simulation Results	53
3.6	Conclusion	63
4	A Sensing-Based Patient Classification Framework for Efficient Patient-Nurse Scheduling	65
4.1	Introduction	65
4.2	Patient Classification Phase	66
4.3	Nurse Scheduling Phase	68
4.4	Research Results	76
4.5	Conclusion	88
5	Conclusion and Perspectives	91
5.1	Conclusion	91
5.2	Perspectives	92
	Appendix A Hadoop Implementation	95
A.1	Hadoop Starting	95
A.2	Results based on Data Collected and Storage	96
	Publications	99
1	Accepted and Submitted Journals	99
2	Accepted and Published Conferences	99
	<i>Bibliography</i>	101

Contents

<i>Abstract</i>	I
<i>Contents</i>	III
<i>List of Figures</i>	VIII
<i>List of Tables</i>	XI
<i>List of Abbreviations</i>	XIII
<i>Notations</i>	XV
<i>Dedication</i>	XVIII
<i>Acknowledgements</i>	XX
Intorduction	XXI
1 General Introduction	XXI
2 Main Contributions of this Thesis	XXII
2.1 At Sensor Level	XXII
Data transmission techniques	XXII
Data collection techniques	XXII
2.2 At sink Level	XXII
3 Thesis Structure	XXIII
3.1 Chapter 1: An Overview About Wireless Body Sensor Networks	XXIII
3.2 Chapter 2: An Adaptive Intra-WBSN Communication approach	XXIII
3.3 Chapter 3: P2D: An Efficient Patient-to-Doctor Framework for Real-Time Health Monitoring and Decision Making	XXIII
3.4 Chapter 4: A Sensing-Based Patient Classification Framework for Efficient Patient-Nurse Scheduling	XXIV
3.5 Chapter 5: Conclusions and Perspectives	XXIV
1 An Overview About Wireless Body Sensor Networks	1
1.1 Introduction	1
1.2 Various Types of Biomedical Sensors	1
1.2.1 Epidermal Biosensors	2
1.2.2 Saliva-based Biosensors	2
1.2.3 Tear-based Biosensors	2
1.3 WBSN in Internet of Things (IoT)	2
1.4 WBSN 3-Tier Architecture	3
1.5 Intra-WBSN Communication Types	3
1.6 WBSN Challenges	3

1.6.1	Reducing Energy Consumption	5
1.6.2	Rapid Emergency Detection	5
1.6.3	Predicting Progress of Patient Situation	5
1.6.4	Big Data Processing	5
1.6.5	Nurse Scheduling	5
1.7	Patient Monitoring and Risk Assessment: A Background	5
1.7.1	Energy Efficient Data Transmission Techniques	6
1.7.2	Analyzing Large Amounts of Data for Decision Making Techniques	8
1.7.3	Real-Time Patient Monitoring and Assessment Techniques	11
1.7.4	Patient Classification and Risk Assessment Techniques	14
1.7.5	Nurse Scheduling Techniques	16
1.7.6	IoT in Healthcare Applications	19
1.8	Conclusion	21
2	An Adaptive Intra-WBSN Communication Approach	23
2.1	Introduction	23
2.2	Comparative study of recent advances in data transmission in IoT	24
2.2.1	An Energy-Efficient Routing Protocol (EERP) for Reliable Data Transmission in WBSN [1]	24
2.2.2	RK-Energy Efficient (RKEE) Routing Protocol for WBSN [2]	25
2.2.3	Self-Adaptive Data Collection and Fusion (SADCF) for Health Monitoring in WBSN [3]	26
2.2.4	Sleep Scheduling in Energy Harvesting (SSEH) WBSN [4]	27
2.3	Optimised Intra-WBSN Communication Approach	28
2.3.1	Network Initialization	28
2.3.2	Emergency Detection	28
2.3.3	Next Hop Node Selection Based on Benefit function (M)	30
2.3.4	Data Transmission:	31
2.4	Simulation Results	31
2.4.1	Comparison of the Studied Approaches	32
2.4.2	Comparison with our Optimised Intra-WBSN Communication Approach	33
2.5	Conclusion	36
3	P2D: An Efficient Patient-to-Doctor Framework for Real-Time Health Monitoring and Decision Making	39
3.1	Introduction	39
3.2	WBSN Framework	40
3.3	Patient Monitoring Model	41
3.3.1	Patient Emergency Detection	42
	Recall of LSA Method	43
3.3.2	Adapting Sensing Frequency	44
3.3.3	Patient Monitoring Algorithm	46
3.4	Medical Team Decision-Making Model	47
3.4.1	Data Ingestion Layer	47
	Flume	47
	Spark Streaming	47
3.4.2	Data Processing Layer	47
	Patient Records Archive Algorithm	48

Moving Average (MA) Algorithm	48
Exponential Smoothing (ES) Algorithm	49
3.4.3 Data Storage Layer	50
3.4.4 Data Visualization and Prediction Layer	50
Grafana	50
Matplotlib	50
Prediction of the Patient Situation Progress	50
3.4.5 Clinical Response Decision-Making Algorithm	52
3.5 Simulation Results	53
3.5.1 Patient Situation Study	53
3.5.2 Performance Study of PRA Algorithm	53
The Effect of LSA Degree (k)	54
The Effect of Division Size (d)	54
3.5.3 Performance Study of PMA Algorithm	56
3.5.4 Data Transmission Study	57
3.5.5 Performance Study of MA	58
3.5.6 Performance Study of ES	60
3.5.7 Performance Study of Processing Speed	60
3.5.8 The performance Study of Prophet Method	60
3.6 Conclusion	63
4 A Sensing-Based Patient Classification Framework for Efficient Patient-Nurse Scheduling	65
4.1 Introduction	65
4.2 Patient Classification Phase	66
4.2.1 Training Phase	66
Shapelet Generation	66
Shapelet Selection	67
4.2.2 Testing Phase	67
At Sensor Level	68
At Sink Level	68
4.3 Nurse Scheduling Phase	68
4.3.1 Patients Assigning to Nurses Algorithm	69
Patients Distribution-based Criticality Balanced (PDCB) Mechanism	69
Patients Sorting-based Grouping (PSG) Mechanism	70
Genetic Algorithm combined with Particle Swarm Optimization (GA-PSO)	70
4.3.2 Patient-Nurse Scheduling (PNS) Algorithm	74
4.4 Research Results	76
4.4.1 Performance Study of Shapelet Generation Method	77
4.4.2 Performance Study of Finding Similar Shapelet	77
4.4.3 Performance of Patient Classification Model	79
4.4.4 Patients Assigning to Nurses Study	79
Performance of PDCB and PSG	79
Performance of GA-PSO	81
4.4.5 Performance of Patients Scheduling	86
4.5 Conclusion	88
5 Conclusion and Perspectives	91

5.1	Conclusion	91
5.2	Perspectives	92
5.2.1	Short to Mid Term Perspectives	92
5.2.2	Long Term Perspectives	92
Appendix A Hadoop Implementation		95
A.1	Hadoop Starting	95
A.2	Results based on Data Collected and Storage	96
Publications		99
1	Accepted and Submitted Journals	99
2	Accepted and Published Conferences	99
	<i>Bibliography</i>	101

List of Figures

1.1	Various types of biomedical sensors	2
1.2	WBSN architecture	4
1.3	Intra-WBSN Communication Types	4
2.1	Flow chart of the proposed protocol	29
2.2	Early Warning Score (EWS) System	29
2.3	Analysis of network lifetime	35
2.4	Analysis of energy consumption	35
2.5	Analysis of transmission time in one period	36
2.6	Analysis of overhead on sink	37
2.7	Analysis of data redundancy reduction	37
3.1	Overview of the proposed WBSN architecture	41
3.2	Round-based sensor lifetime with periodic collection model	41
3.3	Illustrative example of PED	43
3.4	Illustrative example of MA algorithm	49
3.5	Illustrative example of ES algorithm	50
3.6	Illustrative example of applying Prophet	52
3.7	Patient situation according to the percentage of reading scores	54
3.8	The quality of regenerated data after applying PRA in function of k , patient situation is low	55
3.9	The quality of regenerated data after applying PRA in function of d , patient situation is low	56
3.10	The variation of the sensing frequency of the sensor after applying the PMA in function of the patient's situation level, $\mathcal{T} = 3600$	57
3.11	The percentage of sensing data after applying the PMA algorithm in function of the patient's situation	58
3.12	Percentage of data transmission from sensor to the sink, $k = 5$	59
3.13	Raw data vs regenerated data with MA	59
3.14	Raw data vs regenerated data with ES	60
3.15	Processing speed using MA and ES	61
3.16	The predicted data using Prophet compared to the raw data, $\mathcal{T} = 900$, $\mathcal{P} = 15$, patient situation is medium	62
4.1	Patient classification system	66
4.2	Illustrative example of shapelet generation	67
4.3	Illustrative example of PSO exploration space search	72
4.4	Segment selection from the chromosomes	73
4.5	Group shifting	73

4.6	Combining genes	73
4.7	Adding zeros	73
4.8	Offspring after mutation operation	74
4.9	Illustrative example of PNS algorithm	76
4.10	The shapelets generated for <i>SPO2</i>	78
4.11	Similar shapelet elected for every vital sign of the high Patient 1	79
4.12	Similar shapelet elected for every vital sign of the medium Patient 9	80
4.13	Similar shapelet elected for every vital sign of the low Patient 3	81
4.14	The criticalities levels of Patient 1 during 10 periods	81
4.15	The criticalities levels of Patient 9 during 10 periods	82
4.16	Distribution of patients after the first period	82
4.17	The variation of the patient number in each cluster	82
4.18	Accumulation criticality in each cluster	83
4.19	The priority level of patients obtained in function of age and criticality rate	87
4.20	The scheduling results obtained	88
A.1	Hadoop starting	95
A.2	jps	96
A.3	Properties of flume	96
A.4	Results based on Data Collected and Storage	97

List of Tables

1.1	Energy consumption in sensor network: a state of the art techniques	9
1.2	Analyzing large amounts of data in sensor network: a state of the art techniques	12
1.3	Real-time patient monitoring in sensor network: a state of the art techniques	15
1.4	Patient classification and risk assessment in sensor network: a state of the art techniques	17
1.5	Nurse scheduling in sensor network: a state of the art techniques	20
2.1	Resulted values for the different parameters of four approaches	32
2.2	State-of-the-Art: A comparison	34
3.1	Description of the patient criticality level	45
3.2	Sensing frequency decision table (SFDT)	46
3.3	Clinical response decision-making table (CRDM)	53
3.4	Simulation environment.	53
4.1	Level and corresponding score of v_s	68
4.2	Description of patient severity level	68
4.3	Simulation environment	77
4.4	GA-PSO results with ST_1	84
4.5	GA-PSO results with ST_2	85

List of Abbreviations

ABP: Average Blood Pressure
BT: Body Temperature
CH: Cluster Head
CRDM: Clinical Response Decision-Making
DTW: Dynamic Time Warping
EWS: Early Warning Score
ES: Exponential Smoothing
GA: Genetic Algorithm
HR: Heart Rate
HDFS: Hadoop Distributed File System
IoT: Internet of Things
IoHT: Internet of Healthcare Things
LSA: Least Squares Approximation
MIMIC: Multiple Intelligent Monitoring in Intensive Care
MA: Moving Average
MLED: Modified Local Emergency Detection
P2D: Patient-to-Doctor
PULSE: Pulse
PED: Patient Emergency Detection
PRA: Patient Records Archive
PMA: Patient Monitoring Algorithm
PDCB: Patients Distribution-based Criticality Balanced
PSG: Patients Sorting-based Grouping
PNS: Patient-Nurse Scheduling
PSO: Particle Swarm Optimization
RESP: Respiratory Rate
SPO2: Oxygen Saturation
SFDT: Sensing Frequency Decision Table
WBSN: Wireless Body Sensor Network

Notations

M_{ij}	Benefit function of each node neighbor j of sensor node i
$\alpha, \beta, \theta, \gamma$	Weights in M_{ij}
n_i	Sensor node i
Er_i	Remaining energy of sensor node i
t_i	Efficiency of Transmission of node i
Pf_i	Forwarded Packets Successfully by node i
Pr_i	Received Packets
t_{min}	Minimum number of packets that are successfully transmitted
t_{max}	Maximum number of packets that are successfully transmitted
Bav_i	Bandwidth available of node i
B_{min}	Minimum Bandwidth
B_{max}	Maximum Bandwidth
H_i	Number of hops to reach sink o node i
H_{max}	Maximum number of hops to sink
u	Round time
p	Period time
\mathcal{T}	Period size
ws	Window of size
y_i	Slot time
r_i	One reading at y_i
Est_1	Estimated reading at slot time s_1 that is initially equal to r_1 in ES algorithm
Est_i	Estimated value at slot time s_j in ES algorithm
ζ	Smoothing factor in ES algorithm
\mathcal{N}	Set of sensors
\mathcal{VS}	Set of Vital signs
vs	A vital sign
\mathcal{U}	Set of rounds
${}_pR_u^{vs}$	Set of readings of vital sign vs during period p and round u
${}_pS_u^{vs}$	Set of record scores of vital sign vs during period p and round u
${}_pC_u^{vs}$	LSA coefficient set

d	Division size
k	Degree of LSA polynomial
$wgt(s_i)$	Score weight function
\mathcal{C}	Criticality level
\mathcal{R}	Risk level
$ScoreU$	Aggregated score
s_i	score of reading r_i
SH	Shapelet
l	Length of SH
D	Data series
L	Length of D
W	Time warping path
NP	Number of patients
NN	Number of nurses
NS	Number of sensors
Patient i	$0_{i < 0}^{\text{th}}$ patient
Nurse i	$0_{i < 0}^{\text{th}}$ nurse
ε	Balancing threshold
PRI_i	Priority of patient P_i
A_i	Age of patient P_i
λ	Weight of the patient criticality
ψ	Weight of the patient age
δ	Subjectivity threshold
V	set of vertices
C	Nurse shift
tf	Service at time
d_{ij}	Euclidean distance between node i and node j
ST	Serving time
ω	Population size

Dedication

I am, dedicating this work to the memory of my father, Hassan SAAD. I miss him every day. He has always encouraged me to learn, to develop my abilities, and he is credited with reaching the completion of this thesis.

Acknowledgements

First and foremost, I thank and praise Allah (Subhanahu Wa Ta'ala), there is no power but from God. Thanks to God who gave me strength, I completed this productive work. And I thank all the people who followed and contributed with me to produce this work.

I would like to express my sincere gratitude to my supervisors: Prof. Abdelhafid ABOUAISSA and Prof. Lhassane IDOUGHMAR, at the university of Haute-Alsace, and Dr. Hassan HARB and Dr. Nour CHARARA, at the American University of Culture and Education (AUCE), for their guidance during my research. They gave me strength, knowledge and advice, and they supported and encouraged me continuously. I am very grateful to them, for their contributions to ideas and times to make my Ph.D stronger and better.

Moreover, I would like to thank the president and the members of the jury committee: Prof. Abdallah MAKHOUL, Prof. Ye-Qiong SONG, Prof. Laetitia JOURDAN and Prof Jaafar GABER for their time, interest, and helpful comments. It is an honor to have my work examined and assessed by professional experts such as yourselves.

I would like to express my warm appreciation to my family who were fans along the way and must be thanked my mother Nidal, for giving me patience, effort, psychological and moral support. She was and still encourage me on the path of science and success. My brothers Ali and Mohamad, My sisters Mariam, Soha and Rawan. My sister Soha, who supported me in my work, and all my family, God reward you with all the best.

Intorduction

1 General Introduction

In home, at work, in schools, our bodies are the target for hundreds of diseases and viruses. From one hand, our daily food along with the connection human-human or human-animal are the most causes of diseases. On the other hand, pollution of climate and wars are the indirect causes of the propagation of severe viruses. This leads to significantly increase the number of patients as well as the healthcare costs to governments and societies. Therefore, in order to deal with the increasing number of patients, integrating new technologies in healthcare monitoring and assessment to reduce the medical worker's cost and ensure a near patient-doctor interaction has become an essential for hospitals nowadays.

Among other technologies, researchers have focused on wireless body sensor network (WBSN) [5] as an efficient and low cost monitoring system for various healthcare applications, either in-hospital or in-home. Indeed, WBSN consists of a group of sensors that are located in and on the patient's body where each of which collects data for one vital sign (heart rate, systolic blood pressure, body temperature, oxygen ratio, respiration rate, etc.). Then, the collected data are sent periodically to a coordinator that is located on or beside the patient body, and this coordinator sends them to the sink. Finally, the doctors are responsible to check and analysis the data in order to make the ultimate decision.

Indeed, the huge amount of collected data in WBSN along with the growing number of patients provides several challenges for both hospitals and doctors. First, the sensors are equipped with autonomous power batteries but with limited capacities. Optimizing their energy consumption is a fundamental operation, in order to increase the network lifetime and ensure long-term patient monitoring. However, several works have proposed solutions for managing energy consumption based on the aggregation of the data sent or reducing the frequency of sending by transmitting only the relevant data following a change of state. Second, studying the evolution of the patient's behavior and predicting his future situation, based on his current situation, is one of the important objectives of health care. Therefore, the patient must be taken care of in order to receive adequate treatment before entering a critical situation. Finally, a scheduling technique that allows each nurse to determine her group of patients and specify the best conditions for their care is one of the basic operations to establish a balanced workload of medical staff due to epidemic periods.

2 Main Contributions of this Thesis

The main contributions in this thesis concentrate on designing an efficient energy data collection and transmission techniques at sensor nodes, on designing a nurse-patient scheduling algorithm and data analysis models at the sink node, and on building a patient classification model and hadoop-based framework at sensor and sink nodes.

2.1 At Sensor Level

In sensing-based application, the basic unit in the WBSN is the sensor node that continuously monitors the physiological statuses of patients, by collecting, exchanging and then sending analyzed patient's data remotely to processing center in order to make suitable decisions. Indeed, the collection model suffers from the big amount of redundant data collected by sensors. This big data produces two main challenges in WBSN: First, it quickly depletes the small energy battery in sensor which is not always replaceable or rechargeable. Second, it makes the decision-making process is very complex for experts because of the high level of redundancy existing among the data. In this thesis, we propose several techniques based on the process of data transmission and of sensing frequency adaptation, in order to reduce the redundant of collected and transmitted data, and therefore saving the sensor power.

Data transmission techniques

Indeed, the energy consumption is highly dependent on the data transmission operation. Therefore, in this thesis we propose several techniques that aims to reduce the amount of transmitted data in order to minimize the energy consumption. First, we propose an emergency detection algorithm that aims to directly alert the medical staff to any abnormal situation of the patient and to reduce the amount of periodic data transmission from each sensor to the coordinator, and therefore extending the network lifetime and taking the treatment needed. Second, we design the first phase of the patient classification model, that aims to calculate and send the level of criticality of the specific monitored vital sign, instead of sending all the collected readings to the coordinator. Third, we propose a multi-hop routing protocol that reduces the redundant collected data depending on the patient's condition, and improves the data transmission by finding the optimal path to send data between sensors according to several parameters: remaining energy; bandwidth available; transmission efficiency; hops to sink.

Data collection techniques

Obviously, the redundancy level among the collected data is dependent on two reasons: the stabilization of the patient's condition and the short time between the collected measurements. In this thesis, we propose an adaptive sensing frequency algorithm that reduces the amount of the collected data according to the criticality level of the patient.

2.2 At sink Level

In the sensing-based application, the sink node is responsible for stored and analysed the data collected from the sensor network and it can impart knowledge to the medical staff. In this thesis, we propose several techniques that drive the medical staff to make the suitable decision. First, we design a framework based on hadoop tools. The sink receive the data collected from the sensors then forward them to the hadoop cluster, in order to preprocess and store them for later analysis. Second, we propose a patient records archive technique that regenerates the readings for each patient during his staying in the hospital, in order to store and archive them for a latter data analysis. Third, we propose a patient situation prediction technique that predicts the variation of the situation of a patient during the next periods in order to take a quick response and avoid patient to enter in a critical situation. Fourth, we design the second phase of the patient classification model that determines the

criticality level of the patient according to the received levels of all vital signs from the patient's body. Finally, we propose a patient-nurse scheduling technique in order to balance the workload of the medical personnel. This technique starts by distributing the patients to several groups, where each group is assigned to a specific nurse. This distribution assures the similarity between groups in the number of patient and in the levels of criticality, in other hand it ensures the dissimilarity in patient levels in the same group. We proposed three mechanism, two based on grouping operation, while, the third mechanism is based on the combining between the genetic algorithm and particle swarm optimization. After the distribution of patients, we build a scheduling algorithms in order to find the best traffic that a nurse should follow in order to serve all her assigned patients. The algorithm is based on calculated the priority of each patient according to the criticality and age parameters.

3 Thesis Structure

The thesis is structured as follows:

3.1 Chapter 1: An Overview About Wireless Body Sensor Networks

This chapter presents a general review about wireless body sensor networks (WBSNs). Then, it introduces the architecture of WBSN. Also, it shows the various types biomedical sensors. After that, it demonstrates the integration of Internet of Things in the healthcare field. In addition, it describes the challenges that face the implementation of WBSNs along with the growing number of patients while focusing on energy consumption, emergency detection and patients-nurse scheduling. Finally, it presents a state-of-the-art in order to overcome the highlighted challenges while the proposed techniques is classified according to specific criteria (is divided into several categories): patient classification, nurse scheduling, energy efficient data transmission, real-time patient monitoring, and analyzing large data volume for decision-making.

3.2 Chapter 2: An Adaptive Intra-WBSN Communication approach

This chapter presents a comparative study for the recent advances in Internet of healthcare monitoring and it introduces a novel approach based on merging an energy efficient routing protocol with the data transmission process. This approach reduces the data acquisition redundancy, and finds the optimum path for data transmission, according to the patient situation. At the sensor level, the approach proposes a multi-hop routing protocol that select the next hop based on several parameters: residual energy, transmission efficiency, bandwidth available and number of hops to reach the sink. In order to reduce the energy consumption and to ensure an efficient transmission. In addition, the approach integrates a data transmission technique that sends the measurement to the next hop only if the patient's situation has changed, in order to reduce the redundant data, and therefore extending the network lifetime.

3.3 Chapter 3: P2D: An Efficient Patient-to-Doctor Framework for Real-Time Health Monitoring and Decision Making

This Chapter introduces an efficient framework called P2D for health monitoring and decision making. P2D works on two levels: sensors and sink, which covers the entire life-cycle of data including data collection, data ingestion, data preprocessing, data storing, and data visualization. The proposed framework relies on Hadoop ecosystem tools. At the sensor level, P2D proposes an emergency detection algorithm that allows to directly detect any abnormal situation of the patient, and proposes an adapting sensing frequency algorithm based on the patient criticality level, that allows to save the sensor energy. Whilst, at the sink level, P2D proposes a patient records

archive technique that allows to store an archive for each patient using the method of least squares approximation, then it introduces two fault detection algorithms (moving average and exponential smoothing) in order to preprocess data before storage, and proposes a prediction technique based on the prophet method that predict the patient situation during the next periods of time and make a suitable decision by the doctors.

3.4 Chapter 4: A Sensing-Based Patient Classification Framework for Efficient Patient-Nurse Scheduling

This chapter introduces a novel and efficient framework for nurse-patient intelligent task organization that treats the scheduling problem. Typically, our framework consists of two phases: patient classification, and nurse scheduling. The first phase introduces an efficient classification model that groups patients according to the severity level of their vital signs. The second phase starts with proposing three distribution mechanisms based on the classification results in order to balance the workload of the stressed medical personnel. Then a scheduling algorithm is proposed in order to find the best scheduling assignment of nurses over patients. This algorithm is based on some priority metrics.

3.5 Chapter 5: Conclusions and Perspectives

This chapter introduces the conclusion of our work, and suggests some extended improvements to our techniques as a future work.

1 An Overview About Wireless Body Sensor Networks

1.1 Introduction

The essential role of primary health care is to ensure continuous and comprehensive care to the patients. Thus, the healthcare system absorbs a significant and growing share of resources. However, Wireless body sensor network (WBSN)[5] is a well-known solution that is used to help nurses in monitoring patients in home or in hospital. WBSN is a network of sensors implanted in and/or on the patient's body, and constantly monitors their health status in real time. Where, each sensor collects periodically data, then send them to the medical center in order to make suitable decision, and therefore in order to prevent any unwanted situation with the necessary intervention medical.

Indeed, WBSN suffers from big data collected by the medical sensors. Therefore, collecting and transmitting data are the major reasons behind the exhaustion of limited energy. Moreover, the researches focus on dealing, storing and analyzing the huge amount of the collected data. In addition, hospitals rely mainly on nurses who have several duties such as: serving the patients, contacting with doctors, giving medicine, detecting an emergency and monitoring vital signs [6]. WBSN is one of the main applications that helps the medical staff in electronic health surveillance with early detection of critical physiological symptoms. Moreover, rapid growth in number of patients, poses pressure on hospitals and increases the nurse duties. Hence, the patient-nurse scheduling algorithm has taken a big attention from researchers for reducing the workload of nurses.

1.2 Various Types of Biomedical Sensors

In WBSN, we distinguish between three main categories of sensors that produce different data types: numerical, images, or video. The numerical biosensors allow converting various forms of stimuli into electrical signals then to numerical records. These sensors take data about vital signs like pressure, temperature, oxygen saturation, respiration rate, etc. The image biosensors take images for various patient organs like X-ray and dental imaging, radiography, mammography, cardiology, etc. The last biosensor category, i.e. video biosensor, records data for various surgery operations of patients like cardiology, invasive surgery, ocular surgery and observation, and artificial retinas. On the other hand, biomedical sensors are implemented on or implanted inside the body [7]. Figure 1.1 shows the most used types of biosensors in WBSN which can be classified as follows:

1.2.1 Epidermal Biosensors

These types of sensors are usually attached on the body of the patient and aim to monitor the glucose, lactate, sodium, potassium and temperature (Figure 1.1(a)), the interleukin-6 and cortisol (Figure 1.1(b)). Mostly, these sensors are self-power supply electronic skin that use piezoelectric-enzymatic reaction coupling.

1.2.2 Saliva-based Biosensors

These biosensors usually mounted onto tooth with integrated wireless electronics. The saliva biomarkers offer meaningful diagnostic information, such as: sodium intake during hypertension management (Figure 1.1(c)), detect foods and fluids during ingestion sugars, alcohol, salinity, pH and temperature (Figure 1.1(d)).

1.2.3 Tear-based Biosensors

These sensors rely on contact lenses and use transparent and soft materials while incorporating wireless electronics. They detect glucose concentrations via resistance-based enzymatic mechanism and allows quick response time for continuous measurements (Figure 1.1(e,f,g)).

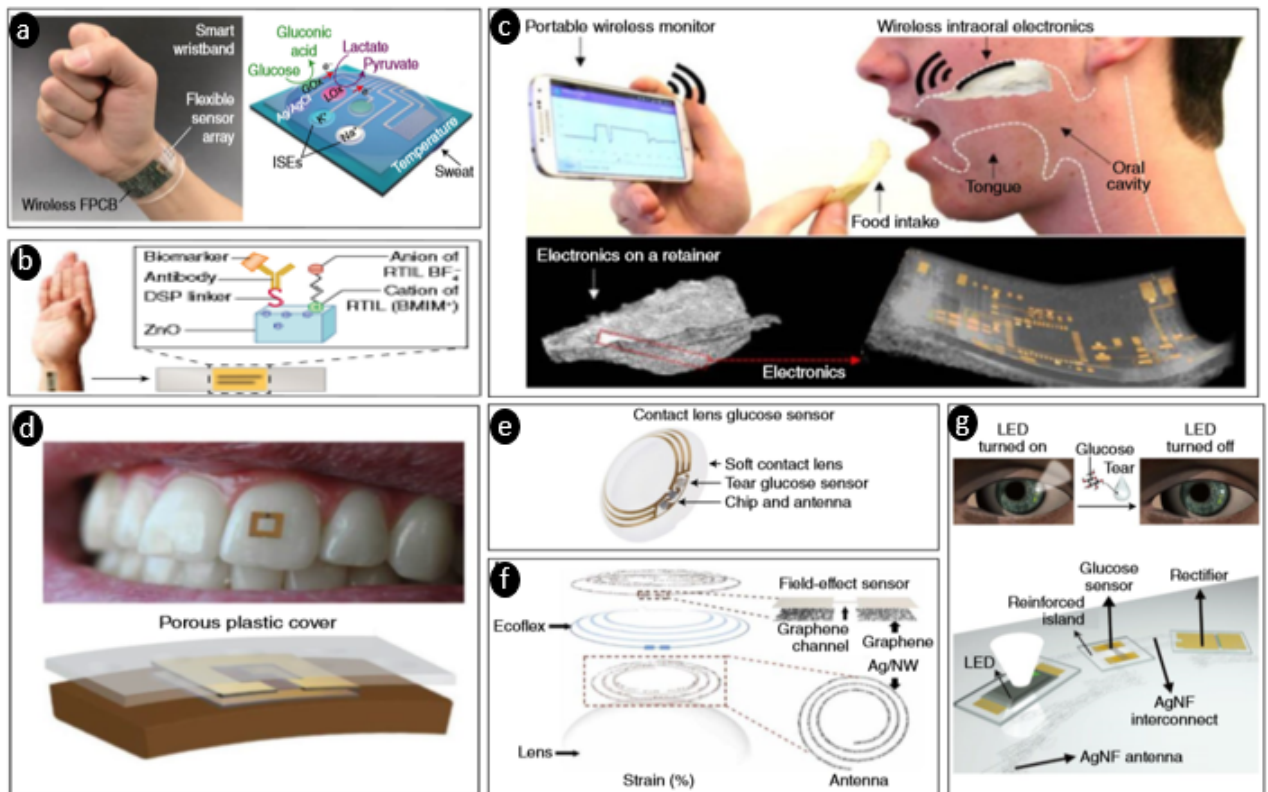


Figure 1.1 Various types of biomedical sensors.

1.3 WBSN in Internet of Things (IoT)

Recently, Internet of Things (IoT) systems were introduced in the field of healthcare, playing a major role in patient monitoring and in ensuring a good communication between the healthcare

providers [8]. IoT describes the network of physical objects that are embedded with sensors, software, and other technologies for the purpose of connecting and exchanging data with other devices and systems over the internet. In fact, the wireless body sensor network (WBSN) technologies are one of the essential technologies of IoT growths of the healthcare paradigm. In WBSN every patient is monitored through a group of small-powered and lightweight sensor nodes [9]. IoT based healthcare or IoHT [10] allows monitoring physiological statuses of patient, by collecting and exchanging their data. These systems help reducing the average workload of medical staff in hospitals. Indeed, the remote monitoring from home reduces the patient hospital visits especially for elderly people. Moreover, adding an analysis part in the IoT healthcare systems contributes significantly in decision making by the doctors. Furthermore, IoHT systems contribute in medical work improvement by reducing the expenses in human, medical and financial resources.

1.4 WBSN 3-Tier Architecture

WBSN system is the key component of the patient monitoring system. The topology of WBSN is composed of three tiers as shown in Figure 1.2. Typically, the first tier consists of several sensors that are distributed inside and on the patient's body in order to monitor and measure the different vital signs. These measurements are sent to a coordinator that is located in the center of the patient's body. The coordinator sends the data to the sink in the second tier. This sink could be a smart phone, personal computer or other smart electronic device. These devices can play the role of a gateway. Finally, some specific information is transmitted from the gateway to the tier-3 via the Internet. This third tier offers several medical services like medical information storage, retrieval and analysis. On tier-3 medical information could be directed to physician or emergency team for necessary treatment. Indeed, Sensor nodes are powered by micro-batteries and each node is responsible of collecting one or more type of physiological data including heart rate (HR), respiratory rate (RESP), oxygen saturation (SPO2), pulse (PULSE) and blood pressure (ABP). In almost WBSN applications, a continuous monitoring of patient situation is required. Therefore, we are interested on the periodic data collection model. In our system, each sensor collects one data measurement on each second, and at the end of the period the sensor sends the collected data to the coordinator which, in turns forwards the data to the sink for a later data analysis and decision.

1.5 Intra-WBSN Communication Types

In this section, we introduce the three types of sensor-coordinator and sensor-sensor communication; broadcast, single hop and multi-hop. In broadcast communication, every sensor communicates with all others. This cluttered communication is considered high energy consuming. In single-hop type (Figure 1.3(a)), the sensor sends the data directly to the coordinator through one hop. Therefore, the signal strength should be significantly increased leading to the increasing of energy consumption. In multi-hop communication (Figure 1.3(b)), the data follow an adaptive routing between the sensors, minimizing by this way the required signal and reducing therefore the energy consumption.

1.6 WBSN Challenges

Typically, WBSN is one of the main technology that used in the monitoring healthcare applications. The life cycle in the WBSN begins with the acquisition of data for a patient and ends with its analysis by the medical profession. However, this process faces several types of challenges. Some essential challenges are related to the limited energy sensor, while other challenges are related to the data management (such as big data collecting, transmitting, storage and processing), and, finally, other

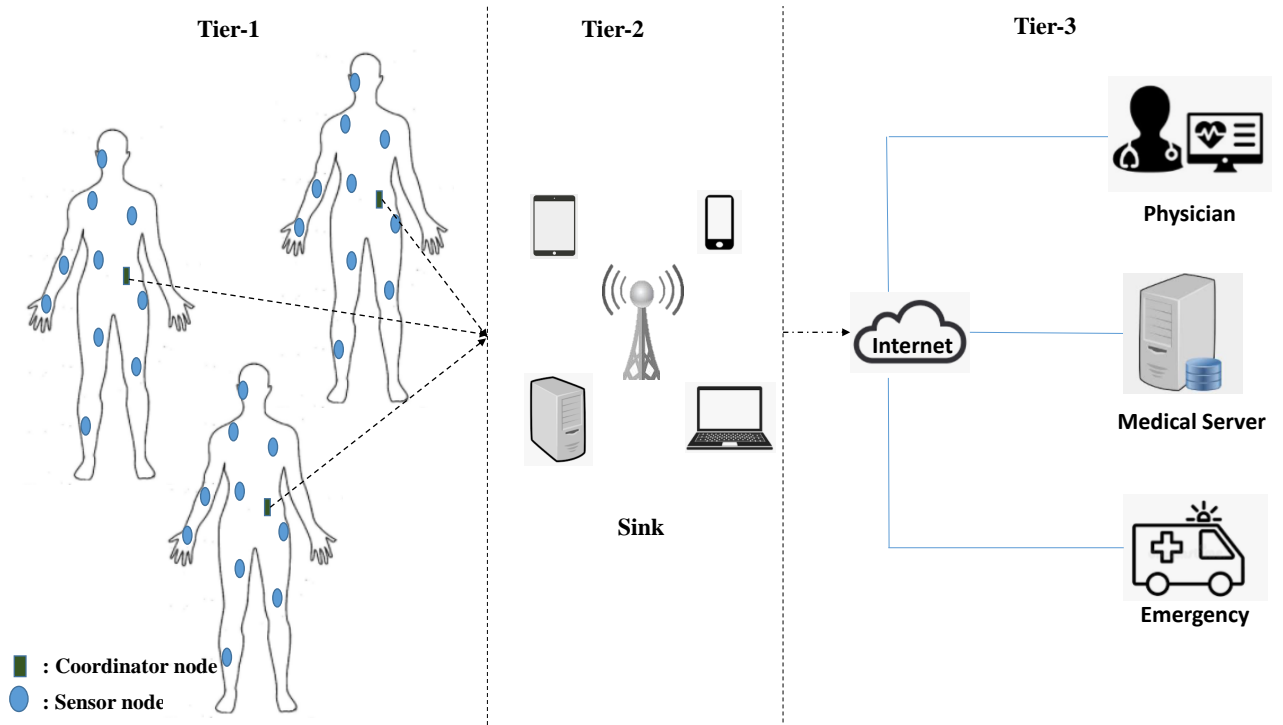


Figure 1.2 WBSN architecture.

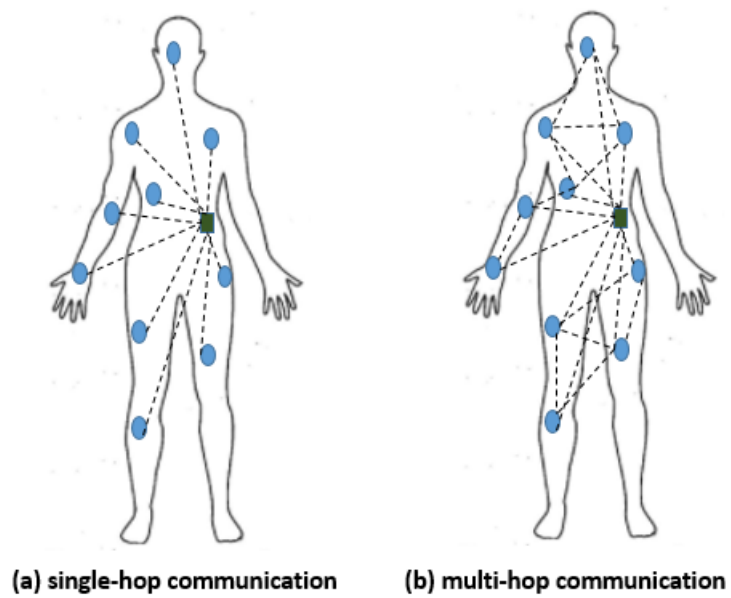


Figure 1.3 Intra-WBSN Communication Types.

ones are related to patient-nurse scheduling. This challenges are for both hospitals and doctors which can be summarized as follows:

1.6.1 Reducing Energy Consumption

As sensors have autonomous power batteries, minimizing energy consumption is a fundamental operation in WBSN in order to ensure a long time patient monitoring and reduce energy cost. Furthermore, data transmission is a heavily cost operation in terms of energy consumption. Hence, adapting sensing frequency techniques have been proposed as an efficient approach for reducing data collection and transmission from sensors.

1.6.2 Rapid Emergency Detection

Usually, the situation of the patient can change from low to high criticality level. Thus, in case of emergency detection, an alert must be directly sent to the medical staff and doctor in order to evaluate the patient situation and take the needed treatment. Therefore, rapid emergency detection is a crucial task for medical team which can negatively affect the life of a patient.

1.6.3 Predicting Progress of Patient Situation

Obviously, patient monitoring and having information about his current situation is not always a sufficient task for medical staff. For instance, a patient is entering hospital in critical case but the rapid variation of his situation can lead to an unpredictable death. Hence, studying the progress of patient behavior and predicting his future situation, according to the current one, is one of the important goals in healthcare. By doing that, patient can be given the adequate treatment before entering a critical situation.

1.6.4 Big Data Processing

The acquisition of various data for a large number of patients leads to the problem of mega-data (Big Data) at the final sink level [11]. On the one hand, this huge amount of data complicates the mission of data analysis by the medical team, especially in the case of emergencies. On the other hand, the richness of this data will be a source of opportunity for Big-Data analysts to refine them and to propose models in line with the different states of patients, to validate hypotheses and of course to take decisions. Similarly, reducing the amount of redundant information transmitted by the nodes reduces the energy consumption in the system. Among the techniques proposed to reduce the transmission of redundant information are: data aggregation, compression and prediction. One of the main contribution of this thesis consists in developing data reduction/compression/prediction techniques and methods for adapting the detection frequency according to the application domain to save energy and reduce data size.

1.6.5 Nurse Scheduling

Indeed, one of the fundamental challenge in the hospital is to balance the workload of medical personnel especially in the epidemic time. A scheduling protocol aims to assign for each nurse her group of patients, then to determine the optimum path in order to serve his group. Normally, this scheduling algorithm depends on several parameters such as the level of severity of the patient and his age, the capacities of the nurses, the occupancy rate of the resuscitation rooms and medical equipment, etc.

1.7 Patient Monitoring and Risk Assessment: A Background

In the last decade, the healthcare sector takes a great attention from both industries and research community. From one hand, the industries try to integrate new technologies in the medical process while the governments aim to ensure a secure public health for the people. On the other hand,

researchers aim to propose efficient patient monitoring and assessment techniques as well as try to overcome the challenges provided in WBSN. The authors of [5] give an overview on the sensing devices fabricated by the industries and dedicated to healthcare applications. In [12, 13], the authors present and compare the remote health monitoring systems existing in the literature for elderly healthcare and well-being. Also, they survey on various textile-based sensors that can potentially be used in wearable systems as long as the communications technologies dedicated to elderly ubiquitous healthcare [14]. Whilst, the authors of [15, 16] summarize the data analysis algorithms proposed by the research community for the sensor-based healthcare applications.

1.7.1 Energy Efficient Data Transmission Techniques

Some works in WBSN are dedicated to save the energies of the sensors and reduce the data transmission in the network.

The authors of [17] propose an energy efficient mechanism for WBSN based on a sleep scheduling strategy and dominating set method. After constructing the dominating graph, the sink selects, based on two approximation algorithms and a polymatroid function, a subset of nodes to collect the data (e.g. active nodes) while switching the other nodes to sleep mode in order to reduce the number of active sensors and therefore the energy consumption. Through a comprehensive simulation analysis, the authors demonstrated that this mechanism extends the network lifetime, but it doesn't take into account the integrity of information, where a number of sensors are in sleep mode which don't collect data during a period of time, and this sleep scheduling creates an information shortage for medical staff.

In [18], a priority-based energy harvesting scheme for charging embedded sensor nodes in WBSN has been proposed. The proposed scheme uses the CSMA/CA protocol in order to switch power from the primary unit to the secondary unit thus, saving the sensor voltage level and reducing the transmission losses. The limitation of this paper is that it consumes more time in transmission data operation, while reducing the period of transmission is important in the critical situation.

The authors of [19] propose a transmission protocol for WBSN where the nodes are divided into two categories: critical nodes and forwarder nodes. The forwarder nodes are responsible to collect data from sensors, aggregating them and sending to the sink. The forwarder nodes are selected according to two criteria: having maximum energies and minimum distance to the sink. The critical nodes use the single hop transmission and aim to send critical records directly to the sink. Simulation is carried out which shows better results than other solutions. However, the link information or signal strength is another significant parameter to select the forwarder node and only distance and energy level is not feasible because the high energy level sensor node does not give surety of better signal strength in the network.

The authors of [20] propose an efficient system in terms of energy saving in wireless body sensor networks, called LEACH-dual fuzzy logic (ELEACH-DFL), which aims to extend the network lifetime. The system is distributed on two stages. First, the authors select the cluster head (CH) using a fuzzy logic algorithm, according to some parameters like remaining energy and distance neighbor. Second, a cluster configuration technique is proposed in order to generate the clusters of each CH. The efficiency of ELEACH-DFL has been comprehensively evaluated through several studies and simulations, which have shown that ELEACH-DFL significantly extends the lifespan of the entire network. However, the selection and generation of clusters consumes more processing time, therefore the transmission time increases.

The authors of [21] propose a tele-monitoring technique for the elderly people using WBSN that sends emergency physiological data through event-based data transmission to reduce the data

redundancy and to improve the end-to-end delay. The data is transmitted on different time slots using Time Division Multiple Access (TDMA). The solution was evaluated through experiments and showed promising results in terms of reducing delay and energy consumption. However, the lack of security aspects seems to be its main shortcoming, where the privacy of patient has taken a great attention from researchers in the healthcare domain.

In [22], a multi-hop priority-based congestion-avoidance routing protocol in WBSN is proposed. The goal was to establish a routing protocol between the sensor nodes. In this work, the data is categorized into two: normal and critical. In case of normal data, the next-hop selection is determined based on three parameters; The remaining power, congestion on the forwarder node, and the signal-to-noise ratio of the path between the source and forwarder node. The proposed scheme uses data aggregation and filtering techniques in order to reduce network traffic, therefore saving energy. This protocol prioritizes the critical vital data for emergency detection in order to reduce the delay. This mechanism has shown promising results in terms of energy consumption, traffic load, delay, lifetime. However, the lack of security aspects seems to be the main shortcoming of them.

In [23], the authors propose an improved stable increased-throughput multi-hop link efficient routing protocol for Wireless Body Area Networks. This protocol uses a cost function in order to select the next node as a forwarder node. This forwarder node collects the data from the other nodes and transmits them to the sink reducing by this way the sensor energy consumption. This selection of the forwarder node depends on the remaining energy and the distance to the sink. Selecting the forwarder hop with maximum residual energy balances the energy consumption among the sensor nodes and the least distance reduces the path loss and hence improves the packet delivery ratio of the network. The simulation results presented in the paper seem to be promising in terms of network lifetime and network throughput using an integer linear program. However, this approach don't take into account other parameters like bandwidth available of nodes.

The authors of [24] propose an energy-balanced routing protocol (EBRP) in order to prolong the network lifetime. The authors used K-means++ algorithm in order to divide the sensor network into clusters. A cluster head is selected, using the fuzzy logical system, in order to collect the data from the whole cluster and send them to the sink. Additionally, the authors get the fuzzy rules using genetic algorithm (GA). The article conducted a series of experiments comparing the results with existing solutions. The efficiency of this work has been verified in terms of the extension of network lifetime. However, the main limitation of this work is that when GA selects the best rule list, it consumes more processing time.

In order to balance the energy consumption in the wireless body area network, the authors of [25] propose a multi-hop routing protocol. In this system, the sensor sends their values to the node that has largest residual energy. However, this protocol suffers from the high packet loss rate and the large delay.

In [26], a protocol for data routing called energy efficient peering routing (ERP) is proposed for handling the indoor WBANs applications. This protocol aims to estimate the communication cost of neighbor sensor nodes, then it uses its routing table to save this information. Choosing the downstream sensor node depends on residual energy and geographical position of nodes. ERP reduces the hello packet transmission using a control mechanism. The ERP architecture consists of three essential objectives: First, maintaining the hello packets generation. Second, building routing tables in order to update the neighbor sensor nodes position and location. Finally, building the owns updated routing table. However, this building operation takes additional time for data routing mechanism. Despite that the time is considered one of the important factor for real-time

data transmission.

The authors of [27] proposed a Multi objective QoS Routing (MQoSR) protocol for WBANs. This protocol aims to select the next node from neighbor nodes based on two parameters: the location that is determined using the GPS services and link information that is identified through residual energy of sensor nodes and reliability of sink node. MQoSR used fault tolerance approach in order to make multiple node-disjoint and it's based on demand routing protocols. Furthermore, this work focus on data transmission, end-to-end delay and network lifetime. The policies of routing depend on source and sink nodes available QoS requirements. However, in WBANs there are several types of QoS requirements needed and the process of achieving all the requirements requires more computational processing which increases the network overhead.

Table 1.2 shows a comparison analysis of the mentioned techniques in terms of the used methods, the performance evaluation metrics, and the limitations. Note that the different limitations mentioned are collected from different state of the art references.

1.7.2 Analyzing Large Amounts of Data for Decision Making Techniques

Other works in WBSN are dedicated to analyze the large amount of data collected in WBSN and making suitable decisions .

In [28], the authors propose a deep learning mechanism based on the fractional cat-based swarm algorithm for patient situation's assessment and decision making. First, the nodes are organized into clusters where a cluster-head (CH) is selected for each cluster based on the harmony search algorithm and a particular swarm optimization. After that, the authors used the deep belief network (DBN) in order to classify the data for better treatment. Where DBN is a multi-level neural network. Furthermore, the CH receives the records from the nodes and classifies them based on the deep belief network (DBN) in order to detect the emergency situations. The results described that the proposed mechanism better performs for living state of the art approaches in case of energy usage, accuracy, and throughput. Moreover, this classification process needs more computational processing time. Where the interesting factor for real-time health application is the time.

The authors of [29] propose a modern platform for healthcare information systems consisting of three layers is proposed. The first layer is composed of various data health sources like sensors, clinical report, medication, etc. The second layer aims to process and store data and it uses various Hadoop tools including Sqoop, HDFS, HBase, MapReduce and Hive. The last layer is responsible of applying business intelligence (BI) solutions over the stored data and it uses SpagoBI tools as an open source BI suite. The simulations results gives an efficient performance in terms of data storage. However, data warehousing process based on hadoop and hive suffer from two challenges. First, Hadoop is unable to handle the online analytical processing (OLAP) operations. Second, the language HiveQL of the Apache Hive query is unable to handle several standard SQL queries.

In [30], the proposed routing architecture is based on the WBAN data-management technique such as data segregation and classification, that plays an important role in handling the continuous transmission of the large amount of data generated in the healthcare field. The authors propose a classification technique based on a defined threshold where sensor's readings are classified into three types: urgent (above threshold), semiurgent (close to threshold) and nonurgent (less than threshold). Furthermore, the authors introduce a routing protocol for a medical sensor that enables transmitting packets during gateway failure. This work ameliorates data reliability and power consumption. In case of failure on the gateway equipment, the segregation algorithm transmits only the urgent packets directly to the gateway and drops the non-urgent packets but the semi-urgent packets are in the

Reference	Year	Techniques	Performance Evaluation	Limitation
[17]	2020	Sleep scheduling strategy Dominating set method	Reducing the number of active sensors Extending the network lifetime	Integrity of information
[18]	2019	CSMA/CA protocol	Saving sensor energy Reducing the transmission losses	Processing time
[19]	2018	Cost function Multi-hop protocol	Emergency detection Saving energy	Don't take into account the signal strength
[20]	2020	Fuzzy logic algorithm Clustering configuration technique	Extending the lifespan	Processing time
[21]	2020	Tele-monitoring technique Time Division Multiple Access	Reducing delay Reducing energy consumption	Lack of security
[22]	2019	Multi-hop routing protocol Data aggregation Filtering techniques	Emergency detection Reducing energy consumption Reducing traffic load Reducing delay Extending lifetime	Ignoring the mobility of sensors lack of security
[23]	2015	Cost function Integer linear program	Extending network lifetime Increasing network throughput	bandwidth available of node is not taken into account
[24]	2018	K-means++ algorithm Fuzzy logical system Genetic algorithm	Clustering accuracy Extending network lifetime	Processing time
[25]	2015	Multi-hop routing protocol	Saving energy consumption	High packet loss rate Large delay
[26]	2012	Hello packets generation Routing tables	Saving energy consumption	Processing time
[27]	2013	Fault tolerance approach On-demand routing protocols	Data transmission Network lifetime	Processing time

Table 1.1 Energy consumption in sensor network: a state of the art techniques.

pending state. The obtained results of the proposed technique shows better performance compared the other state of the art approaches in terms of energy usage, data reliability, PDR and packets transmission rate with avoiding network congestion. However, data classification technique at sensor node level increases overhead at sensor nodes. This work is not supported to handle several sensor nodes. In addition, there are three different processes such as classification, scheduling, and vertical handover decision were included in the monitoring of the telehealth system.

The authors of [31] propose a multi-sensor fusion and decision making mechanism for patient monitoring through WBSN. The objective of the proposed system is to detect gait abnormality in subjects with neurological disorders based on the gait features (especially spatio-temporal correlation, gait asymmetry and regularity) and machine learning approach. Disorders in the movement of patient are monitored and analyzed continuously in order to ameliorate the diagnosis rate. The authors identified the disorders by analyzing the health data using machine learning and layered data fusion to verify the correlation with clinical validations. However, the research bind the sensor to the ankle, and therefore, users have to wear sneakers, which sometimes is difficult for users to accept attaching the sensor to a fixed position.

In [32], the authors present a new paradigm called CloudDTH combining between digital twins and healthcare that is particularly dedicated to monitor elderly in their homes. Where, through precise analyzes the digital twin provides fast analysis and real-time decisions. The objective of CloudDTH is to improve medical services such that remote monitoring, diagnosing and predicting aspects of the health individual in terms of accuracy and speed, by integrating medical physics and virtual space. Moreover, this work suffer from several shortcomings. First, the effectiveness of digital twins in the data analytics process requires a connected and well thought through IT infrastructure. Second, the digital twins requires a stable, noise-free data flow that allows for high performance digital twins. Third, the large amount of used data affects the privacy and security aspect of digital twins.

In this research article [33] the authors exploit the advances of federated learning (FL) in the field of healthcare informatics. FL is a learning paradigm seeking to address the problem of data governance and privacy by training algorithms collaboratively without exchanging the data itself. Towards this goal, federated learning could be a promising direction, which trains algorithms across decentralised edge devices (eg, individual mobile phones) or servers hosting different local samples (eg, data owned by different samples). Data samples are not shared or centralised and only the trained models are communicated, which might improve data security and privacy of patient data for drug–disease outcome validation in drug repurposing. The performance of the proposed model is evaluated in simulations and the results obtained demonstrate significant improvement over other solutions.

In [34], the authors used the big data analytics in order to address the decision-making process. the performance comparison of the techniques is evaluated using metrics such as Accuracy, Recall and Specificity.

In [35], the authors investigate the performance of integration of big data analytics with the machine learning techniques like decision trees, support vector machine (SVM) and K nearest neighbor in enhancing the quality of healthcare services such as heart disease examination. The simulation presented in this work shows the performance comparison of the techniques in terms of accuracy, recall and specificity. The authors calculate the values of this metrics by the confusion matrix. Where, the obtained results shows that SVM performs better compared to decision tree and KNN techniques and gives better output for the dataset used.

The authors of [36] build a model based on the edge-cloud architecture for the secured big data analytics in order to ensure a timely decision-making process in the healthcare system. This model makes a private and secure healthcare monitoring system through the collaborating between the edges of mobile devices and the cloud level in order to increase the reliability of the biosensors connected. In addition, the authors used the greedy heuristics in calculating cost function to decrease the data retrieval rate based on mobile edges and therefore to enhance the handling of big data analytics. The authors reduce the network overhead by implementing the privacy algorithms on the sink and the mobile edges. The performance of the model is evaluated using many experimental tests. Moreover, on the security side, the model suffer from some potential network vulnerabilities. One other limitation depends on the optimization of the scalability factor for the mobile users that connected to several clouds.

Table 1.1 shows a comparison analysis of the mentioned techniques in terms of the used methods, the performance evaluation metrics, and the limitations. Note that the different limitations mentioned are collected from different state of the art references.

1.7.3 Real-Time Patient Monitoring and Assessment Techniques

Recently, the authors of [37, 3, 38, 39, 40, 41, 42, 43, 44, 45] open a new trend in sensing-based healthcare by proposing several frameworks for real-time patient monitoring and assessment.

In [37], a framework for a stress detection and evaluation is proposed. The framework works by detecting first stress signals according to skin conductance parameter, then the stress level is evaluated through fuzzy inference system based on patient vital signs, particularly heart rate, respiration rate and average blood pressure. The framework was evaluated through experiments and showed promising results in terms of predicting stress level. However, the framework lack of security and privacy.

In an early work [3], the authors propose a data management framework for data collection and decision making in sensing-based healthcare. The framework relies on three algorithms: first, an emergency detection algorithm aims to send critical records directly to the coordinator; second, an adaptive sampling rate algorithm based on ANOVA and Fisher test in order to allow each sensor to adapt its sampling to the variation of patient situation; third, a data fusion and decision making model is proposed to the coordinator based on decision matrix and fuzzy set theory. Although its great advantages for patient monitoring and assessment, the proposed framework suffers from several disadvantages: 1) in case of low critical patient none of the data would be archived in the hospital thus, revising patient archive to check patient progress from doctors is not possible; 2) predicting the progress of the patient situation in the next periods of time is not possible by the doctors.

In [39], an example of an integrated mobile platform is proposed. The system has originally been created for patients with diabetes and was modified to meet the specific needs for clinical trials in PBH patients. The platform consists of three components: First, a mobile app for patients that integrates manual data input (drugs, symptoms, meals,..), data from the sensors , and from the activity tracker. Second, a web interface for researchers and healthcare professionals that follows the health patient situation in real time using browser dashboard. Third, a cloud database that ensures secure environment for data collection and real-time monitoring. However, the mobile platform does not support all the mobile operating system like Android.

The work of [40] describes the use of machine learning in improving fall detection devices. The authors propose an effective fall detection system that uses edge computing to send data from cheap

Reference	Year	Techniques	Performance Evaluation	Limitation
[28]	2019	Harmony search algorithm Particular swarm algorithm Deep belief network	Emergency detection Energy usage Throughput Clustering accuracy	Computational processing time
[29]	2014	Hadoop tools (Sqoop, HDFS, HBase, MapReduce and Hive) Business intelligence (BI) solutions (SpagoBI tools)	Data storage	OLAP operations SQL queries
[30]	2017	Data segregation and classification techniques Routing protocol	Data reliability Power consumption Packets transmission rate Network congestion	Overhead at sensor level
[31]	2019	Machine learning Layered data fusion	Gait abnormality detection	Attaching the sensor to a fixed position
[32]	2019	Digital twins	Fast analysis Real-time decisions Diagnosing and predicting accuracy	IT infrastructure Noise data Privacy and security aspect
[33]	2020	Machine learning techniques : Decision trees Support Vector Machine (SVM) K nearest neighbor	Accuracy Recall Specificity	Privacy and security
[34]	2021	Edge-cloud architecture Greedy heuristics	Network overhead Data retrieval rate	Security vulnerabilities Scalability factor

Table 1.2 Analyzing large amounts of data in sensor network: a state of the art techniques.

wearable devices, apache flink to analyse the health data flow by implementing a data analytic pipeline, and long short-term memory technique to detect and classify the fall status. Through simulation, the efficiency of this approach is evaluated in terms of detecting falls in real-time. However, the work does not support all the types of sensors as well the parallel data-processing pipelines.

In [41], the authors propose an efficient sensor-based data analytics for real-time patient monitoring and assessment in order to achieve three goals: First, the framework allows the emergency detection by sending the abnormal health condition to the medical staff, in order to ensure a continuous monitoring of the patient's situation. Second, the authors create an algorithm in order to adapt the sensing frequency based on the criticality of the patient's situation, and therefore reduce the rate of transmitted data. Finally, a prediction technique is proposed based on the long short-term memory (LSTM) technique, which is an advanced recurrent neural network, it helps the medical staff in the right medical intervention process at the right time, in order to avoid any deterioration of the patient's situation. The authors applied this approach on real sensor data while the obtained results show its relevance in terms of reducing data transmission, predicting the patient's situation and saving network lifetime. Moreover, the adapting sensor frequency algorithm don't take into account the correlation between neighbor nodes, which lead to increase the rate of repeated collision.

The authors of [38] proposed a generalized multi-sensor fusion technique, called health risk assessment and decision-making (Health-RAD) algorithm, in order to control the situation of patients using WBSN. Health-RAD aims to detect the vital signs scores using a fuzzy inference system and early warning score systems (EWS). Then, based on the scores it determines the patient severity level that represented by a risk variable between 0 and 1. When the risk variable is higher, the health situation is more critical. Through several simulations, the obtained results proves the efficiency on reducing the energy consumption in processing data and extending the network lifetime. Moreover, the technique needs a careful effectiveness in real-case scenarios.

The authors of [42] created an ubiquitous computing environment for controlling the patient's health conditions in real time. The proposed algorithm consists of three components, which accumulate physiological data and offers indicators that support public policies to support physical activities. This technique did not implement WBSN module for deploying within Ubiquitous Computing Environment for Monitoring and Evaluating Physical Activity (UCEMEPA) environment to evaluate the physical activity sessions.

In [43], the authors designed a preventive health care system that allows monitoring of a patient's health care records. The architecture used in this system consists of heterogeneous nodes in order to ensure continuous monitoring and specific controls. In addition, the authors used a model checking approach that is based on model transformation for validating the WBSN behavior for health monitoring. The solution has been evaluated by experiments and has shown promising results in terms of handling the WBSN system, but it did not consider the scalability using huge WBSN system.

The authors of [44] designed a framework, named Mobile Agent Platform for SunSPOTs (MAPS) framework, for monitoring human activities by creating wireless sensor networks using Java-programmable Sunspot sensor platform. The agent-oriented programming gives detailed description of health care that helps the medical staff to monitor in continuous manner. The architecture of the WBSN is consists of a coordinator and two sensors. The coordinator is based on Java Adverse Drug Event (JADE) based enhancement tool, which handles configuring sensors, receiving data and recognizing the human activities. However, the method efficiently managed low-level sensor node functions and could provide high-level services to agents, but it was not capable of processing in deployed computers.

In [45], a data mining algorithm, called Online Distribution Resource Aware (ODRA) algorithm, is developed that aims to predict in real time the health risk of the patients based on previous health records. The ODRA was verified in real-time patient monitoring system for monitoring patient's physiological data. This algorithm ensures accurate status of patients and notifies patients of their health status to get immediate attention when needed. Moreover, the method was not able to detect the risk level alerts in the diagnostic system.

Table 1.3 shows a comparison analysis of the mentioned techniques in terms of the used methods, the performance evaluation metrics, and the limitations. Note that the different limitations mentioned are collected from different state of the art references.

1.7.4 Patient Classification and Risk Assessment Techniques

Other works in sensing-based healthcare aim to build some techniques for patient classification and risk assessment [46, 47, 48, 49, 50].

In [46], the authors propose an efficient and robust big data analytical platform for real-time sensing-based healthcare applications. The proposed framework relies on Hadoop ecosystem, using data analytical techniques for data analysis and disease diagnosis, and composed of four layers: a layer to control the patient, a layer for data storage and making decision, a layer for patient classification, and a layer for data visualisation. The efficiency of this platform has been comprehensively evaluated through several studies and simulations based on the Hadoop ecosystem, which have shown that the platform significantly improves healthcare applications, in terms of efficiently performing patient classification and disease diagnosis. However, the authors did not consider real scenarios for evaluating the performance.

In [47] the authors built a Fractional Cat-based Salp Swarm Algorithm (FCSSA) based on WBSN, the biomedical nodes collect the data then transmit them to the aggregator, which is selected using the hybrid Harmony Search Algorithm and Particle Swarm Optimization (hybrid HSA-PSO). Then, the deep belief network is trained to classify the vital conditions for the health risk assessment. The algorithm was evaluated through several simulations, and showed promising results in terms of accuracy, energy, and throughput compared to other existing models. However, this operation of classification requires more computational processing time.

In [48], a new smart-fabric based body area sensor network for work risk assessment is proposed. The network contains a smartphone and an artificial intelligent algorithm that aim to determine the criticality level of psychological and physiological work, and a set of sensors integrated into a textile substrate. The solution was evaluated through experiments and showed promising results in terms of physiological signals and physical activity detection. Moreover, this framework needs an accurate validation in a huge dataset collected from workers in real setting scenarios.

The authors of [49] proposed a machine learning-based system for classifying diabetic patients. The framework uses the logistic regression in order to determine the features that characterize the diabetes disease relying on p value and odds ratio. In addition, the authors applied four prediction diabetic patients models such as naïve Bayes, decision tree, Adaboost, and random forest. The authors used US-based National Health and Nutrition Survey data of diabetic and nondiabetic individuals. The obtained results show the accuracy of combination between logistic regression and random forest for diabetes patients classification. Moreover, the framework requires a careful validation in classification with other kinds of medical data.

Reference	Year	Techniques	Performance Evaluation	Limitation
[37]	2018	Fuzzy inference system	Predicting stress level	Security and privacy
[3]	2016	Anova Fisher test Decision matrix Fuzzy set	Emergency detection Adaptive sampling rate Energy consumption	Predicting the progress of the patient situation
[39]	2021	Mobile app Web interface Cloud database	Real-time monitoring Security for data collection	Android not supported
[40]	2019	Machine learning Edge computing Apache flink Data analytic pipeline	Detecting falls in real-time	Does not support all the types of sensors Does not support parallel data processing pipelines
[41]	2020	Sending abnormal records Adapting the sensing frequency Long short-term memory	Reducing data transmission rate Right medical intervention Predicting the patient's situation Saving network lifetime	Repeated collision
[38]	2019	Fuzzy inference system Early warning score systems	Controlling the situation of patients Reducing energy consumption Extending network lifetime	Did not consider real scenarios
[42]	2018	Collecting physiological data Supporting physical activities	Real-time monitoring	Did not implement WBSN module for deploying within Ubiquitous Computing Environment
[43]	2017	Model checking approach Model transformation	Controlling the situation of patients	Scalability using huge WSN system
[44]	2011	Java-programmable sunspot sensor platform	Monitoring human activities Ensuring accurate status of patients Notifying patients of their health status	Not capable of processing in deployed computers
[45]	2018	Data mining		Not able to detect the risk level alerts in the diagnostic system

Table 1.3 Real-time patient monitoring in sensor network: a state of the art techniques.

The authors in [50] developed a classifier for breast cancer on histopathologic images. The authors used a pre-trained VGG-16 model and a fine-tuned AlexNet in order to extract features, which were then classified using a support vector machine (SVM). Preliminary and encouraging results are shown in terms of accuracy. However, the model must be evaluated in a large data set to verify its accuracy.

Table 1.4 shows a comparison analysis of the mentioned techniques in terms of the used methods, the performance evaluation metrics, and the limitations. Note that the different limitations mentioned are collected from different state of the art references.

1.7.5 Nurse Scheduling Techniques

Patient scheduling plays an important role in organizing time and allocating resources such as nurses and machines to perform operations and provide services efficiently [51, 52, 53, 54, 55, 56, 57, 58, 59, 60, 61].

The authors in [51] proposed a model based on a stochastic Petri nets that aims to specify the workload in order to fix the daily number of available appointments. In addition, it aims to minimize the waiting time of patients by scheduling the patients, using a genetic algorithm. Numerical results show that patients' waiting time decreased by using the proposed model. Moreover, the model don't take into account the preferences of nurses for choosing their days off.

In [52], the authors propose an algorithm for non-emergency outpatient appointment scheduling, which aims to reduce the costs through the effective use of expensive human resources and medical devices, and enhance the wait time management of hospitals. This framework determines the duration of every single appointment, and takes into account the priorities of patients. To optimize this model, a genetic algorithm is designed where the procedure based on crossover and mutation operators. The efficiency of the proposed model and GA was evaluated through several numerical experiments, which showed promising computational results in terms solution quality and computational times in most cases compared with MLIP. Moreover, the framework don't take into account the stochastic events such as patients' no-show and machine breakdown, which are inevitable aspects of an appointment scheduling system.

In [53], the authors propose an efficient model for patients' scheduling in emergency department, by using the Markov decision process, then designing the algorithm of the deep reinforcement learning based on deep Q-networks. Their objective is to minimize the crowdedness in hospital and the weighted waiting time of the patients. Several experiments were carried out to investigate the performance of the proposed model. Computational results confirmed that the presented deep RL can outperform the dispatching rules in terms of minimizing the weighted waiting time of the patients and the penalty of emergent patients in the suggested scenarios. However, this study was not addressed in a multi-agent RL with other factors such as operating rooms.

The authors of [54] aim to ameliorate the efficiency of the emergency department by proposing an algorithm for patient scheduling. They built a mixed integer linear programming, abbreviated MILP, that aims to reduce the patients' waiting time and distributed the patients into four classes based on ILOG CPLEX Optimization Studio. The program has been applied to a real case study. The proposed approach is shown to outperform the current configuration in terms of decreasing the patients' waiting time. Moreover, the numerical results show that the model is unable to find a solution when the number of patients is more than 17.

Genetic algorithms have been applied in [56] to ameliorate nurses' scheduling problem-solving time in an effort to reduce the cost of allocating nurses to inferior skill levels. Through a comprehensive

Reference	Year	Techniques	Performance Evaluation	Limitation
[46]	2020	Hadoop ecosystem Data analytical techniques	Patient classification Disease diagnosis	Did not consider real scenarios for evaluating the performance
[47]	2020	Harmony Search Algorithm Particle Swarm Optimization Deep belief network	Accuracy Energy Throughput	Computational processing time
[48]	2020	Artificial intelligent algorithm	Detecting the physiological signals and physical activity Determining the criticality level of psychological and physiological work	Need an accurate validation in a huge dataset collected in real setting scenarios
[49]	2020	Logistic regression Naïve Bayes Decision tree Adaboost Random forest	Classifying the diabetes patients	Need a careful validation in classification with other kinds of medical data
[50]	2018	Pre-trained VGG-16 model Fine-tuned AlexNet Support vector machine	Classifying the breast cancer patients	Must be evaluated in a large data set to verify its accuracy.

Table 1.4 Patient classification and risk assessment in sensor network: a state of the art techniques.

simulation analysis, the authors demonstrated that the suggested method created a nurse scheduling faster in time and better in quality compared to the traditional genetic algorithm. Moreover, the method needs a careful validation with real hospital data with more constraints and diversity of requirements.

The authors of [57] proposed a mathematical programming model and a meta-heuristic algorithm based on simulated annealing, works on maximizing nurses' preferences for work shifts and weekends. It considers the work laws and regulations and the hospital policy as well. This algorithm presents superior outcomes when compared to the head nurse's programs. However, the authors don't take into account the different skill levels for nurses, in addition the priorities for senior nurses.

On the other hand, lots of techniques have been proposed in the literature to optimize the nurse workloads. The authors of [58] have provided a decision-making system for optimizing the nurse's patient assignment in order to reduce job overload. A linear program solved on CPLEX was used to model the problem. Then they use a stochastic programming model to handle the problem and illustrates how it can save up to 273 hours of nurse overload each year in each medical/surgical unit. Moreover, this model does not take into account the preferences of nurses for choosing their days off.

An optimization model is proposed in [55] that relies on the goal programming method to generate the ideal schedule by means of the Lingo 11.0 software. This model has met all the scheduling conditions which were previously unmet with the manual scheduling method; such as the minimum number of male nurses or senior nurses required per shift and the balancing of the working days of all nurses. The efficiency of this model has been comprehensively evaluated through several studies and simulations, which have shown that it significantly provides better solution for regular nurses and better assignment patterns for all nurses. However, it has not been achieved for senior nurses, due to meeting the requirements of senior nurses in each shift. In addition, this model does not take into account the preferences of nurses for choosing their days off.

The authors of [59] designed a two-stage heuristic algorithm for generating a nurse schedule with a balanced weekend workload. The algorithm takes into account the individual nurse's preferences, patient volume variability, required patient-to-nurse, and other work and rest rules. The paper conducted a series of experiments using measurement data from US hospital, and the obtained results verify the importance of the investigation of uncertain impacts. In addition, the results verified that good shift design can lead to revenue savings and fairness. Moreover, the authors don't take into account the different skill levels for nurses.

[60] interviews were performed and evaluated applying the Patient Safety System Engineering Initiative model's qualitative content analysis method. The findings demonstrate what nurses consider to be a source of fatigue, as well as factors that are helpful and harmful in coping with fatigue in their work environment.

The nurses are working under extreme duress owing to the COVID-19 pandemic, and they are doing it benevolently all around the planet. Nurses who are responsible for caring for COVID-19 patients 24 hours a day may become more exhausted as a result of this emergency circumstance. As a result, nurse scheduling should be adjusted to account for the new scenario.

In [61], a new mathematical model for Nurse Scheduling Problem (NSP) considering the fatigue factor is proposed. The authors developed a hybrid Genetic Algorithm (GA) in order to generate a nurse schedule for all three shifts of a day. The authors conducted a series of experiments, in a

real case study comparing the results with manual schedule, in order to show the applicability of the proposed approach. Numerical results verify that this approach outperforms the current manual scheduling in both fatigue factor and time of finding a time schedule. However, it imposes a few more expense on the department.

The nurse's schedules must adhere to a number of constraints. The scheduling must respect each nurse's working time, taking into account the maximum working hours, the minimum break duration, and the maximum number of consecutive work days. The limits can be established for each nurse, such as workload, days off, and so on; it must also take into account the nurses' preferences for teams and vacations [62, 63, 64].

The authors of [62] concentrate on constructing a socially problematic nursing schedule. This latter should consider mainly the following: three shifts, suitable placement of experienced personnel, fairness of job assignment, and legal work standards. As perplex as the nurse scheduling is, the nurses in charge strive relentlessly to set the schedule manually and ensure fairness and legitimacy. In contrast, the automated nurse schedule necessitates considerably less time and effort while respecting the legal work standards and allocating fair distribution of shifts. These automated systems rely on I/O Q-Learning algorithm using Python and a Web Application. Moreover, it imposes some additional costs on the hospital.

In order to assure the effectiveness of the hospital operation, an efficient and practical method of drafting nurse schedules is required. The constraint programming technique was utilized to solve the problem of nurse scheduling in [63]. In terms of the results achieved after numerous tests on various numbers of nurses, this technique demonstrates its capacity to discover an optimal Nurse Schedule Problem (NSP) solution with a higher computational footprint. The performance of the proposed depends on the number input and the present computational resources, taking into account the both hard and soft constraints such as hospital requirements, nurses preferences,.. However, it imposes a more cost to the hospital.

The goal of the study in [64] is to create a system that may be used to organize a nurse's schedule. The obtained working schedule is examined in light of the limits that were imposed. Simulated Annealing was utilized in conjunction with the Probabilistic Cooling Scheme's cooling approach to check the constraint falsification's value. Transitional rules make use of a cost matrix to create a new, more efficient state. Results approve that while the PCS cooling methods combined with the transition rules of the cost matrix imposes a little more cost to the department, it outperforms the the cooling method exponential and logarithmic in generating objective function value of new solutions better and faster in processing time. In addition, this system generates a work schedule in a better quality than the schedules generated manually by the head of the room.

Table 1.5 shows a comparison analysis of the mentioned techniques in terms of the used methods, the performance evaluation metrics, and the limitations. Note that the different limitations mentioned are collected from different state of the art references.

1.7.6 IoT in Healthcare Applications

The Internet of Healthcare Monitoring Things describes the connected infrastructure of hardware, software, medical devices, and services in order to collect and analyze data supporting by this way the decision-making by healthcare professionals. Subsequently, the researchers have payed a great attention in order to improve the integration of the IoT in the healthcare sector [65], [66], [67], [68]. Indeed, many IoHT system have been prove the positive effect at the level of low-cost remote diagnosis [69].

Reference	Year	Techniques	Performance Evaluation	Limitation
[51]	2020	Stochastic Petri nets Genetic algorithm	Specifying the workload Minimizing the waiting time of patients	Don't take into account the preferences of nurses
[52]	2020	Genetic algorithm	Reducing the costs	Don't take into account the stochastic events
[53]	2020	Markov decision process deep reinforcement learning	Reducing the waiting time Minimizing the crowdedness Minimizing the weighted waiting time	Not addressed in a multi-agent RL
[54]	2017	ILOG CPLEX	Decreasing the patients' waiting time	Unable to find a solution when the number of patients is more than 17
[56]	2014	Mixed integer linear programming Optimized genetic algorithm	Reducing the cost	Needs a careful validation with real hospital data
[57]	2015	Mathematical programming model Meta-heuristic algorithm Simulated annealing	Maximizing nurses' preferences	Don't take into account the different skill levels for nurses and the priorities for senior nurses.
[58]	2006	Linear program solved on CPLEX Stochastic programming model	Reducing the nurse overload	Does not take into account the preferences of nurses
[55]	2021	Goal programming method	Balancing of the working days of all nurses.	Does not take into account the preferences of nurses
[59]	2017	Lingo 11.0 software Heuristic algorithm	Revenue saving and fairness	Don't take into account the different skill levels for nurses
[61]	2021	Mathematical model Hybrid Genetic Algorithm	Reducing the time and effort in finding a time schedule	Expensive
[62]	2021	I/O Q-Learning algorithm Web Application	Reducing the time and the effort in finding a time schedule	Expensive
[63]	2019	Constraint programming technique Simulated Annealing	Generating better and faster the time schedule	Expensive
[64]	2018	Probabilistic Cooling Scheme's cooling Transition rules of the cost matrix	Generating better and faster the time schedule	Expensive

Table 1.5 Nurse scheduling in sensor network: a state of the art techniques.

The authors of [70] give an overview of the Internet of Healthcare Things (IoHT) providing an accurate service in tracking the patient's condition in order to prevent any critical situation. In [71], the authors explain the important role of IoT system during COVID-19 pandemic, by validating the efficient transmission of patient's data without any human interaction.

1.8 Conclusion

In this chapter, we presented a general overview about wireless body sensor networks technology that includes network architecture as well as the different types of biomedical sensors. Then, we demonstrated the importance of the integration of internet of thing in the monitoring healthcare applications. In addition, we have described the challenges imposed in the wireless body sensor networks. Finally, we presented a background of several techniques that works on the mentioned challenges, where we discussed their advantages and their weakness. In the next chapter, we will present a comparative study for the recent advances in Internet of healthcare monitoring, using Friedman's statistical test. In addition, an optimized intra WBSN communication (OIC) is proposed in order to ensure the efficient transmission of critical data and to extend the network lifetime.

2 An Adaptive Intra-WBSN Communication Approach

2.1 Introduction

Healthcare requires the cooperation of many administrative units and medical specialties. The Internet of Things (IoT) is involved into healthcare field and plays an extremely important role by providing healthcare services. In the Internet of Healthcare Things (IoHT) several challenges appeared in terms of limited battery life, long processing time, large amounts of collected data, packets overhead on sink,...etc. A lot of research studies have been done in order to improve the healthcare IoT based applications. However, these proposed systems have been focused on some specific purpose without ensuring an effective solution for all problems.

In this chapter, we have attempted to achieve two main objectives: First, a comparative study between the recent advances data collection and communication method in IoHT. This study is based on several parameters namely as follows:

- Energy consumption: Reducing the energy consumption is one of the important challenges in WBSN systems, which aims to extend the network lifetime and therefore to extend the patient monitoring operation.
- Redundant data: Reducing the redundant data, transmitted from the sensors to sink, is essential to improve the decision making operation.
- Overhead on sink: This parameter depends on the number of packets received by the sink from the sensors. It is necessary to reduce the overhead to facilitate the decision making.
- Execution time: This parameter describes the time required to transmit the data in one period for each studied approach.

And this study aims to determine the shortness of each method. All these methods fail to overcome the all overmentioned challenges. Second, we propose an optimized intra WBSN communication (OIC) that is based on merging of a multi-hop routing protocol with the data transmission process of WBSN. The objective of our approach is to ensure the efficient transmission of critical data and to reduce the energy consumption of embedded medical devices, by determining the optimal route toward coordinator, according to the patient situation.

The rest of this chapter is organized as follows. Section 2.2 presents the comparative study of four recent state of the art method in IoHT. Section 2.3 details our proposed transmission approach. The simulation results are evaluated in Section 2.4. Finally, Section 2.5 summarizes this chapter.

2.2 Comparative study of recent advances in data transmission in IoT

This section presents a comparison study of four state-of-the-art IoHT systems. The comparison is based on following parameters: the average lifetime of embedded sensors, the average transmission time, the average overhead rate on the sink, and the average data redundancy reduction. The following articles are selected based on specific criteria: they are recent works, published in well-known journals, and they deal with similar challenges with our work.

2.2.1 An Energy-Efficient Routing Protocol (EERP) for Reliable Data Transmission in WBSN [1]

The authors aim to ensure a reliable and efficient routing transmission of data and also to balance the energy consumption by prolonging the lifetime of the WBSN. The architecture of WBSN, which is used in this framework, is composed of 3 layers. First layer is composed of several biosensors that aims to collect different vital signs from the body of the patient. Second layer is composed of several electronic devices that aims to collect the data from the sensors. Third layer is composed of several remote servers that aims to analyse the data received from the second layer for decision making process. The authors of this approach propose a routing protocol with multi-hops. The steps of the protocol is described in algorithm 1. For each embedded sensor, first, the status of each captured measurement is determined as abnormal or normal. Then, the benefit function M is calculated for each neighbor sensor n_i according to equation 2.1.

$$M_i = \alpha \times Y_{i1} + \beta \times Y_{i2} + \theta \times Y_{i3} + \gamma \times (1 - Y_{i4}) \quad (2.1)$$

Where Y_{i1} depends on the initial energy and the remaining energy of a sensor, Y_{i2} is the transmission efficiency that depends on the number of received packets, Y_{i3} depends on the available bandwidth, Y_{i4} depends on the number of hops to reach the sink. α , β , θ and γ are the weights for Y_{i1} , Y_{i2} , Y_{i3} and Y_{i4} respectively. Finally, the sensor n_i selects the next node that has the maximum value of benefit function M_i , in order to send the measurement to this selected node.

The weights in the benefit function are distributed into two bins and fixed according to the patient's health status:

- If the status is normal: α takes the big value, reducing by this way the energy consumption.
- If the status is abnormal: β , θ and γ take the big value, ensuring an efficient and fast transmission.

Algorithm 1 EERP Algorithm.

Require: sensor node n_i , neighbor table: NT.
Ensure: best next hop node.

- 1: **if** n_i has packet to send **then**
- 2: **if** level is normal **then**
- 3: Select the weights for normal level
- 4: **for** each neighbor node n_j in NT **do**
- 5: Calculate the benefit function M_j
- 6: **end for**
- 7: Select the node has maximum M
- 8: **else if** level is abnormal **then**
- 9: Select the weights for abnormal level
- 10: **for** each neighbor node n_j in NT **do**
- 11: Calculate the benefit function M_j
- 12: **end for**
- 13: Select the node has maximum M
- 14: **end if**
- 15: Send the measurement to the selected node
- 16: **end if**

The paper conducted a series of experiments using MATLAB platform, comparing the results with existing solutions in the literature. And the obtained results showed that the proposed algorithm significantly improves the performance of routing, in terms of network throughput, network lifetime, energy efficiency and reliable transmission of emergency data. Moreover, the values of weights in the benefit function $\alpha, \beta, \theta, \gamma$ are fixed. In addition, this multi-hops routing can't handle the overhead on the sink and the big rate of redundant collected data from the biosensors.

2.2.2 RK-Energy Efficient (RKEE) Routing Protocol for WBSN [2]

The authors propose a routing protocol in order to extend the lifetime of the WBSN, which their steps are described in the algorithm 2 . Similarly to the EERP approach [1] this protocol determines first if the measurement is normal or abnormal. If abnormal, the sensor transmits the measurement directly to the sink using single hop. If normal, this sensor transmits the measurement to a forwarder node. The forwarder node is responsible of collecting the normal data from the sensors and transmit them to the sink. This forwarder node n_i is selected among all the sensors, on each round, according to the CF_i value (equation 2.2). CF_i depends on the distance d_i between the sensor and the sink, and depends on the remaining energy RE_i of this sensor. The sensor having the minimum CF_i is selected to become the forwarder node, means that the sensor node which has a minimum distance from the sink and maximum residual energy, it has the highest chance of becoming the forwarder node. In order to balance the remaining energy of all the sensors in the WBSN, and therefore to extend the network lifetime.

$$CF_i = \frac{d_i}{RE_i} \quad (2.2)$$

Algorithm 2 RKEE Algorithm.

Require: Set of sensor nodes $N=n_1, n_2, \dots$

- 1: **for** each round **do**
- 2: **for** each sensor n_i in WBSN **do**
- 3: Calculate the CF_i
- 4: **end for**
- 5: Select the n_i that has minimum CF_i to be the forwarder node
- 6: **for** each sensor n_i has packet to send **do**
- 7: **if** level is normal **then**
- 8: Send the data to the forwarder node
- 9: **else if** level is abnormal **then**
- 10: Send the data to the sink
- 11: **end if**
- 12: **end for**
- 13: **end for**

The solution has been evaluated through experiments and has shown promising results in terms of energy consumption. Moreover, the proposed algorithm don't take into account the huge amount of the redundant data collected from the patient.

2.2.3 Self-Adaptive Data Collection and Fusion (SADCF) for Health Monitoring in WBSN [3]

The authors aim to optimize the data transmission and to estimate the detection frequency in real time in order to reduce the energy consumption. First, the authors proposed a Modified Local Emergency Detection. This method tries to reduce the data sent by each sensor to the sink and thus save energy. The sensor sends periodically the measurement to the coordinator only in case of criticality change level of the patient, as described in algorithm 3 (lines 2-15). Second, a distributed Adaptive Sampling (AS) algorithm is proposed (lines 16-21). This algorithm allows to change the sampling frequency on each round, in order to control the sensing and therefore to minimise the energy consumption. Consequently, the authors apply ANOVA model with fisher test in order to determine the variation level in sensing. Then, the sampling rate is balanced based on the sensed data variation and the patient criticality.

Algorithm 3 SADCF Algorithm.

Require: Threshold: F_t .

- 1: **for** each round **do**
- 2: **for** each period **do**
- 3: **if** it's the first slot **then**
- 4: Get first measurement r_0
- 5: Send first measurement r_0
- 6: $s =$ score of r_0
- 7: **else**
- 8: Get measurement r_i at slot i
- 9: $s_i =$ score of r_i
- 10: **if** $s_i \neq s$ **then**
- 11: Send measurement r_i
- 12: $s = s_i$
- 13: **end if**
- 14: **end if**
- 15: **end for**
- 16: Compute fisher test F
- 17: **if** $F < F_t$ **then**
- 18: Compute the behavior function BV and adapt the sampling rate
- 19: **else**
- 20: Sampling rate is balanced to the maximum
- 21: **end if**
- 22: **end for**

2.2.4 Sleep Scheduling in Energy Harvesting (SSEH) WBSN [4]

The authors present a three-tiered sleep scheduling approach, providing a new path to further reduce energy consumption in EH-WBNAs, i.e. a combination of sensor sleep scheduling and energy recovery. In order to maximize the lifetime of the low-latency network.

- First level: Sensor Node Scheduling, aims to activate the minimum number of sensors, in order to minimize the network's communication. The authors develop the global and local methods.
 - Global method: It works each round, first it adds to the initial active sensor group the sensor with the largest number of neighbors. Then, for each sensor, it determines if the sensor don't have any neighbor in the active sensor group, in order to increase by one a variable called the unscheduled number. After that, the sensor who has the maximum gradient of reducing the overall unscheduled number is elected to be added to the active sensor group. Repeat until all the remaining sensors have at least one neighbor in the active sensor group, except for the active sensors.
 - Local method: For each round, each sensor calculates the number of its unscheduled neighbors. After that, the sensors who have a larger number than their one-hop neighbors are elected and added to the active sensor group. Each sensor selected to be active will send an "active" message to its neighbors. In other hand, sensors that receive an "active" message from one of their neighbors can switch to sleep. Then, in the current round, the active sensors and their neighbors are pruned, while the rest of the unscheduled sensors will be scheduled for the next round. Repeat until all the sensors are scheduled, means there are no unscheduled sensors, the status of each sensor is either active or asleep at last.

- Second level: Active Sensor Group Discovery, aims to find the maximum number of active sensor groups. First, this method converts the original network to a virtual network. The sensor is represented by several virtual sensors. The number of virtual sensors is defined according to the remaining energy of the sensor. Then, it uses the global or local methods in order to build a minimum active sensor group. After that, from the virtual network, it removes all the elected virtual sensors in the active sensor group. Finally, it converts all obtained virtual networks to original networks.
- Third level: Active Sensor Group Scheduling, aims to arrange these active groups in a specific order, in order to run a specific active group every round. In fact, the authors have proposed three scheduling mechanisms: Retentive Scheduling, Round Robin scheduling, and Heuristic scheduling .
 - "Retentive Scheduling": where one active group is enabled to work frame by frame until its minimum energy is lower than a threshold. Therefore, these mechanism rotate among all active sensor groups until the minimum energy of all active sensor groups is smaller than the threshold.
 - "Round Robin scheduling": each active sensor group runs in turn in a circular order until all active sensor groups loose their battery.
 - "Heuristic scheduling": calculates the minimum energy of each active sensor group and the active group with the highest minimum energy is selected to be enabled in a frame. These mechanism continues until no active sensor group remains to work.

Simulations are offered as a potential perspective to show the effectiveness of the approach. Based on the obtained results, "Heuristic scheduling" is the mechanism that ensure the longest life of the system.

2.3 Optimised Intra-WBSN Communication Approach

In this section we show the overall communication/routing process (Figure 2.1). Multi-hop routing protocol is used in our system in order to save the consumed energy on sensor-coordinator and sensor-sensor communication level. First, the routing protocol that is proposed in the EERP approach is optimized by controlling the data transmission based on the patient status. Second, an adaptive model is adopted at the next hop selection level, where the model weights are fine-tuned based on the patient status. Our proposed routing protocol is divided into four stages:

2.3.1 Network Initialization

First, the system is initialized by exchanging hello message between all sensors. This message includes the following information: remaining energy, number of hop toward the sink and transmission bandwidth.

2.3.2 Emergency Detection

At this stage, each sensor collects data periodically. The sensor calculates the score of the captured measurement using Early Warning Score (EWS) as shown in Figure 2.2, we show one of the most used EWS guides that is developed in UK and distributed around all the world, called National EWS (NEWS). EWS is a guide that is built based on the vital signs, and it is used by the medical staff within hospital in order to track the criticality level of a patient. For each vital sign, the collected measurement is compared to a normal range in order to calculate a score between 0 and 3; 0 means normal measurement where other values indicate abnormal situation where increasing score of severity when the score increases [72]. After score calculation and in order to reduce the transmission

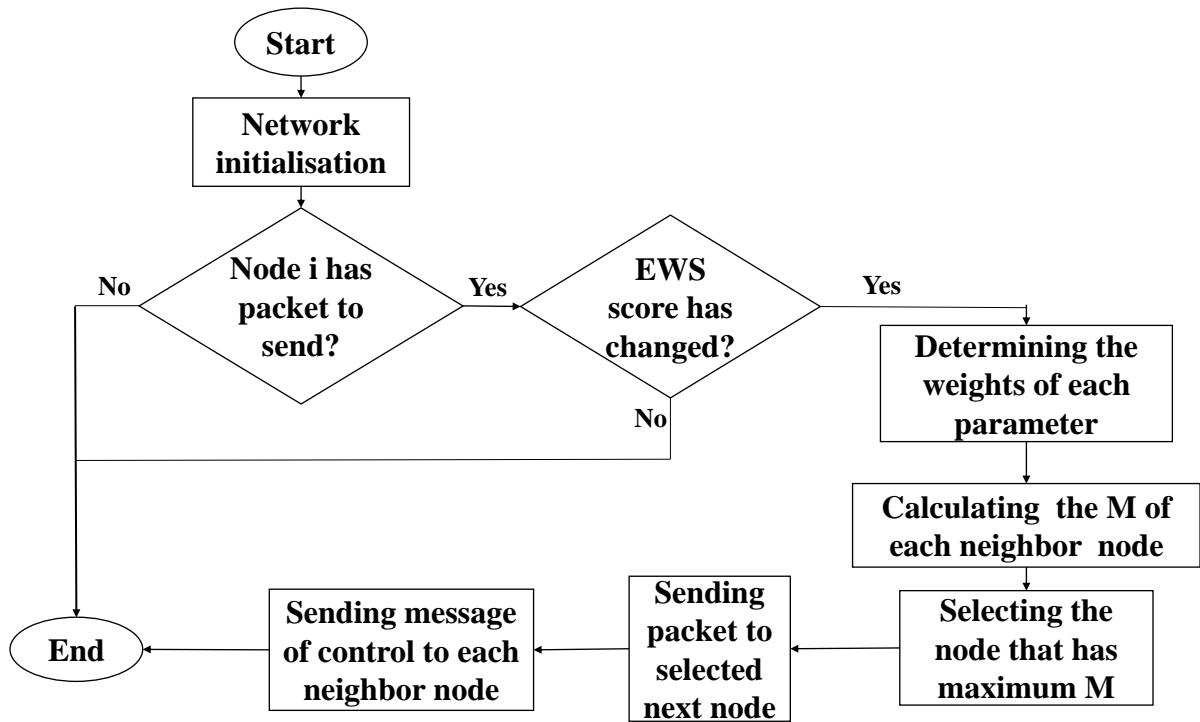


Figure 2.1 Flow chart of the proposed protocol.

of redundant data, the sensor sends the measurement to the next hop only if the patient’s situation has changed e.g. if two successive scores are different. Consequently, the energy consumption is reduced and the decision making operation is become more simple.

PHYSIOLOGICAL PARAMETERS	3	2	1	0	1	2	3
Respiration Rate	≤ 8		9 - 11	12 - 20		21 - 24	≥ 25
Oxygen Saturations	≤ 91	92 - 93	94 - 95	≥ 96			
Any Supplemental Oxygen		Yes		No			
Temperature	≤ 35.0		35.1 - 36.0	36.1 - 38.0	38.1 - 39.0	≥ 39.1	
Systolic BP	≤ 90	91 - 100	101 - 110	111 - 219			≥ 220
Heart Rate	≤ 40		41 - 50	51 - 90	91 - 110	111 - 130	≥ 131
Level Of Consciousness				A			V, P, or U

Figure 2.2 Early Warning Score (EWS) System.

2.3.3 Next Hop Node Selection Based on Benefit function (M)

This stage aims to find the optimal path, by determining the next node for each sensor. The next node is the one that will receive the captured measurement instead of sending it directly to the coordinator. Therefore, the optimal path is the rout that saves most energy, speeds up the transmission and maintains the quality of the communication. The energy consumption is highly dependent on the data transmission which depends on the distance to the destination. The greater the distance is, the greater the power consumption increases. Therefore, the sensor that is near to reach the coordinator, it sends the measurement directly using single-hop communication. In order to find the optimal path, we calculate the benefit function M_{ij} of each node neighbor n_j of the current sensor node n_i . This function includes the following parameters (equation:2.3):

$$M_{ij} = \alpha \times X_j + \beta \times Y_j + \theta \times W_j + \gamma \times (1 - Z_j) \quad (2.3)$$

X_j depends on the initial energy E_j and the remaining energy Er_j of the node neighbor n_j and a minimum threshold E_{min} (equation 2.4).

$$X_j = \frac{Er_j - E_{min}}{E_j - E_{min}} \quad (2.4)$$

Y_j is calculated by equation 2.6. Y_j depends on the efficiency of transmission of node n_j . t_j (equation 2.5) depends on the number of packets that are successfully forwarded by the node neighbor n_j : Pf_j from the received packet: Pr_j . t_{min} and t_{max} are the minimum number and maximum number respectively of packets that are successfully transmitted.

$$t_j = \frac{Pf_j}{Pr_j} \quad (2.5)$$

$$Y_j = \frac{t_j - t_{min}}{t_{max} - t_{min}} \quad (2.6)$$

W_j is the normalized parameter with the available bandwidth Bav_j of the node neighbor n_j . This parameter is calculated according to the equation 2.7, while $Bmin$ and $Bmax$ are the minimum and maximum bandwidth in the WBSN. The higher the bandwidth of the sensor, the more the transmission will be efficient.

$$W_j = \frac{Bav_j - Bmin}{Bmax - Bmin} \quad (2.7)$$

Z_j is the normalized parameter of the number of hops to reach the coordinator H_j of the node n_j (equation 2.8). $Hmax$ denotes the maximum number of hops from the candidate next hop n_j to the coordinator.

$$Z_j = \frac{H_j}{Hmax} \quad (2.8)$$

α , β , θ and γ are the weights for X_j , Y_j , W_j and $(1-Z_j)$ respectively. Moreover, in order to calculate the benefit function, the optimum values of these four weights must be determined according to the patient's status. The patient's status is divided into two cases: normal and abnormal. If the case is abnormal, the focus will be on ensuring the efficient transmission of data. So the weight θ that depends on the available bandwidth will be maximized. Otherwise, if the case is normal, the focus is more on reducing the energy consumption. Therefore, the weight α that depends on the remaining energy will be maximized. This update of α increases the chance of selecting the sensors having the highest remaining energy.

Algorithm 4 Adapting Weights Algorithm.

Require: Initial condition: $\alpha + \beta + \theta + \gamma = 1$, $a = 50\%$, $b = 25\%$, $c = 25\%$, E .

Ensure: Determine the value of each weight.

```

1: for each measurement  $m$  do
2:   Determine the score  $s$  of  $m$ :  $s = \text{EWS}(m)$ 
3:   if level is normal then
4:      $\alpha = \frac{1}{1+e^s}$ 
5:      $E = 1 - \alpha$ 
6:      $\theta = a \times E$ 
7:      $\beta = b \times E$ 
8:      $\gamma = c \times E$ 
9:   else if level is abnormal then
10:     $\theta = \frac{1}{2 \times (1+e^{-s})}$ 
11:     $E = 1 - \theta$ 
12:     $\alpha = a \times E$ 
13:     $\beta = b \times E$ 
14:     $\gamma = c \times E$ 
15:   end if
16: end for

```

The steps of optimization weights is described in algorithm 4. For the initial conditions: the sum of the weights is equal to 1. First, the score s is calculated for the captured measurement using the EWS table (line 1-2). The status of the patient can be normal or abnormal. This status is determined by the doctor, for example a cancer patient has a normal status, but a patient who underwent surgery has a abnormal status. If the status is normal, α is calculated by the equation in line 4, when s is decreased α is increased and the opposite is true. E is the subtraction result of α from 1 (line 5), the largest part (a) of E is for θ (for bandwidth) line 6, and the other parts (b and c) are for other weights (θ and γ). If the status is abnormal, θ is calculated by the equation in line 11, when s increases θ increases and vice versa. And E is distributed in this way (line 13-15), the largest part a is for α (for the remaining energy) and the other parts (b and c) are for other weights (θ and γ). Finally, the next node is the neighbor that has the maximum M .

2.3.4 Data Transmission:

The sensor n_i sends the measurement to the selected next node. Then n_i sends a control message to these neighbors that contains the new remaining energy.

2.4 Simulation Results

In order to evaluate the performance of our approach, several parameters are calculated for the four approaches: the network lifetime, the data transmission time, the overhead on sink and the data redundancy reduction. These approaches are implemented using Java. We used real health data collected from Multiple Intelligent Monitoring in Intensive Care (MIMIC) database of PhysioNet [73]. MIMIC contains data for about 72 patients where recorded on vital signs including Heart Rate (HR), Ambulatory Blood Pressure (ABP), Respiration Rate (RESP), pulse (PULSE) and Oxygen Saturation (SPO2). Every second, the biosensor collects new reading for each vital sign then it sends toward the coordinator for archive purpose. Then, this coordinator forwards this reading to the sink. In this simulation, we used health data are collected from 30 patients. Each WBSN consists of 10 sensors. All sensors are provided with an initial energy level of 0.0005 Joules (J), the maximum

wireless transmission distance is 2 m, the size of transmitted frames of each value is fixed to 64 bits, and two sensors are considered neighbors if the distance between them is less than 0.4 m. The network lifetime is divided into rounds and each round is divided into 10 periods. On each period the sensor collects 500 measurements. The simulation tests are repeated 10 times and all resulted values for the parameters are therefore averaged and presented for 30 patients with different criticality (normal and abnormal) as well. This criticality is determined by the medical team.

2.4.1 Comparison of the Studied Approaches

The table 2.1 presents the calculated values of the several parameters for the four approaches, with both patient's status (normal and abnormal). In addition, it presents the algorithm complexity of each approach that strongly depends on the data transmission time.

parameters	Patient's status	EERP	RKEE	SADCF	SSEH
Lifetime (seconds)	normal	752.3917	338.53	37791.067	2899
	abnormal	704.732	449.13	28588.27	2899
Transmission time(ms)	normal	24.4925	19.6921	21.9591	19.606
	abnormal	20.4437	21.4721	22.835	19.3757
Overhead on sink(packets)	normal	3003.0917	1710.67	1092.67	37
	abnormal	2821.8	3059.2	1499.13	37
Data redundancy reduction(%)	normal	0	0	97.81466	63
	abnormal	0	0	97.00174	63
Complexity	-	O(n)	O(n)	O(n)	O(n)

Table 2.1 Resulted values for the different parameters of four approaches.

We used the non-parametric Friedman test in order to test the difference between the four parameters on the four simulated approaches. The procedure involves ranking each row, then considering the values of ranks by columns. The data table has eight rows and four columns. The rows represent the parameters (lifetime, transmission time, overhead on sink and data redundancy reduction), on both status(normal and abnormal). On the other hand, the columns represent the four approaches to be compared. Two statistical assumptions in Friedman's test:

- The null hypothesis (H0): there is no difference between the simulated approaches and they behave similarly.
- The alternative hypothesis (H1): at least one approach is different from the others.

Table 2.1 is considered as a matrix with b rows (the parameters), k columns (the approaches). The ranks are calculated within each row R_{bk} . The sum of ranks is calculated for each columns (approaches). Friedman test is applied as follow:

- step 1: A is calculated:

$$A = \sum \sum R(X_{ij})^2 = 251.5 \quad (2.9)$$

Where R_{ij} is the rank of approach j for the parameter i .

- step 2: B is calculated, following this equation:

$$B = \frac{1}{b} \times \sum RX_j^2 = 221.75 \quad (2.10)$$

Where b is the number of parameters.

- step 3: T is calculated:

$$T = \frac{(b-1) \times [B - bK(k+1)^2/4]}{A - B} = 5.1176 \quad (2.11)$$

- step 4: The value of FS is determined according to the Fisher's law distribution table, and the smallest value of $\alpha = 0.01$ is toked to have a stricter analysis.

$$FS_{1-\alpha; k-1, (b-1)(k-1)} = 3.4 \quad (2.12)$$

We obtain $T > FS$, so Friedman's hypothesis H_0 is rejected. Therefore, there is one approach that performs better than the others. In order to determine the best approach, the bilateral test is used. This test verifies the difference between two approaches by using a reference critical value C equation 2.13.

$$C = t_{(1-\alpha/2)} \left[\frac{2b(A-B)}{(b-1)(k-1)} \right]^{1/2} = 9.593 \quad (2.13)$$

Where t is a constant value in the distribution table. t corresponds to α , b or k . The bilateral test is applied on all approaches. In this study, the SADCF approach demonstrate the superiority. However, the SADCF approach presents some important limitations such as: the large required transmission time and the high overhead on the sink.

2.4.2 Comparison with our Optimised Intra-WBSN Communication Approach

In this section, the performance of our optimised intra-WBSN communication approach is compared with the four studied approaches. The table 2.2 shows a comparison analysis of the four state-of-the-art techniques in terms of the addressed challenges, used methods, and the performance evaluation metrics. Unfortunately, proposed systems have been focused on some specific purpose without ensuring an effective solution for all problems, such as limited battery life, long processing time, large amounts of collected data, paquets overhead on sink.

The Figure 2.3 shows the lifetime of network with each approach. The obtained results show that the network lifetime is extended more than 50% using our approach compared to the SADCF approach and more than 90% compared to the others approaches, for the two patient's situation normal and abnormal. This results is obtained due to the merging of data transmission technique that reduces the number of data sent with the multi-hop routing protocol that reduces the transmission distance. Subsequently, this reductions reduce the energy consumption. Where, the energy consumption of a sensor node for transmitting n data set to the another node, which is at distance 'D' is calculated in our thesis as follows:

$$E_{TX} = E_{elec} \times n \times \tau \times 64 + \beta_{amp} \times n \times \tau \times 64 \times D^2 \quad (2.14)$$

where τ is the period size, 64 indicates the bit representation of each value, E_{elec} is the energy consumption of a sensor node in its electronic circuitry (usually $E_{elec} = 50nJ/bit$), and β_{amp} represents the energy consumption in RF amplifiers for propagation loss (usually $\beta_{amp} = 100pJ/bit$).

Reference	Title	Year	Challenges				Methods	Performance Evaluation
			limited battery life	long processing time	large amounts of collected data	packets overhead on sink		
[1]	EERP	2019	X				Multi-hops routing protocol benefit function	Energy consumption Extending network lifetime
[2]	RKEE	2021	X				Routing protocol C function	Extending network lifetime
[3]	SADCF	2016	X		X		Data collection technique Adaptive Sampling algorithm ANOVA model Fisher test	Emergency detection Data reduction Energy consumption
[4]	SSEH	2019	X				sleep scheduling approach	Energy consumption
Our	OIC	2022	X	X	X	X	Multi-hops routing protocol Data transmission technique Optimization...	Energy consumption Data reduction Emergency detection Reducing overhead

Table 2.2 State-of-the-Art: A comparison.

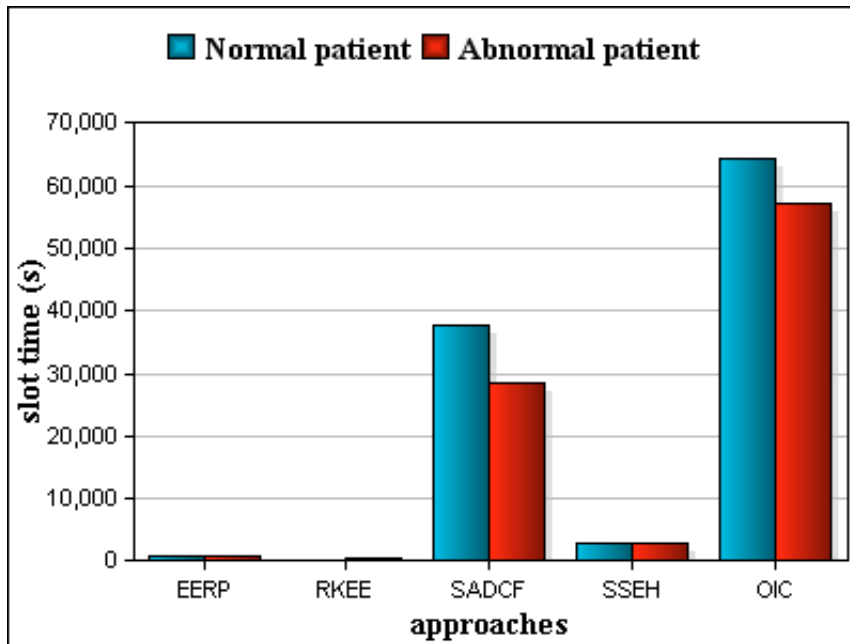


Figure 2.3 Analysis of network lifetime.

Figure 2.4 shows the reduction process of residual energy for the first died node. We can clearly observe that our approach (OIC) efficiently increases the battery life of sensors. This enhancement is obtained due to two reasons: First, each sensor doesn't send a measurement except a patient's situation variation occurs. Second, the sensors that are far away from the coordinator node use the multi-hop communication in order to send the measurements and saving more energy.

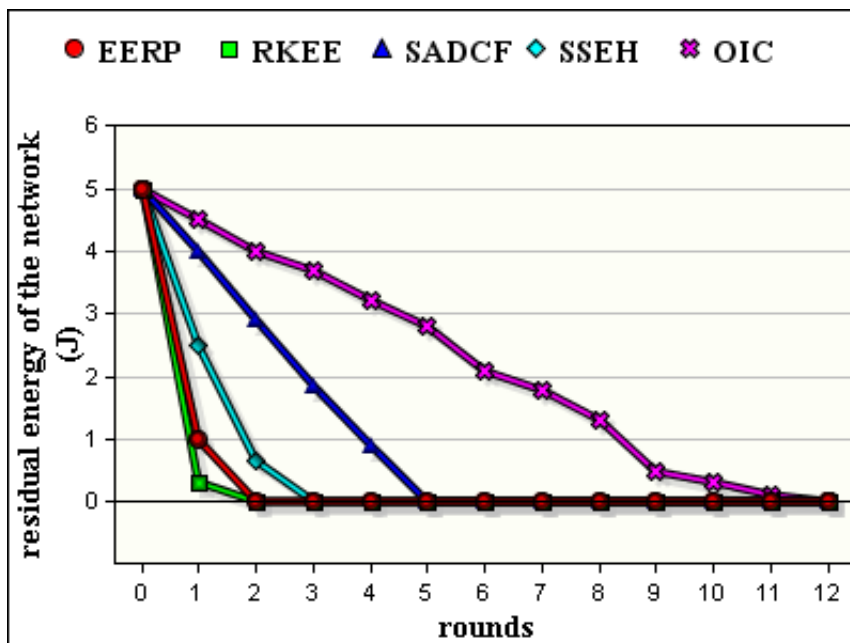


Figure 2.4 Analysis of energy consumption.

In Figure 2.5, the different required transmission time in one period (ms) for each approach are shown. The obtained graphs show that the SSEH approach and our approach (OIC) have less required

time than others for both patient's situation (normal and abnormal).

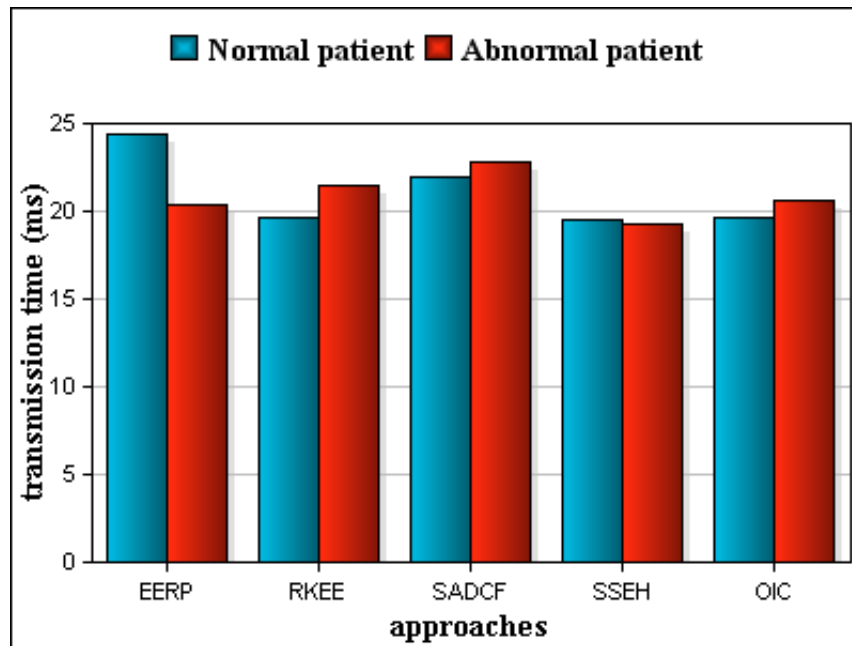


Figure 2.5 Analysis of transmission time in one period.

In Figure 2.6 the comparison of overhead on sink for all studied approaches has been presented. This parameter describes how many packets have been received on the sink in each round. The results show that our approach reduces the overhead on sink up to 83%, 69% and 59% compared to EERP, RKEE, and SADCF approaches respectively in the normal situation, and up to 79%, 82% and 63% respectively in the abnormal situation. This enhancement is obtained due to the low number of sensors that send the data directly to sink. The SSEH approach reduces the overhead on sink more than our proposed. This result is interpreted by the sleep scheduling mechanism that is applied in this approach. Where, this mechanism aims to reduce the number of active sensors each round and therefore reduce the number of packets sent by the active sensors toward the sink.

Reducing the redundant transmitted data is an important challenge in WBSN, this increases the accuracy of the decision making process. Figure 2.7 shows the percentage of reducing data after applying the compared approaches. The obtained results show that both SADCF and our approach (OIC) reduce the transmission of useless data up to 64% compared to SSEH approach. While EERP and RKEE approaches don't take into account this challenge, the sensors in these networks send all the collected data to the coordinator.

2.5 Conclusion

In healthcare field, the IoT is typically employed in order to increase patient safety and to optimize the medical staff operations. The IoHT systems monitor the person's vital signs in a non-intrusive and effective way. In this chapter, four approaches for patient monitoring in context of IoHT are studied, and an adaptive intra-WBSN communication approach (OIC) is proposed. Our approach aimed to optimize the routing protocol for the data transmission process. By comparing the simulation results on real biosensor data, the efficiency of our approach has been verified in terms of the extension of system lifetime, the adaptation of transmission time, the reduction of overhead on sink, and the elimination of redundancy in data transmission. In the next chapter, we will present an efficient framework called P2D for real-time health-monitoring applications. This solution aims to face

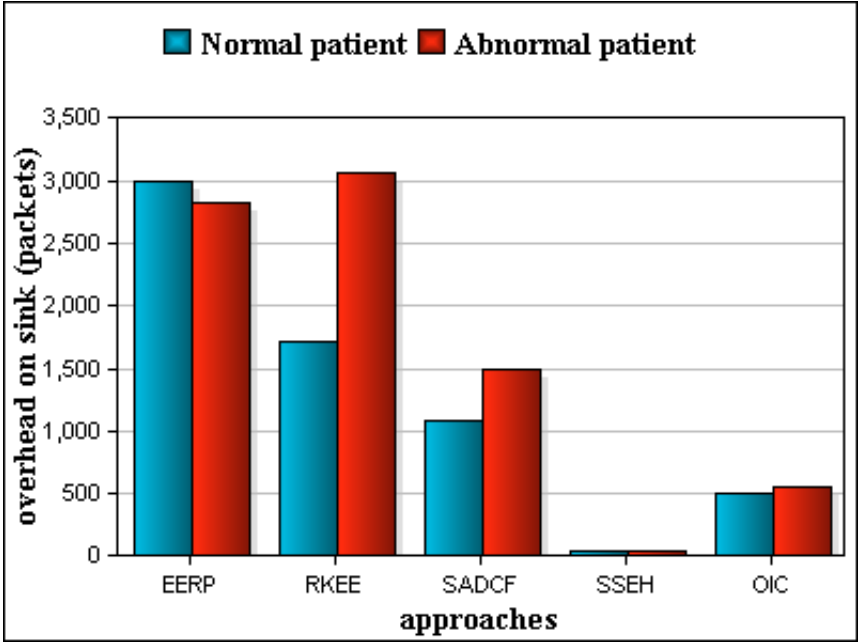


Figure 2.6 Analysis of overhead on sink.

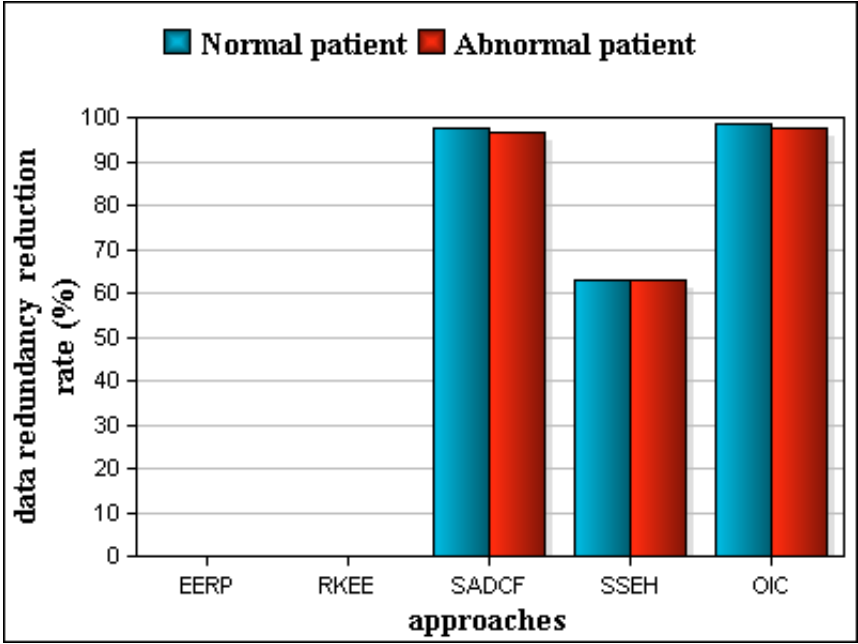


Figure 2.7 Analysis of data redundancy reduction.

several challenges in WBSN such as: emergency detection, energy consumption, reducing redundant data, and decision making.

3 P2D: An Efficient Patient-to-Doctor Framework for Real-Time Health Monitoring and Decision Making

3.1 Introduction

Nowadays, the fastest growth in the big data applications is occurring within wireless body sensor networks (WBSN). In addition to its big size, researchers face two other main challenges: first, the velocity in which data are collected in large speed, sometimes it reaches the level of Terabytes, thus storing such data using traditional servers prone to hardware failures and data loss. Second, the collected data has no structure and takes several formats like numerical, images, videos, etc. This makes the relational database systems is not suitable to big data applications. Therefore, in order to overcome these challenges, Hadoop ecosystem has been proposed as an efficient platform to handle big data applications.

WBSN used in healthcare applications allows to monitor the vital signs (like heart rate, oxygen saturation, blood pressure, etc.) of a patient (e.g. numerical data), to take images of patient organs (like brain, lungs, etc.), to make operations for the patient (like surgery, appendectomy, etc.). Therefore, the periodic and multivariate data collection along with the densely deployment of the sensors makes WBSN as one of the biggest data contributors in this era.

In this chapter, we aim to allow a near communication between patient situations and doctors by proposing an efficient framework for health monitoring and decision making. Our proposed framework is called Patient-to-Doctors (P2D) and works on both sensor and sink levels. This efficient framework start from the collection phase to the visualization phase. The proposed framework relies on Hadoop ecosystem and consists of five layers: collection, which is dedicated to the sensor level and the other layers (ingestion, processing, storage, and visualization) are dedicated to the sink level. The first layer, e.g. collection data, collects the data from the sensors of WBSN and sends via the internet to the second layer, e.g. ingestion data, receives data from the WBSN and forwards to the third layer, e.g. processing layer, which in its turn searches missing data before storing them in the fourth layer. The last layer allows the end users to visualize, analyse and understand the stored data. Moreover, our framework uses various Hadoop tools in each layer in order to ensure fast data processing and reliable storage. Subsequently, our contributions in this chapter can be described as follows:

- *At the sensor level:* P2D proposes two algorithms: emergency detection and adapting sensing

frequency. The first one aims to alert the medical team in case of any critical records are detected in order to take a rapid decision. Thus, the sensor only transmits its collected data to the coordinator if the patient's situation has changed, then this coordinator forwards the received data to the sink. The second algorithm seeks to reduce the amount of periodic data sent from the sensor by eliminating data redundancy in order to save its energy.

- *At the sink level:* P2D guarantees an archive of each patient based on the least squares approximation method and two fault detection algorithms (moving average and exponential smoothing), and it allows the medical team to predict the progress of the patient situation during the next periods, based on the prophet method. Thus, the medical team can take preventive actions in order to reduce the risk level on the patient.

The rest of this chapter is organized as follows. In section 3.2, we present our WBSN platform. Sections 3.3 and 3.4 detail the algorithms proposed at the level of sensors and sink respectively. In section 4.4, we discuss the obtained results. Finally, the section 4.5 deduces our work.

3.2 WBSN Framework

In this section, we present the architecture of our framework proposed for the WBSN. First, the sensors constitute the data sources for our framework that will collect various types of data (especially numerical, images and videos) and send them toward the sink node. In its turn, the sink will forward the data to the Hadoop cluster through the Internet. The selection of the Hadoop cluster at the end user has motivated by the high scalability and fast storage needed for the WBSN. Moreover, the architecture of framework consists of 5 layers (collection, ingestion, processing, storage and visualization) where each layer has its set of tools and, sometimes, some algorithms. Figure 3.1 presents the architecture of our framework with all layers which can be, briefly, introduced as follows:

- *Data collection layer:* Two algorithms are applied over the data collected by the sensors, e.g. emergency detection and adapting sensing frequency. Where, they are presented in details in the section 3.3 (Patient Monitoring Model).
- *Data ingestion layer:* This layer consists of ingestion tools (Flume and Spark streaming) and aim to receive data from the sensors (or data sources) and forward to the next layer (e.g. processing layer).
- *Data processing layer:* Once data are sent by the Spark streaming, the processing layer uses the machine learning library in the Spark in order to detect the fault tolerant in the data before sending to the storage layer. First, we apply the patient records archive algorithm based on LSA method in order to regenerate the data, then we use two algorithms (moving average and exponential smoothing) in order to recover the missing data in WBSN, that has been ignored after the adapting rate of sensing process.
- *Data storage layer:* After the preprocessing of the data, the storage layer uses the Hadoop HDFS (Hadoop Distributed File System) in order to store the data into the Hadoop cluster nodes in a distributed manner.
- *Data visualization and prediction layer:* This layer allows the end user and the experts to retrieve the data stored in the Hadoop HDFS, predict the patient situation using prophet method, and make a real time analysis in order to understand the situation of the monitored zone and make suitable decisions. In this layer, we used two main tools: Grafana, e.g. a graphical user interface, and Matplotlib proposed in Python.

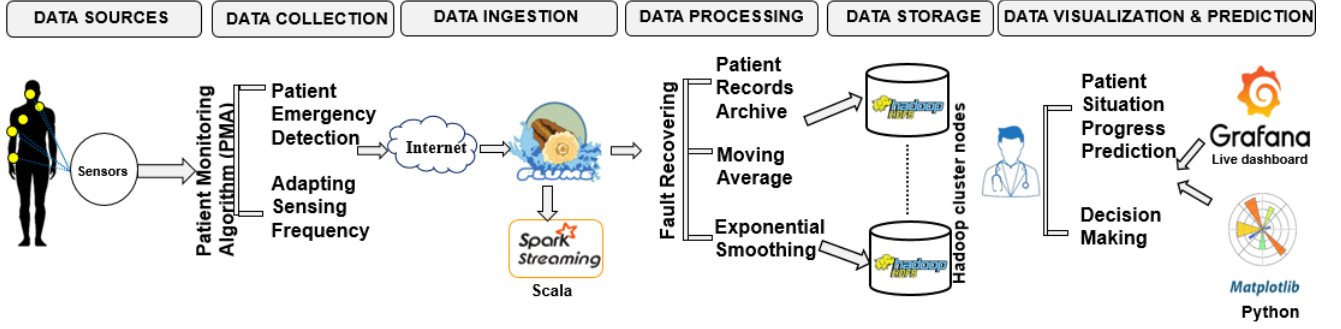


Figure 3.1 Overview of the proposed WBSN architecture.

On the other hand, we are interested on the periodic data collection model in WBSN. In such model, each sensor collects the data for a period of time then it sends the collected data toward the medical team at the end of the period. Assume that, for each patient, we assign a set of NS sensors, e.g. $\mathcal{N} = [n_1, n_2, \dots, n_N]$, in order to monitor his vital signs, e.g. $\mathcal{VS} = [HR, SBP, RESP, SPO2, BT, \dots]$, where each sensor $n \in \mathcal{N}$ monitors the vital sign $vs \in \mathcal{VS}$. For analysis purposes, we assume that the lifetime of each sensor is divided into a set of rounds, $\mathcal{U} = [U_1, U_2, \dots]$, where each round $u \in \mathcal{U}$ consists of a set of \mathcal{P} periods. Then, in each period $p \in u$, each sensor n collects a set of \mathcal{T} readings and it forms a vector ${}_pR_u^{vs} = [r_1, r_2, \dots, r_{\mathcal{T}}]$ that will be sent to the medical team (Figure 3.2).

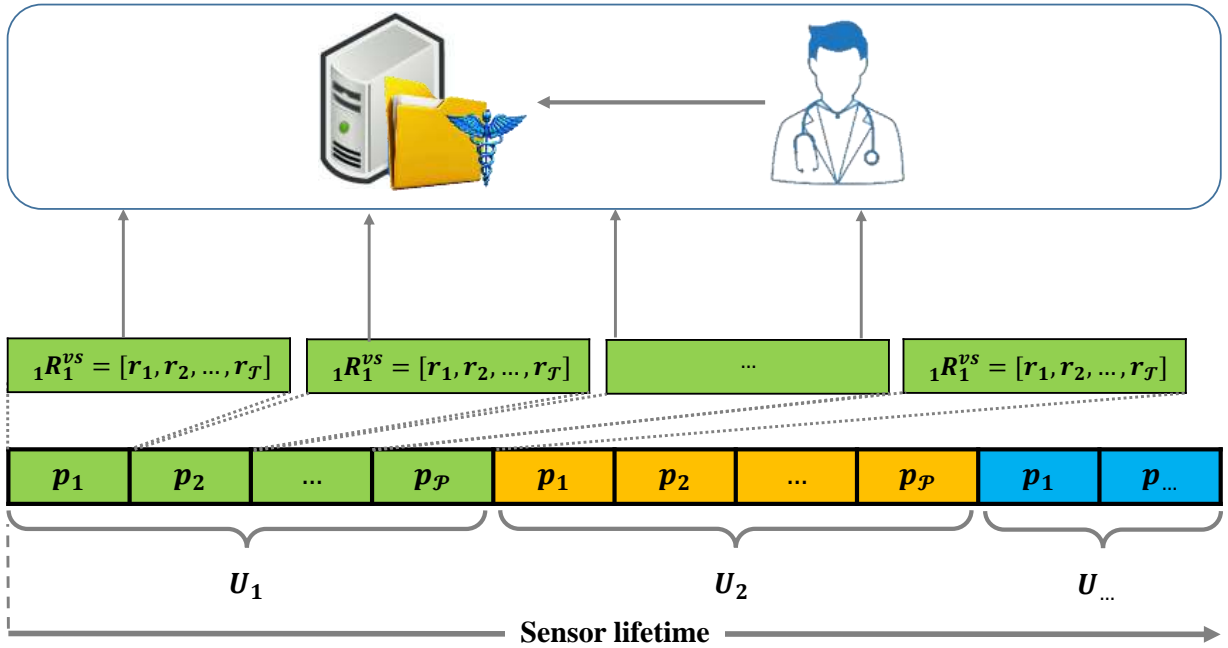


Figure 3.2 Round-based sensor lifetime with periodic collection model.

3.3 Patient Monitoring Model

In this section, we introduce our proposed model at the sensor node level that applied in the collection layer. It aims to ensure an efficient monitoring of patient staying at hospital or in his home (to avoid unnecessary hospitalization and reduce the general healthcare costs). The patient monitoring

model proposed in our platform consists of two major algorithms: emergency detection and adapting sensing frequency. Sometimes, the sensed data can be very critical and requires fast actions to be taken in order to save the person's life thus, the first algorithm, e.g. emergency detection, allows to alert the medical team in case of any abnormal records are detected. Whilst, the adapting sensing frequency algorithm aims to adapt the number of collected data by the sensor in order to save its energy and ensure a long patient monitoring period. In the following, we detail each of the proposed algorithm.

3.3.1 Patient Emergency Detection

In WBSN, there is a need to continuously monitor the variation of the patient's situation. However, the huge amount of data collected and sent by the sensor will quickly deplete its available energy. Thus, in order to ensure a long time patient monitoring, the amount of data transmission from each sensor should be reduced without loss of the information indicating the patient's situation variation, especially in emergency cases. In [3], the authors proposed a modified local emergency detection, abbreviated MLED, algorithm for WBSN by only sending the non-sequential similar records to the sink. Although MLED can highly reduce the data transmission and save the sensor energy however it badly works when the patient situation is stable, e.g. low or high critical situation. Subsequently, in a stable situation, MLED will never send any records from the sensor to the sink which results two challenges: first, the variation of the patient situation will not be noticed by the medical team; second, archiving the patient records will not be completed. In this section, we aim to propose an efficient patient emergency detection algorithm that overcomes the mentioned challenges as well as it saves the energy of the sensor. In order to verify abnormal patient situations, we used the same guide EWS [72] presented in chapter 3.

According to the EWS guide, we propose an efficient patient emergency detection (PED) algorithm that has three folds: first, it reduces the amount of data periodically sent by the sensor; second, it ensures an update information about the criticality of the patient situation; third, it performs an efficient patient archive. The intuition of PED is to search the similarity between successive collected records, based on their scores, then it only sends one record with a score to the sink every time a variation in the patient situation is detected. Furthermore, at the end of each period, PED searches a correlation model among the records collected during this period that will be sent to the sink in order to regenerate and store the collected data for a later analysis and studying. More formally, PED algorithm works according to the following steps and the illustrative example shown in Figure 3.3:

- During each period p , the sensor n collects a set of records about the vital sign vs and forms the set ${}_pR_u^{vs}$. For instance, assume n collects data about the heart rate (HR) and forms a set of 12 readings during p , e.g. ${}_pR_u^{HR} = [90,90,91,99,110,120,123,131,132,125,120,122]$ as shown in the Figure 3.3.
- The sensor n calculates the score of each reading in ${}_pR_u^{vs}$ according to EWS and then it forms the score set ${}_pS_u^{vs}$. Subsequently, ${}_pS_u^{HR} = [0,0,1,1,1,2,2,3,3,2,2,2]$.
- The sensor n sends the first reading collected in p and all readings whose not similar, e.g. have different score values, with the previous sent ones. In Figure 3.3, we show that the first two readings are similar with the same score 0, then n sends only the first reading (e.g. 90); the next readings (91, 99 and 100) have the same score 1 then, s only sends the first one (e.g. 91) and so on. Therefore, a subset of ${}_pR_u^{HR}$, assume ${}_pS_u^{HR}$, will send to the sink instead of the whole set. By doing that, the number of readings sent by the sensor will be reduced while, at the same time, the medical team will be up to date about the variation of the patient situation.
- Finally, at the end of each period, PED studies the correlation between the readings collected in p and generates a prediction model to send to the sink in order to allow to regenerate the

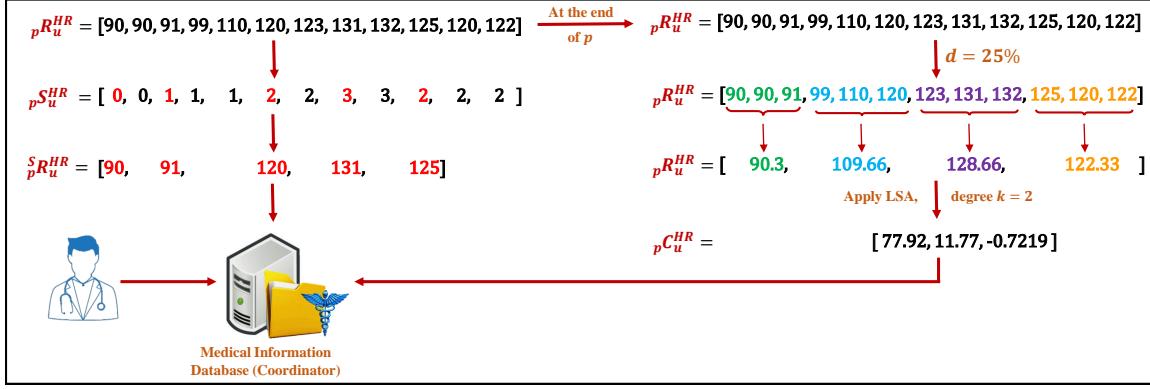


Figure 3.3 Illustrative example of PED.

non-sending readings during p and ensures a patient archive at the hospital. In our work, we are interested in the Least Square Approximation (LSA) method (explained in the next paragraph) which is one of the most standard approaches used in the literature to make statistical analysis. Therefore, LSA allows the sensor n to find the coefficient set of the equation of degree k , e.g. ${}_pC_u^{HR}$, that is best fitting the readings collected during p . Thus, n will send ${}_pC_u^{HR}$ to the coordinator at the end of p which, in turns forward them to the sink in order to maintain the patient situation during his staying in the hospital.

Recall of LSA Method

The method of least squares approximation (LSA) [74] is one of the most standard approaches used in statistical analysis. It aims to determine the curve that best describes the relationship between expected and observed data sets by minimizing the sums of the squares of deviation between observed and expected values. Thus, it limits the distance between a function and the data points that a function is trying to explain. LSA has lots of applications in various domains like healthcare [75], business [76] and finance [77].

Mathematically, LSA can be formally defined as follows:

- Definition 1: (LSA function). Given a set of q data points $\{(x_1, y_1)(x_2, y_2) \dots (x_q, y_q)\}$. The polynomial or parabola of degree k , represented by $y = a_0 + a_1x + a_2x^2 + \dots + a_kx^k$, that can best fit all points (x_i, y_i) , where $i \in [1, q]$, can be calculated according to the following equations:

$$\begin{aligned}
 na_0 &+ a_1 \sum x_i &+ \dots &+ a_k \sum x_i^k &= \sum y_i \\
 a_0 \sum x_i &+ a_1 \sum x_i^2 &+ \dots &+ a_k \sum x_i^{k+1} &= \sum x_i y_i \\
 \dots &+ \dots &+ \dots &+ \dots &= \dots \\
 a_0 \sum x_i^k &+ a_1 \sum x_i^{k+1} &+ \dots &+ a_k \sum x_i^{2k} &= \sum x_i^k y_i.
 \end{aligned}$$

where such $(k + 1)$ equations contain $(k + 1)$ unknowns, i.e. a_0, a_1, \dots, a_k , that should be

calculated in order to find the coefficients of the best fit polynomial to the data points.

Indeed, the processing time needed to calculate the LSA polynomial of degree k will be huge, especially when the period size \mathcal{T} is high. Hence, in order to reduce its time complexity, we propose to reduce the number of taken records when calculating the LSA polynomial by selecting a subset of records from ${}_pR_u^{vs}$ instead of taking all records in the set. Our algorithm works on three steps (see also Figure 3.3):

- We divide the whole set ${}_pR_u^{vs}$ into a number of D divisions where each division D_i contains $d\%$ of the records in ${}_pR_u^{vs}$, e.g. $d\%$ of \mathcal{T} . For instance, if $d = 25\%$ then ${}_pR_u^{HR}$ will be divided into 4 equal divisions as shown in Figure 3.3.
- For each division D_i , we calculate the mean value of the records in the division. Thus, a reduced set of records from ${}_pR_u^{vs}$ will be obtained.
- We calculate the mean value for each division then we apply LSA method in order to find the equation of degree k . Finally, we send the set of equation coefficients, e.g. ${}_pC_u^{HR}$, to the sink at the end of the period. For instance, in Figure 3.3, we find the equation of degree $k = 2$, e.g. $a_0 + a_1x + a_2x^2 = 0$, and we send the coefficient set ${}_pC_u^{HR} = [77.92, 11.77, -0.7219]$ to the sink.

Algorithm 5 describes the process of PED algorithm which is applied at the sensor level at the end of each period. Indeed, PED sends the first collected record to the sink after calculating its score to be the current score (lines 1-3). Then, it calculates the score for each next record in the period and compares with that of the current score; if the scores are the same then the new record will not send to the sink otherwise, the record is sent and its score becomes the new current score (lines 4-10). At the end of the period, PED divides the records collected in p into equal divisions then it calculates the mean for each division and, finally, it finds the set of coefficients that will be sent to the sink, based on LSA method (lines 11-14).

3.3.2 Adapting Sensing Frequency

In WBSN, there is a need to continuously collect the vital signs of the patient in order to better understand his situation as well as to make a suitable decision. This leads to a high level of redundancy between the collected records due to two reasons: first, the stability of the patient situation, especially in low and high critical cases; second, the short slot time between the collected readings. Therefore, the redundancy level produced in WBSN leads to quickly consume the available energy in the sensor and complicate the mission of the medical team in analyzing a huge amount with un-useful information. Hence, adapting the sensor frequency to the dynamic of the patient situation has becoming an essential in WBSN in order to reduce the amount of data collection and send only the useful information to the sink. In this section, we propose an efficient algorithm that studies the variation of the patient situation during a set of successive periods, e.g. a round, then it determines the best sensing frequency of the sensor according to the patient criticalty level.

As mentioned before, assume a round u consists of \mathcal{P} periods, e.g. $u = [P_1, P_2, \dots, P_{\mathcal{P}}]$. Then, for each period $p \in u$, the sensor collects the set of readings ${}_pR_u^v$ thus, during u , a set of readings set is collected as follows: $R_u^{vs} = [{}_1R_u^{vs}, {}_2R_u^{vs}, \dots, {}_{\mathcal{P}}R_u^{vs}]$. Moreover, we assign the set of scores ${}_pS_u^v$ for each set of readings ${}_pR_u^v$. Hence, we define the weight of a score s_i where $i \in [0, 1, 2, 3]$ as follows:

Algorithm 5 PED Algorithm.

Require: A sensor n , A vital sign vs , A period p , Period size: \mathcal{T} , Readings collected during p :

${}_pR_u^{vs} = [r_1, r_2, \dots, r_{\mathcal{T}}]$, Division size: d .

Ensure: Set of values sent to the sink.

- 1: calculate score s_1 of r_1
- 2: send s_1 to the sink
- 3: $current_score = s_1$
- 4: **for** each record $r_i \in {}_pR_u^{vs}$ where $i > 1$ **do**
- 5: calculate score s_i of r_i according to EWS
- 6: **if** $s_i \neq current_score$ **then**
- 7: send r_i
- 8: $current_score = s_i$
- 9: **end if**
- 10: **end for**
- 11: divide ${}_pR_u^{vs}$ into $100/d$ equal divisions
- 12: calculate the mean for each division
- 13: find ${}_pC_u^{vs}$ based on definition 1
- 14: send ${}_pC_u^{vs}$ to the sink

- Definition 2: (the weight of a score s_i , $wgt(s_i)$). Given a set of readings' set collected during a round u . Then, we define the weight of s_i , where $i \in [0, 1, 2, 3]$, as the number of occurrence of the score i during the round u as calculated in the following equation:

$$wgt(s_i) = \frac{\sum_{k=1}^{\mathcal{P}} \sum_{j=1}^{\mathcal{T}} 1_{j=i}}{\mathcal{P} \times \mathcal{T}} \quad (3.1)$$

Based on the score weight function, we define the criticality level, \mathcal{C} , of a patient during the round u according to the equation:

$$\mathcal{C} = \frac{\sum_{i=0}^3 s_i \times wgt(s_i)}{\sum_{i=0}^3 s_i} \quad (3.2)$$

Therefore, the criticality level of the patient can be described as shown in the Table 3.1:

Table 3.1 Description of the patient criticality level.

Criticality level, \mathcal{C}	Description
$0 \leq \mathcal{C} \leq 0.3$	low criticality level
$0.3 < \mathcal{C} \leq 0.6$	medium criticality level
$0.6 < \mathcal{C} \leq 1$	high criticality level

In addition to the criticality level, we take into account another parameter in order to adapt the sensing frequency of each sensor which is the risk level of the patient. This parameter is highly determined by the medical team when the patient enters to the hospital. In our platform, we define two levels for the risk level, \mathcal{R} , depending on the situation of the patient:

- *Low risk*: in which a patient enters to hospital in a stable situation and the collected data about his vital signs are almost normal. In this case, the patient requires minimum patient observations from the medical staff.

- *High risk*: in which a patient enters the hospital in an unstable situation and most of the collected records are different from the normal score range. The patient, in this case, requires a continuous monitoring from the medical team in order to evaluate his situation progress.

Therefore, we adapt the sensing frequency of a sensor according to both mentioned parameters: the criticality level (\mathcal{C}) and the risk level (\mathcal{R}) of the patient. The sensing frequency decision table (abbreviated SFDT as shown in Table 3.2) shows the new sensing frequency of the next round that the sensor should adapt its sampling in order to reduce the amount of data collected, without losing the integrity of the information, thus save its energy. Indeed, SFDT shows two main facts: 1) the new sensing frequency is calculated based on the original period size \mathcal{T} ; for instance, if \mathcal{T} is set to 100 records and $(\mathcal{C}, \mathcal{R})$ is (*low*, *high*) then the sensor should adapt its sensing frequency to 40 records in the next round, e.g. $(40\%/100) \times \mathcal{T}$. 2) the sensing frequency of the sensor is increased with the increasing of \mathcal{C} and/or \mathcal{R} .

Table 3.2 Sensing frequency decision table (SFDT).

\mathcal{C}	$\mathcal{R} \rightarrow$	<i>normal</i>	<i>abnormal</i>
<i>low</i>		20% of \mathcal{T}	40% of \mathcal{T}
<i>medium</i>		40% of \mathcal{T}	60% of \mathcal{T}
<i>high</i>		60% of \mathcal{T}	\mathcal{T}

3.3.3 Patient Monitoring Algorithm

In this section, we aim to integrate the patient emergency detection (PED) and the adapting sensing frequency proposed at the sensor level. The new algorithm is called patient monitoring algorithm and is abbreviated as PMA. Algorithm 6 shows the process of PMA that takes the parameters of PED in addition to the risk level and the SFDT table. First, PMA applies the PED algorithm over the set of readings collected during each period of the round u (lines 1-3). Then, it calculates the weight of each score appeared in the round, based on the equation 3.1, followed by the criticality level of the patient (lines 4-7). Finally, PMA calculates the new sensing frequency of the sensor according to the SFDT table based on the criticality and the risk levels of the patient.

Algorithm 6 PMA Algorithm.

Require: A sensor n , A vital sign vs , A round u , Set of reading sets collected in u : $R_u^{vs} = [{}_1R_u^{vs}, {}_2R_u^{vs}, \dots, {}_\varphi R_u^{vs}]$, Period size: \mathcal{T} , Division size: d , Risk level: \mathcal{R} , Sensing frequency decision table: SFDT.

Ensure: void.

- 1: **for** each reading set ${}_iR_u^{vs} \in R_u^{vs}$ **do**
 - 2: apply PED($vs, \mathcal{T}, {}_iR_u^{vs}, d$)
 - 3: **end for**
 - 4: **for** $i = 0$ to 3 **do**
 - 5: calculate $wgt(s_i)$
 - 6: **end for**
 - 7: calculate \mathcal{C} based on equation 3.2
 - 8: new sensing frequency = SFDT (\mathcal{C}, \mathcal{R})
-

3.4 Medical Team Decision-Making Model

In this section, we introduce the second level of our P2D platform which is dedicated to the medical team at the hospital. Indeed, one of the most challenges that faces the medical team is the analyzing of the huge amount of data generated in WBSN and making a right and fast decision. Moreover, the prediction of the patient progress situation is also a crucial task for the medical team in order to provide a pretreatment and avoid any undesired situation, especially the death. On the other hand, maintaining an archive of each patient is a must in almost hospitals for several purposes, especially disease diagnosis and improving the healthcare quality. In order to overcome these challenges, P2D provides two main services for the hospital and medical team: patient archiving and prediction of patient situation progress. In the next sections, we detail each of the proposed algorithm, that are applied from the ingestion layer to visualization and prediction layer based on hadoop framework (see Appendix A), in order to handle the storing and the processing of the big data and to ensure the fault tolerance.

3.4.1 Data Ingestion Layer

This layer is the first step for the data coming from the sensors in WBSN in which data is prioritized and categorized in order to flow smoothly in the other layers. We used two Hadoop tools in this layers: Flume and Spark streaming.

Flume

It is a distributed and reliable software that aims to efficiently collect, aggregate and transfer a huge amount of data to the Hadoop cluster [78]. In addition, Flume uses batch mode to move streaming data flows and proposes several recovery mechanisms to ensure the fault tolerance. In our framework, Flume receives period data collected by the sensors and deliver to the data collector tool (e.g. Spark streaming). Among other data ingestion tools, Flume has been chosen due to its huge ability in handling data with various types, e.g. the case of WBSN.

Spark Streaming

It uses the fast scheduling capability of Spark Core's in order to perform streaming analytics based on Scala programming [79]. It receives ingested data from Flume and process them in batches before sending to the rest of the data pipeline. In our framework, Spark streaming is selected among other existing tools due to three major characteristics required in WBSN: first, it supports both batch and streaming processing which are necessary to apply various data analytical algorithms (explained in next layer); second, it ensures a lower latency level than other tools like MapReduce, which is strongly required in WBSN, especially in critical application; 3) It guarantees scalability to any number of cluster nodes required for the WBSN application. Therefore, in our framework, we implemented Spark on the master node in order to receive data streaming from Flume, doing processing (in the next layer) and send data to the Hadoop HDFS for storing purpose.

3.4.2 Data Processing Layer

In this section, we introduce the third layer used in our framework which is responsible for pre-processing data before storing them in the next layer. Indeed, data sent by the sensors is usually prone to loss due to many reasons: 1) the long distance between the sensor and the sink; 2) the bad conditions in the monitored environment; 3) the congestion resulted from the densely network; 4) the failure in the sensor itself. In such cases, we need to recover the missing data before any processing in order to ensure the accuracy of the taken decision. We are interested, in this chapter, in the numerical missing data generated from WBSN. Hence, in this layer, we introduce the Patient Records Archive

algorithm, which is implemented based on the machine learning module in Apache Spark, in order to generate the data based on the coefficients of LSA method. In addition, after data generation process, we applied two algorithms Moving Average (MA) and Exponential Smoothing (ES) in order to recover the missing data, which is reduced after applying the adapting sensing frequency algorithm.

Patient Records Archive Algorithm

Patient data archiving is a key operation in hospitals. However, most of the data reduction techniques proposed in the literature for WBSN aim to minimize the amount of data sent by the sensor, in order to save its energy, without taking into account that patient records must be regenerated for archiving purpose [30, 80, 37, 3, 81, 38]. In addition, data collected by the sensors are vulnerable to loss before reaching the sink due to several reasons: 1) a long distance; 2) the congestion due to the overloaded network in case of dense biosensors deployment; 3) the obstacles; 4) the failure in the biosensor itself. In such cases, the medical team will not be able to make the right decision about the patient situation or store missing records for a later analysis. Thus, in order to overcome missing records, a preprocessing of data should be made before any decision or storage process. Algorithm 7 shows the patient records archive (PRA) process proposed in our P2D platform in order to archive the patient history during his stay in the hospital. PRA takes advantages from the sensor records and the coefficient set of LSA equation received at each period. For each slot time in the period, if the sink receives the record from the sensor, which means that situation of the patient is varied, then it directly stores in the patient archive; otherwise, e.g. in case of the record is not sent by the sensor due to the similarity with the previous one, the sink generates the record of the current slot time based on the set of coefficients of LSA method then it adds to the patient archive data.

Algorithm 7 PRA Algorithm.

Require: Period size: \mathcal{T} , Readings collected during p : ${}_pR_u^{vs} = [r_1, r_2, \dots, r_{\mathcal{T}}]$, LSA coefficient set: ${}_pC_u^{vs}$.

Ensure: void.

- 1: $StoredSet \leftarrow \emptyset$
 - 2: **for** $i = 1$ to \mathcal{T} **do**
 - 3: **if** r_i is sent by the sensor **then**
 - 4: $StoredSet = StoredSet \cup \{r_i\}$
 - 5: **else**
 - 6: generate r_i based on ${}_pC_u^{vs}$
 - 7: $StoredSet = StoredSet \cup \{r_i\}$
 - 8: **end if**
 - 9: **end for**
 - 10: store $StoredSet$ at the sink
-

Moving Average (MA) Algorithm

It is a well-known statistical method used to estimate the missing values in time series. Thus, if a value is missed during a slot time y then the average of the previous ws slots' time is calculated and assigned to the missed value. Hence, MA is based on the concept of shifting window of size ws . Particularly, if any value is missed from the window, the first existent future value is assigned to it. The most applications of moving average method are to study the trend direction whether in business, weather, network, etc [82, 83].

Let assume that the sink receives a vector of \mathcal{T} readings from a sensor n_i during \mathcal{T} time slots, e.g. $R_i = \{(y_i, r_i) \text{ such that } i \in [1, \mathcal{T}]\}$. Also assume that MA defines a window of ws readings thus if

10	11	11	?	?	11	?	12
y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
10	11	11	10.66	?	11	?	12
y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
10	11	11	10.66	10.88	11	?	12
y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
10	11	11	10.66	10.88	11	10.84	12
y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8

Figure 3.4 Illustrative example of MA algorithm.

any reading r_j at slot time j is not received by the sink then it will be calculated according to the following equation:

$$r_j = \frac{\sum_{i=j-ws}^{ws} r_i}{ws} \quad (3.3)$$

Figure 3.4 shows an illustrative example of the moving average algorithm for a set of 8 readings collected by a sensor in which 3 of them are missing, e.g. at slots y_4 , y_5 and y_7 respectively. We can also show that the window is of size 3. Then, in order to find the missed reading at slot y_4 , we calculate the average of the readings within the window, e.g. $10 + 11 + 11/3$, and we assign the result value to the slot y_4 , e.g. 10.66. After that, we calculate the missed reading at slot y_5 by first shifting the window to the slots y_2 to y_4 then we calculate the average of reading window. Lastly, the same process is applied for the reading at slot y_7 by calculating the average of the readings window at slots y_4 to y_6 .

Exponential Smoothing (ES) Algorithm

Similar to MA method, the exponential smoothing (ES) is an analysis technique of time series data that is based on the window concept. Whereas in MA the previous observations are equally weighted, ES assigns exponentially decreasing weight for the observations over slots time. Given the set of readings $R_i = \{(y_i, r_i) \text{ such that } i \in [1, \mathcal{T}]\}$ collected by a sensor, the missing reading r_j at slot y_j can be calculated according to ES formula as follows:

$$\begin{cases} Est_1 &= r_1 \\ Est_j &= \zeta \times r_{j-1} + (1 - \zeta) \times Est_{j-1} \end{cases} \quad (3.4)$$

where:

- Est_1 is the estimated reading at slot time y_1 that is initially equal to r_1 .
- Est_j is the estimated value at slot time y_j .
- ζ is the smoothing factor where $0 < \alpha < 1$.

Figure 3.5 shows the same set of readings collected by the sensor in Figure 3.4 but we apply the ES algorithm with a fixed value of ζ equal to 0.5. According to equation 3.4, the first predicted reading at slot time y_1 is the same to the collected reading, e.g. 10. Then, for the predicted reading Est_2 at slot y_2 , it calculated as $0.5 \times 10 + (1 - 0.5) \times 10 = 10$. To calculate the missed reading at the slot time y_4 , r_4 will equal to Est_4 that is calculated as $0.5 \times 11 + (1 - 0.5) \times 10.5 = 10.75$. The same process is performed for the other missing readings.





R_i	10	11	11	?	?	11	?	12
	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8
								
Est_i	10	10	10.5	10.75	10.75	10.75	10.87	10.87
	y_1	y_2	y_3	y_4	y_5	y_6	y_7	y_8

Figure 3.5 Illustrative example of ES algorithm.

3.4.3 Data Storage Layer

This layer aims to store the data sent from the WBSN after applying fault recovering algorithms (MA or ES). It is based on the Hadoop HDFS [?] that splits the received data into small blocks then it distributes across the cluster nodes. Generally, HDFS is characterized by two aspects: first, it ensures the fault tolerance of the stored data thanks to the replication process of data into nodes (at least three). Second, the process of data is performed in a parallel computing which decreases the latency of data when taking a decision. In this work, we implemented our framework on a cluster forming of three nodes where the Hadoop HDFS is installed in each one.

3.4.4 Data Visualization and Prediction Layer

In the last layer of our framework, we are interested in providing an efficient data access and retrieval method to allow the end user to visualize and analyze the stored data. In addition, we implemented a prediction technique based on prophet method in order to predict the patient situation. We propose to use two visualization tools, e.g. Grafana and Matplotlib, where the first one allows a graphical user interface to access the data where the second one is based on the Python programming. The ultimate goal of using each of one is to perform a graphical representation of the retrieved data using statistical graphics and plots. In next sections, we highlight each tool used in our framework.

Grafana

It is a multi-platform open source analytics software that help users visualize and understand trends within vast amounts of data [84]. Grafana is used on top of different data stores, i.e. Hadoop in our framework, and allows the end users to create and modify dashboards for their applications. In addition, Grafana supports a large number of charts and graphs in order to analysis the stored data. Also, Grafana comes with a built-in alerting engine that allows the end users to set conditional rules to the dashboard and sends a notification, using email or slack, when an alert is triggered. With Grafana, our framework offers the following services to the WBSN end users: 1) understanding the data collected by the sensors and stored in the Hadoop clusters using various Grafana plots; 2) create various alerts to notify the end users of any abnormal situation; 3) visualize the metrics of the Hadoop cluster such as system CPU, memory, disk, etc. and scales the cluster when necessary;

Matplotlib

It is the most used plotting library in Python programming with its extension NumPy [85]. In our framework, we created a Python script that allows to read periodic WBSN data sent to Hadoop storage, visualize them, using Matplotlib, on a real-time graph to the end users, and predict the patient situation using prophet method. An important extension which can be added later to our framework is to send plotting graphs to the end users using mobile application.

Prediction of the Patient Situation Progress

In healthcare applications, the ability of medical team to ensure a continuous monitoring and predicating the situation progress is crucial for patient safety. Hence, healthcare forecasting has been

taken a great attention from researchers in WBSN [86, 87]. The main objective of the healthcare forecasting is to provide not only estimates of the current situation of the patient but also forecast to its future situation. This will help the medical team knows when and where there is a risk on the patient and, through this understanding, it can perform preventative actions to reduce the effect of illness and avoid the death. In this section, we propose a prediction algorithm in order to allow the medical staff to forecast the progress of the patient situation and take the appropriate clinical response. Indeed, one can find a lot of prediction techniques that have been proposed in the literature like linear and logistic regression, decision tree, random forest, neural network, etc. These techniques are introduced in various domains such as stock production, scientific studies, sport monitoring, financial sector, psychology, etc. In this chapter, we are focusing on the Prophet method which is recently proposed and not yet adapted to WBSN. Let first recall the Prophet prediction method, then we adapt it to the WBSN case.

- Recall of Prophet: Prophet [88] is an open source tool released by Facebook in 2008 that is implemented in R and Python. Basically, Prophet is a procedure used for forecasting time series data based on an additive model where non-linear trends are fit with yearly, weekly, and daily seasonality, plus holiday effects. In addition, Prophet takes advantage from the correlation between the data in order to handle the missing data and outliers. Recently, Prophet has been used in various domains such as streamflow of river [89], data warehouse [90], business [91, 92] and allows data scientists and analysts to provide efficient forecasting models for a variety of problems in a business scenario. Mathematically, Prophet uses additive regression model with three main components (trend, seasonality and holidays) which are combined in the following equation:

$$y(t) = g(t) + s(t) + h(t) \quad (3.5)$$

where:

- Trend, $g(t)$: is the overall increase or decrease of the series during this time. Prophet automatically detects changes in trend by selecting change points from the data.
 - Seasonality, $s(t)$: is the data component that undergoes regular and predictable changes over a fixed period of time. It is modeled using Fourier series (for yearly seasonal) and dummy variables (for weekly seasonal).
 - Holidays, $h(t)$: is an optional component that models the effect of holidays or major events which is determined by the users.
- Integrating Prophet into WBSN: Although that Prophet has been adapted to large number of domains however, for the best of our knowledge, it is the first that it is integrated in the body sensor networks. Our objective is to study the variation between the collected records about a patient during periods of time then try to forecast the progress of its future situation. As mentioned in section IV, we assume that the lifetime of a sensor is divided into a set of rounds where each round u contains a number of \mathcal{P} periods. We divide the periods in each round into two parts:
 - *Training data periods*: in which periodic datasets collected by the sensor are received by the sink. Assume that at a given period of the round, the sink receives σ datasets from the sensor which will act as training and historical data for the Prophet. Fortunately, more the size of the training data is more the accuracy of the forecasting given by the Prophet becomes.

- *Predicted data periods*: in which the sink has to predict, based on Prophet, the datasets that will be sent for the sensor in the remaining $\mathcal{P} - \sigma$ periods. For every new dataset received by the sink, the training data size will increment by one and the prediction process will be more accurate.

Figure 3.6 shows an illustrative example of how to apply Prophet forecasting in our platform with a round set to 5 periods. When the sink receives the dataset at the first period, e.g. $\sigma = 1$, it applies the Prophet in order to estimate the datasets of the remaining 4 (e.g. $5 - \sigma$) periods. Then, when the second dataset is received (e.g. $\sigma = 2$), it applies the Prophet to forecast the remaining 3 periods and so on. Indeed, in our platform, we omitted the holidays component from the equation 3.5 and we considered the trend as the variation within each training data while seasonality represents the variation between the training periods.

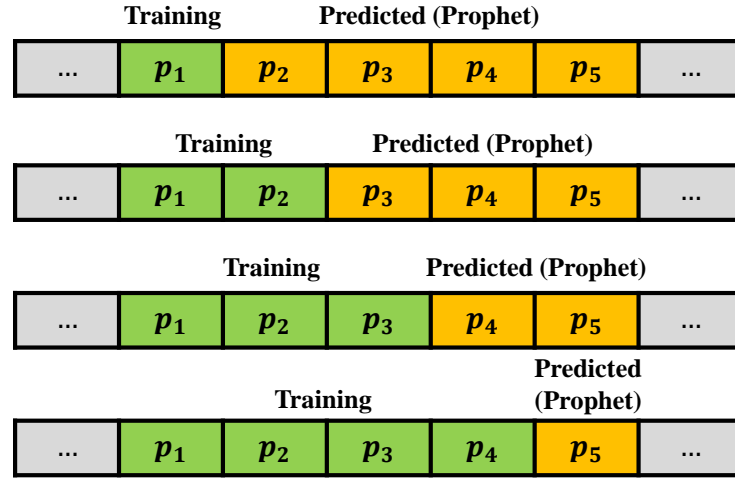


Figure 3.6 Illustrative example of applying Prophet.

3.4.5 Clinical Response Decision-Making Algorithm

Clinical decision making is the process by which the medical team know the progress situations of patients and then it determines who needs what and when. In order to make more efficient our platform, we propose a clinical response decision making algorithm that allows the medical team to determine the number of observations needed to check the patient situation during a period of time. The proposed algorithm takes advantage from the results of the Prophet forecasting and the criticality level of the patient and works based on the following steps:

- Assume a period p during a round u , we estimate the predicted datasets for the remaining $\mathcal{P} - \sigma$ periods based on the Prophet forecasting.
- We calculate the estimated criticality level \mathcal{C}_i , based on equation 3.2 and for all remaining periods, for each vital sign $\mathcal{V}\mathcal{S}_i$ of the patient.
- We calculate the aggregated score, e.g. ScoreU, for all criticality levels of the vital signs as follows:

$$ScoreU = \sum_{i=1}^{|\mathcal{V}\mathcal{S}|} \mathcal{C}_i \quad (3.6)$$

- The medical team makes the clinical response according to a predefined clinical response decision-making (CRDM) table (see Table 3.3). Typically, CRDM contains a set of m decisions where each decision is taken according to a range of ScoreU, e.g. $[b_i, b_j]$, with the observation

time needed for the patient. Indeed, the CRDM table is strongly defined by the experts and the doctors.

Table 3.3 Clinical response decision-making table (CRDM).

Decision #	ScoreU	Patient observation
\mathcal{D}_1	$[0, b_1]$	one time every 6 hours
\mathcal{D}_2	$] b_1, b_2]$	one time every 2 hours
...
\mathcal{D}_m	$\geq b_l$	one time every 10 minutes

3.5 Simulation Results

In order to evaluate the performance of our platform, we used the same real health data collected from Multiple Intelligent Monitoring in Intensive Care (MIMIC) database of PhysioNet [73]. In our simulation, we used a file that includes a log of about 100000 readings for each patient. We assume that each biosensor reads the data from its corresponding file for a period of time and sends them to the sink placed at 50 meters after applying our mechanism. We implemented the algorithms used in our platform based on Java simulator and we compared the obtained results to those obtained in the technique proposed in [3], e.g. modified local emergency detection (MLED). The parameters used in our simulations are shown in the Table 3.4:

Table 3.4 Simulation environment..

Paramter	Description	Description
\mathcal{T}	period size	900, 1600 and 3600
\mathcal{P}	round size	15 periods
k	degree of LSA polynomial	3, 5 and 7
d	division size	5 % and 10 %
vs	vital sign	heart rate

3.5.1 Patient Situation Study

In our simulations, we selected heart rate data of three patients having different situations' level (low, medium and high) in order to evaluate the performance of our framework. The calculated scores of readings collected for each patient are shown in Figure 3.7. The heart rate readings collected of the first patient (Figure 3.7(a)) are totally in the normal range (score = 0). The readings of the second one are almost abnormal with score criticality equals to 2 (Figure 3.7(b)). Whilst, the third patient shows a severe criticality of its situation where the scores vary between 2 and 3 (Figure 3.7(c)). Therefore, in the next of the simulations, we evaluate each metric according to the variation of the patient situations.

3.5.2 Performance Study of PRA Algorithm

In this section, we study the accuracy of the PRA algorithm proposed in our framework in terms of regenerating the raw data collected by the sensors. Indeed, the accuracy of PRA is highly dependent on two parameters (LSA degree and division size) which are studied in the following sections:

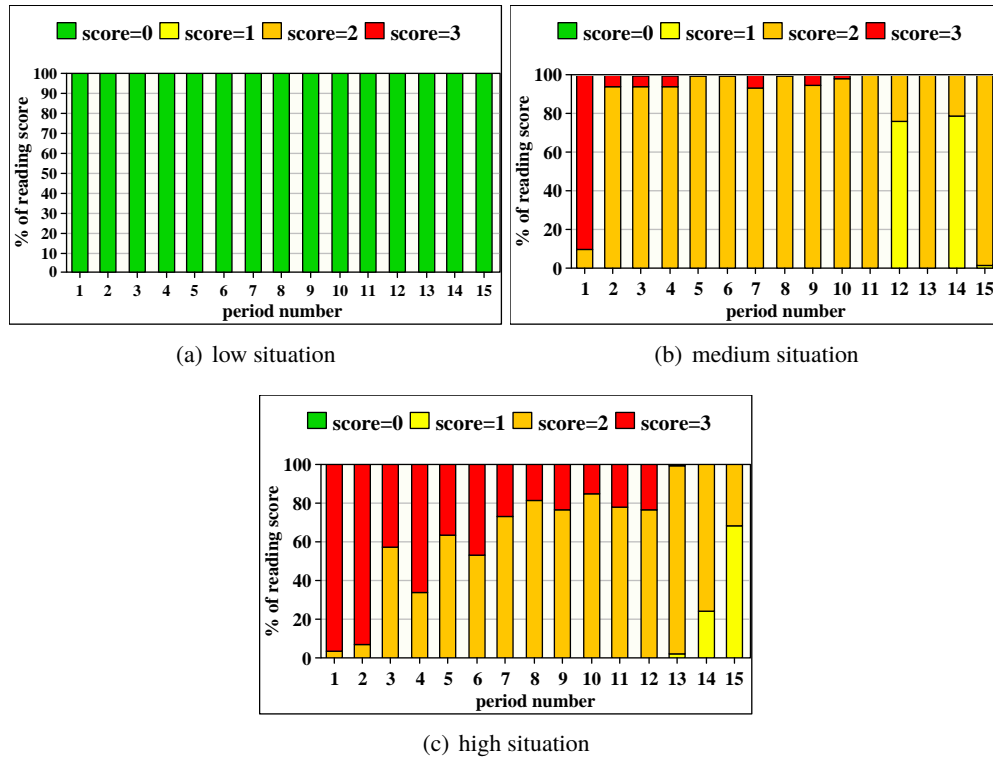


Figure 3.7 Patient situation according to the percentage of reading scores.

The Effect of LSA Degree (k)

At the end of each period, PED allows the sensor to send the non-sequential similar readings along with the set of LSA coefficients to the sink that tries to regenerate the raw data of the sensor in that period based on the coefficient set. Obviously, more the degree k of the LSA polynomial is more the regenerated data is accurate. Figure 3.8 shows the difference between the raw data and the data regenerated using PRA algorithm when varying the LSA degree k to 3, 5 and 7 respectively. From one hand, the obtained results show a high data accuracy of the regenerated data using the LSA method (Figure 3.8(a), or 3.8(b) or 3.8(c)). On the other hand, the increasing of the value of k leads to decrease the differences between the raw data and the regenerated ones (Figure 3.8(a) to 3.8(c)). Therefore, PRA ensures a high quality of data for both medical team and patient archiving process.

The Effect of Division Size (d)

In order to reduce its overhead process, PED selects a subset of readings based on the division size d to calculate the LSA polynomial instead of using the whole readings collected during the period. Obviously, more the division size is, more the number of selected readings reduces. Figure 3.9 shows the data regenerated at the sink using PRA, compared to the raw data, when varying the value of d to 5% and 10% respectively. The obtained results allow two observations: 1) the accuracy of the regenerated data using PRA is highly ensured in both cases; 2) the accuracy of PRA increases with the decreasing of the division size. This is because, the number of selected readings using PED increases with the decreasing of the division size which increases the accuracy of the LSA polynomial and therefore increases the performance of PRA.

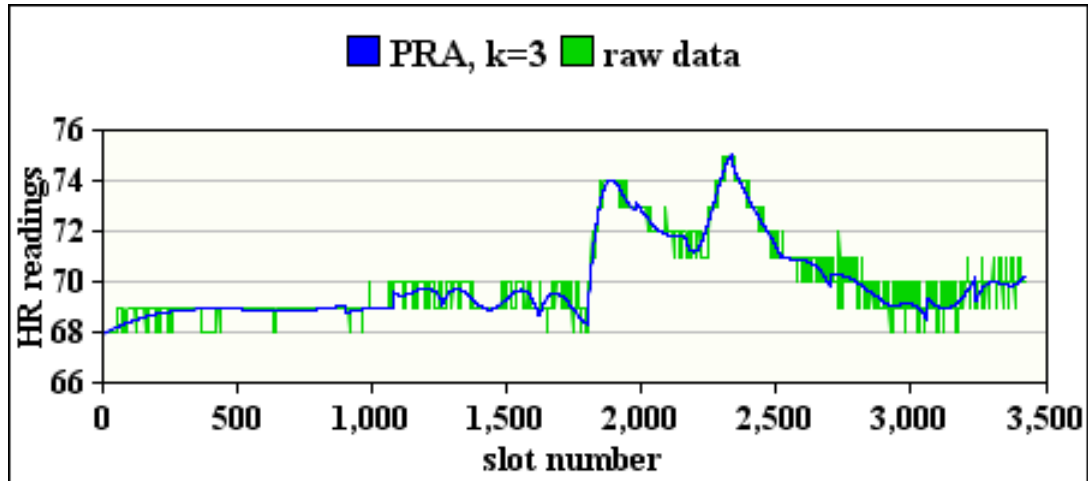
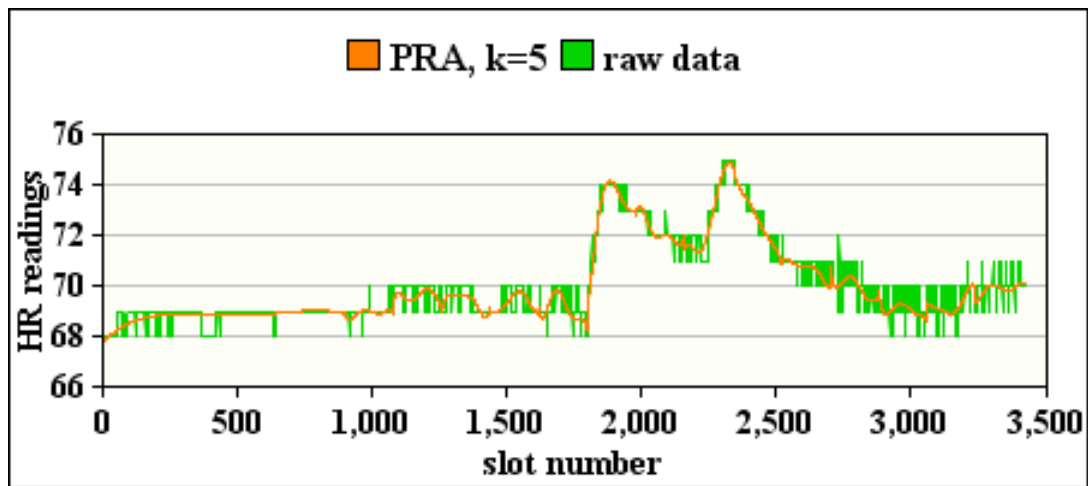
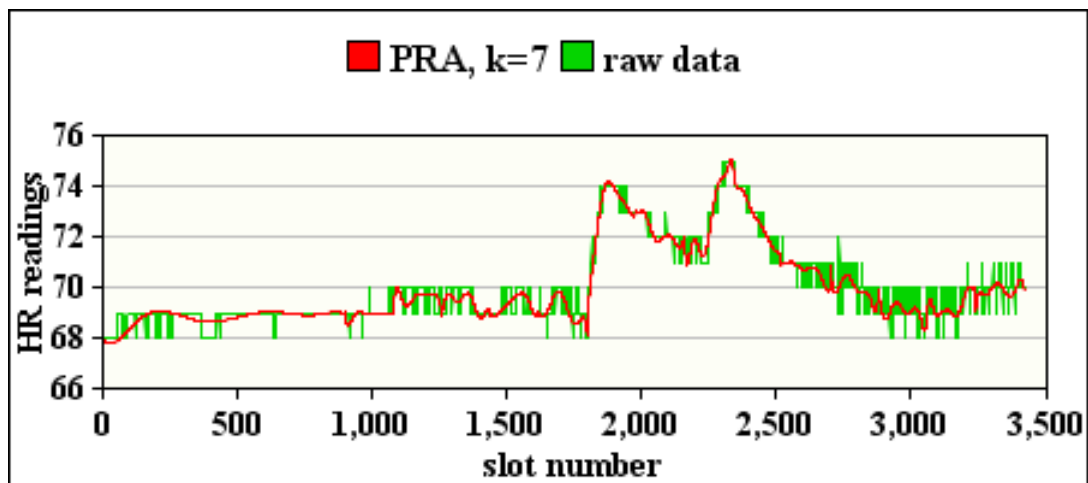
(a) LSA with $k = 3$ (b) LSA with $k = 5$ (c) LSA with $k = 7$

Figure 3.8 The quality of regenerated data after applying PRA in function of k , patient situation is low.

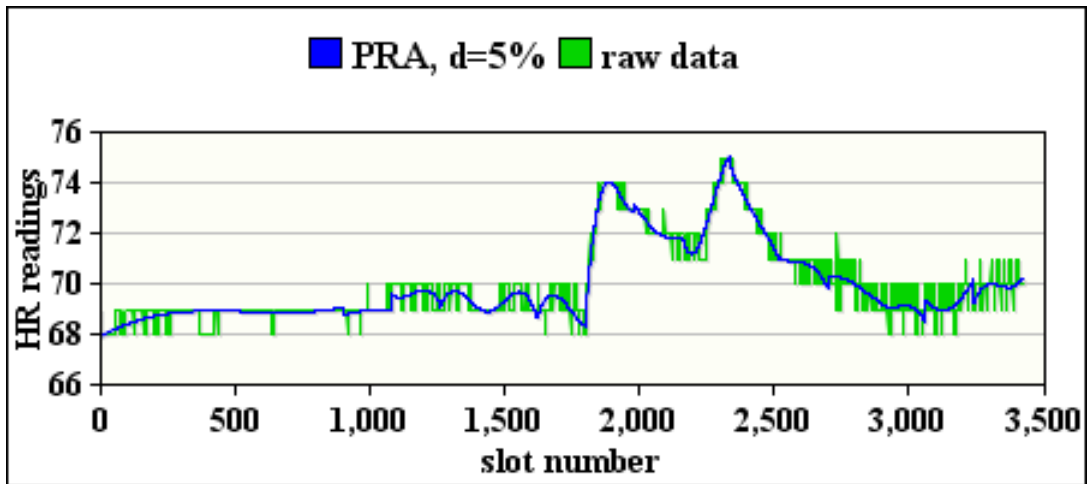
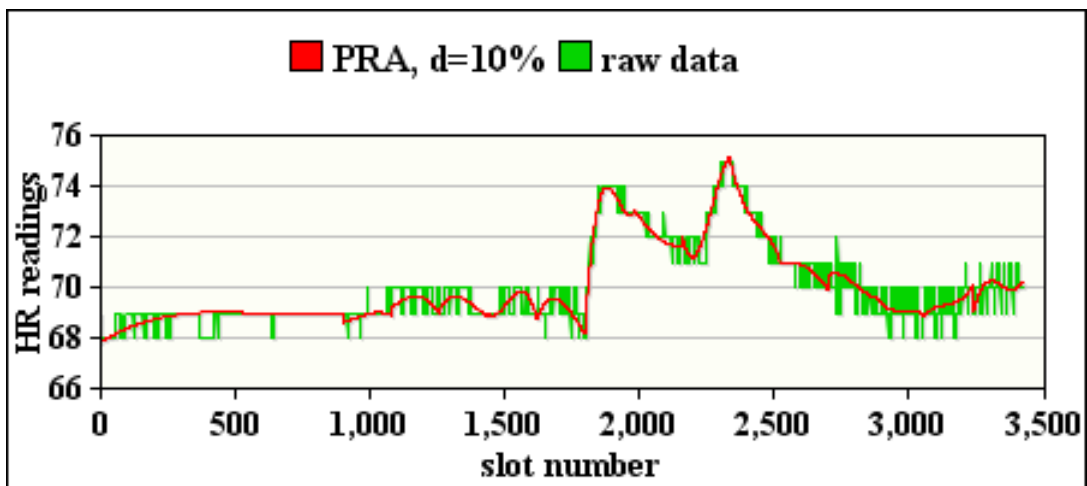
(a) LSA with $d = 5\%$ (b) LSA with $d = 10\%$

Figure 3.9 The quality of regenerated data after applying PRA in function of d , patient situation is low.

3.5.3 Performance Study of PMA Algorithm

In our framework, the PMA algorithm allows each sensor to adapt, at the end of each period, its sensing frequency according to the sensing frequency decision table (SFDT). Figure 3.10 shows how each sensor is able to adapt its sensing frequency after each period for the three taken patients with different criticality situations. We fixed the period size to 3600 readings and we compared the results of PMA to those obtained with MLED and naïve method, e.g. where all readings collected by the sensor are sent to the sink. The obtained results show that both algorithms allow sensor to dynamically adapt its sampling frequency according to the variation of the patient situation. However, PMA outperforms MLED in terms of reducing sensing frequency in cases of low and medium patient situations (Figure 3.10(a) and 3.10(b)) while the sensing frequency is more adapted using MLED in case of high patient situation. This reveals one of the drawbacks of the MLED algorithm in which a sensor should collect more data in an emergency case in order to continuously update the medical team about the patient situation. We can also observe that the sensing frequency of the sensor, using our algorithm, increases with the increasing criticality of the patient situation (low, medium and high respectively). For instance, PMA adapts the sensor frequency of the sensor to its minimum in the case of low patient situation (Figure 3.10(a)) while it is less adapted in the

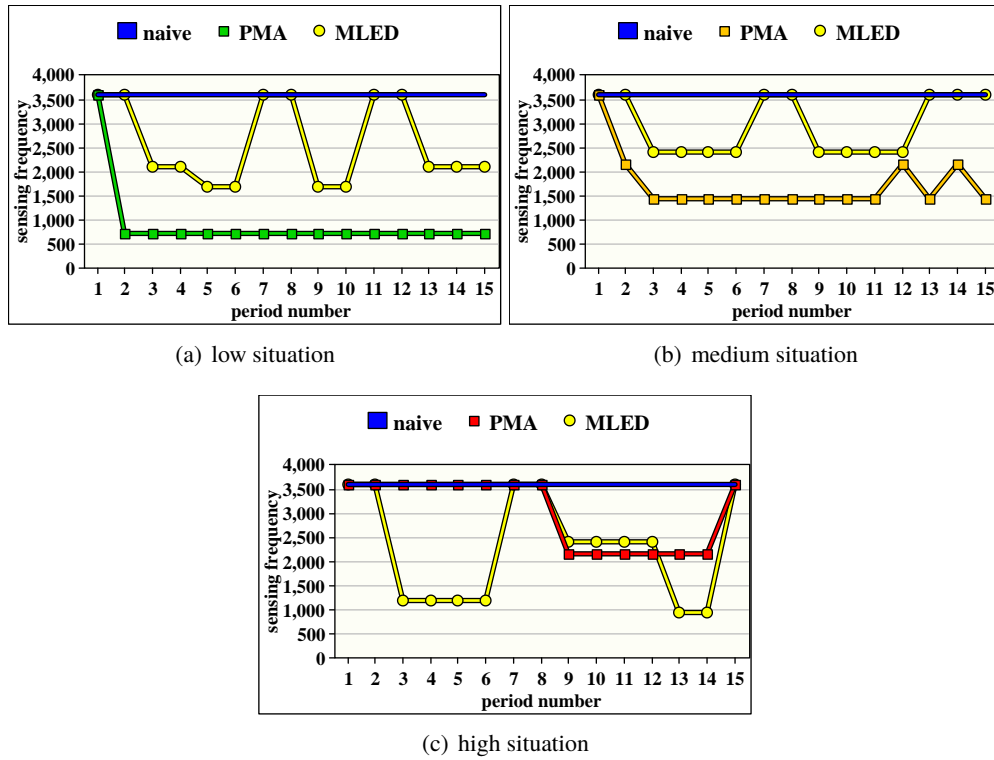


Figure 3.10 The variation of the sensing frequency of the sensor after applying the PMA in function of the patient’s situation level, $\mathcal{S} = 3600$.

high patient situation case (Figure 3.10(c)). This confirms the behavior of our algorithm in terms of providing an accurate patient situation transmission to the medical team.

On the other hand, Figure 3.11 shows the percentage of the sensed data using PMA, MLED and naïve algorithms in function of the variation of the period size. The obtained results are strongly related to those shown in Figure 3.10. Consequently, both algorithms (PMA and MLED) reduce the data collected by the sensor compared to the naïve approach where the amount of sensed data is reduced up to 76% and 54% using PMA and MLED respectively. Moreover, we observe that PMA reduces the sensed data from 27% to 66% compared to MLED in low and medium patient situations while MLED reduces up to 52% of the sensed data in the high patient situation compared to PMA. We also notice that, using PMA, the sensing frequency of the sensor is not greatly affected by the variation of the period size while, using MLED, it is decreased with the decreasing of the period size. This is because, the sensing frequency using PMA is dynamically adapted according to the patient situation level thus it gives better results among patients and not on the same one. Whilst, MLED uses the Fisher test to adapt the sensor frequency which gives better results when the variation between the sensed data is small, e.g. small period size.

3.5.4 Data Transmission Study

In Figure 3.12, we show the amount of data transmission after applying PED, MLED and the naïve approach in function of the period size. Indeed, PED allows the sensor to send the sequential readings with different scores, accompanied with the LSA coefficient set, to the sink in order to reduce the amount of data transmission. Whilst, MLED limits the sensor data transmission to the first reading of the period in addition to the critical readings, e.g. having score > 0 . The obtained results show two observations: for the low patient situation, both algorithms send a few amount of

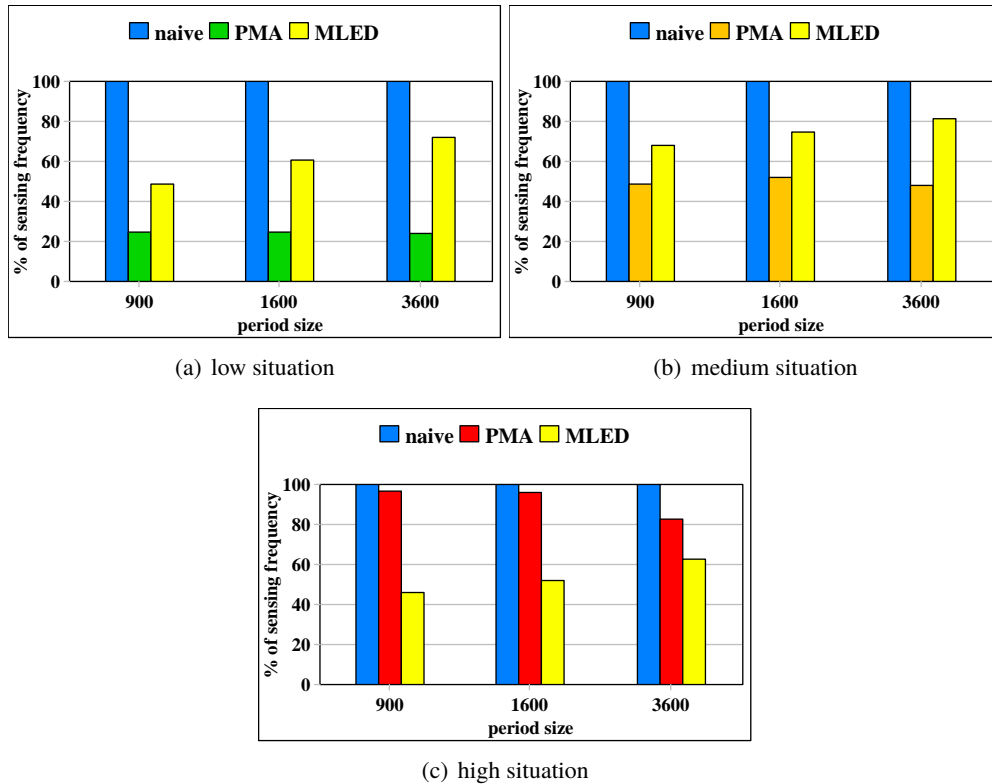


Figure 3.11 The percentage of sensing data after applying the PMA algorithm in function of the patient's situation.

data, up to 2%, to the sink compared to the naïve approach. This is due to the non-critical readings collected about the patient. For medium and high patient situations, PED reduces up to 95% the data transmitted from the sensor compared to the MLED. This is because the data transmission using PED is limited to the non-sequential similar score while MLED sends all critical readings to the sink.

3.5.5 Performance Study of MA

In Figure 3.13, we show the performance of the moving average method in terms of recovering the missing readings for a portion of about 1300 HR (Figure 3.13(a)) and RESP (Figure 3.13(b)) readings, compared to the original raw sensor data. We varied the window size (ws) in MA to 3 and 6 respectively. The obtained results show the following observations: 1) MA ensures a high accuracy for both abnormal and normal conditions. 2) more the size of window is low the accuracy of MA is; this is because, a missed reading will be similar to its near previous readings more than the far ones.

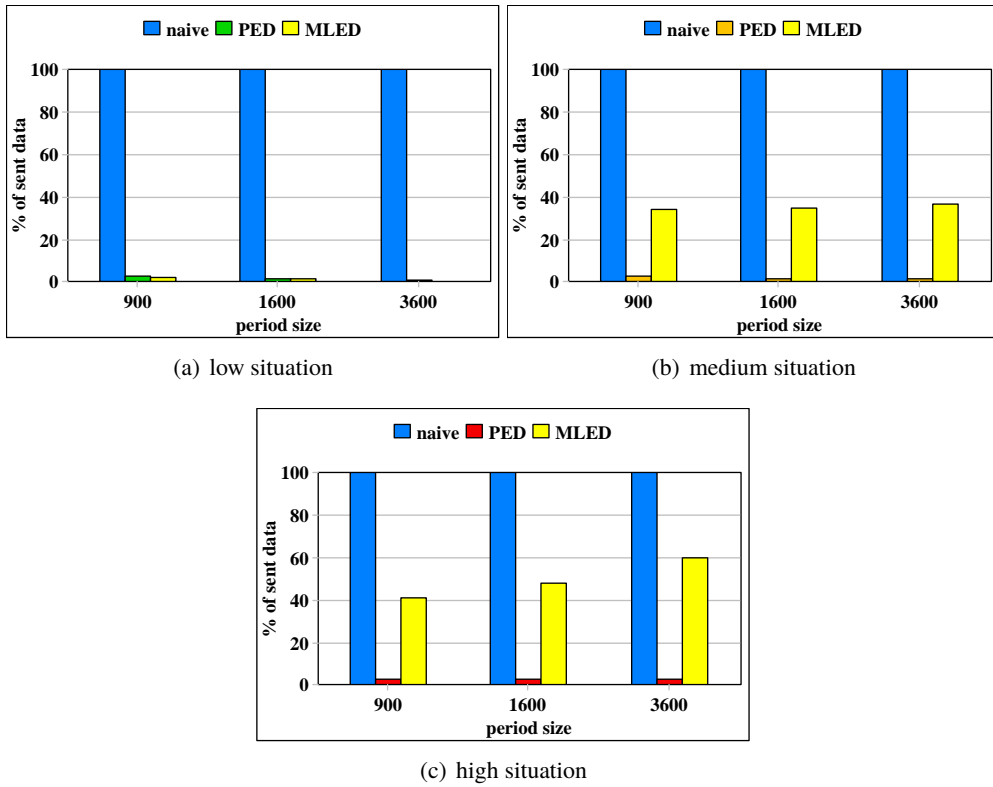


Figure 3.12 Percentage of data transmission from sensor to the sink, $k = 5$.

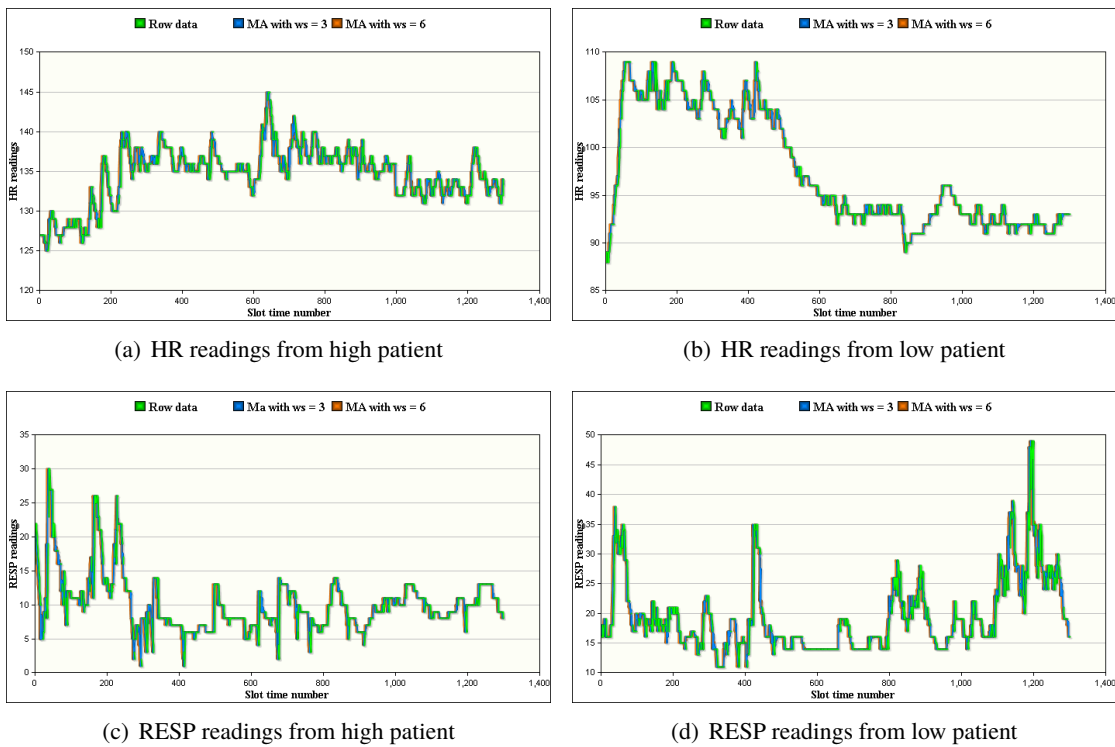


Figure 3.13 Raw data vs regenerated data with MA.

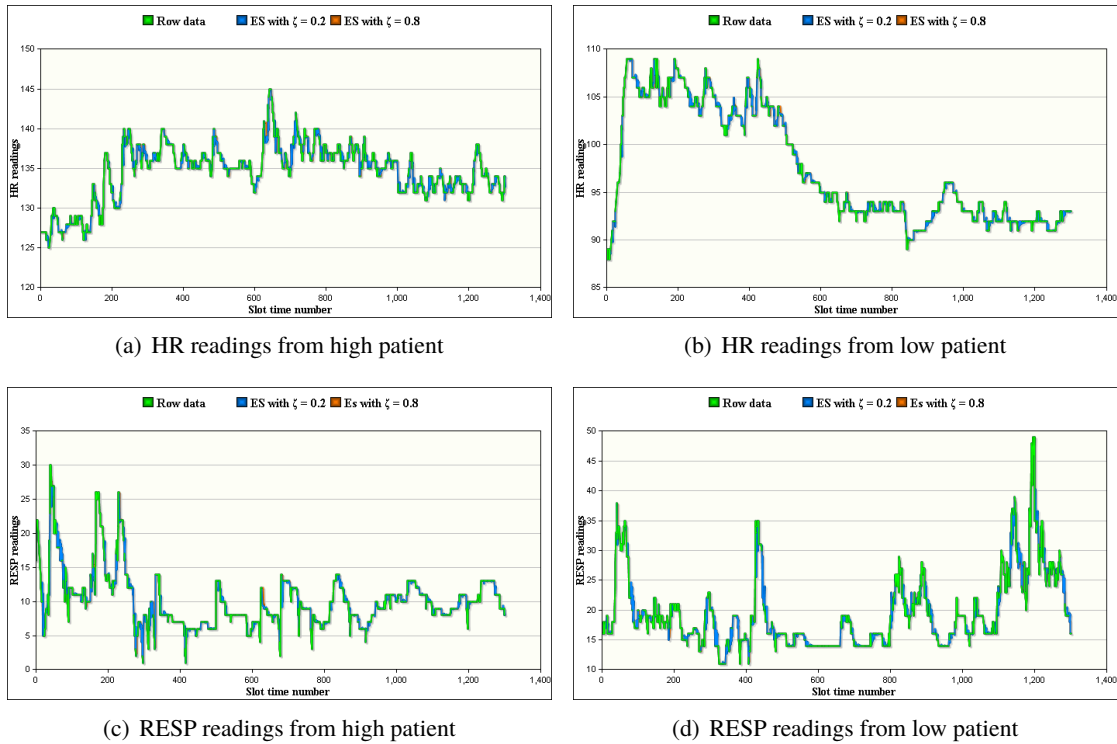


Figure 3.14 Raw data vs regenerated data with ES.

3.5.6 Performance Study of ES

Figure 3.14 studies the performance of the exponential smoothing method for the same portion of data shown in Figure 3.13 compared to the raw generated sensor data. The obtained results are highly dependent on the value of the ζ used in ES which it is varied to 0.8 and 0.2 respectively. Indeed, several observations are eminent: 1) ES gives better results with HR readings compared to those obtained with RESP. This is because the HR condition varies slowly which results a high similarity between the collected data thus, ES will effectively estimate the missing values. 2) The accuracy of regenerated data increases with the increasing of the value of ζ . 3) ES gives less data accuracy compared to the MA.

3.5.7 Performance Study of Processing Speed

In this section, we evaluate the performance of our framework in terms of processing (using MA and ES) and storing (using HDFS) the whole WBSN data described in the Simulation Results section. We calculated the processing speed, or the execution time by simulation, after regenerating the missing readings and store the whole WBSN data. The obtained results are shown in Figure 3.15. We first observe that the MA outperforms ES in terms of regenerating the missing values in a rapid execution process. We can also notice that execution time of MA decreases with the decreasing of the window size value while the processing speed in ES is almost fixed independently from the α threshold.

3.5.8 The performance Study of Prophet Method

In this section, we show the efficiency of the Prophet method integrated to the WBSN in terms of predicting the patient progress situation. In our simulations, we assumed a round of 15 periods where each period is set to 900 readings. Thus, at a given period $i \in [1, 15]$, the sink has to predict the readings for the remaining $15 - i$ periods according to the Prophet method. Figure 3.16 shows a

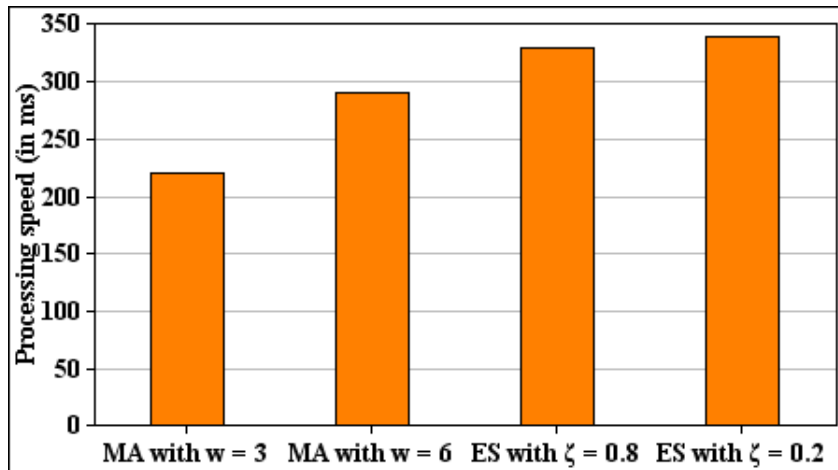
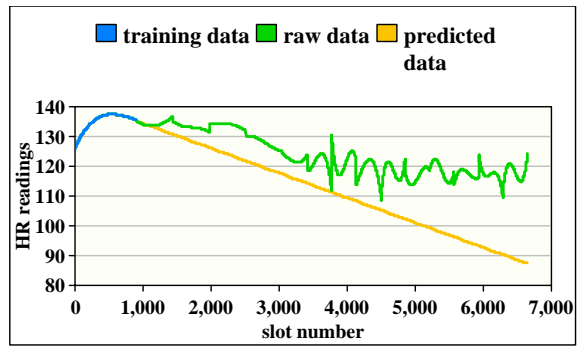
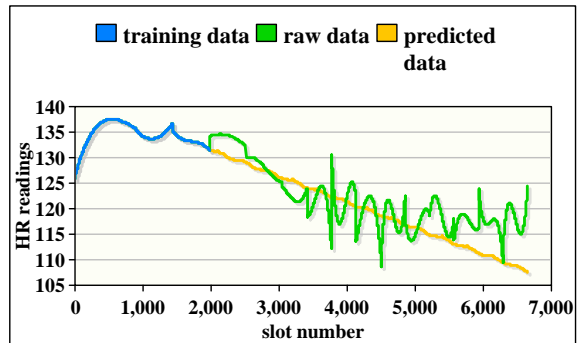


Figure 3.15 Processing speed using MA ans ES.

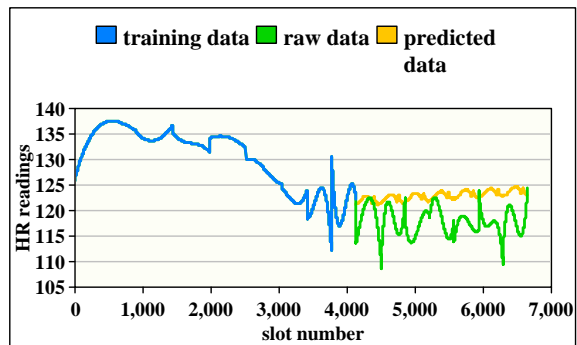
comparison between the raw data and the predicted data generated using the Prophet after the period numbers 1, 3, 8 and 12 in the round; the blue curve in each figure shows the readings received by the sink, e.g. training data, in which it has to train them in order to forecast the criticality level for the patient during the next periods, e.g. orange curve. From one hand, the obtained results show that Prophet ensures a good level of data accuracy regarding the forecasting of the patient readings. Hence, the medical staff will have an accurate information about the patient situation progress for the next periods thus, they can prevent him to enter in a critical situation and the nurses can determine the appropriate observations needed for that patient. On the other hand, we observe that predicting data generated using Prophet becomes more accurate when the increasing number of the period (Figure 3.16(a) to 3.16(d)). This is because the size of the training data will increase which allows Prophet to better understand the variation of the patient situation thus increase the accuracy of his progress situation.



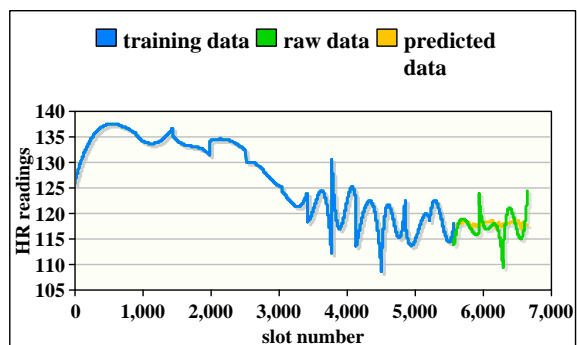
(a) period # = 1



(b) period # = 3



(c) period # = 8



(d) period # = 12

Figure 3.16 The predicted data using Prophet compared to the raw data, $\mathcal{T} = 900$, $\mathcal{P} = 15$, patient situation is medium.

3.6 Conclusion

Public health will stay one of the main concern of governments and industries due to the population growth and the increasing number of aging and elderly. Thus, WBSN will take more attention in patient monitoring as an efficient and low cost solution for hospitals. In this chapter, we have proposed a robust Patient-to-Doctor (P2D) framework for real-time health monitoring and decision making which works on two levels (sensor and sink). This framework covers entire life-cycle of data science including data collection; data ingestion; data preprocessing; data storing; and visualization, and it relies on hadoop ecosystem. P2D aims to reduce the energy consumption in the sensor and send any abnormal situation to the coordinator. On the other hand, P2D allows to make an archive for each patient and help the medical team to make a decision regarding the progress of the patient situation. Through simulations on real health data, we demonstrated the efficiency of our platform compared to other existing systems. The next chapter emphasizes the nurse scheduling issue, and it presents an efficient framework for nurse-patient organization. We will also present detailed simulations of this framework to show its effectiveness in balancing the workload of the medical personnel.

4 A Sensing-Based Patient Classification Framework for Efficient Patient-Nurse Scheduling

4.1 Introduction

Nurses are directly responsible for monitoring and evaluating patients and performing immediate interventions, to reduce risks or prevent medical complications [5]. A treating nurse even helps educate patients and family members about post-hospital care before the hospital discharge. Furthermore, the nurse expenses represents more than half of the hospital's expenditure. Moreover, the nurses suffer from two major challenges using WBSN; analyzing the large data provided by the sensors, and organizing their monitoring programs and shifts.

In this chapter, we present a novel and efficient framework for nurse-patient intelligent task organization that treats the scheduling problem on two phases. Subsequently, the contributions of this framework can be summarized as follows:

- *Patient classification:* One of the basic tasks assigned to the nurses is to classify the patients according to their critical states. The critical state of a patient is based on the sensed vital signs during a specific period. In this framework, we introduce a novel machine learning based method that aims to classify the criticality degree of each patient between low, medium or high, based on the criticality of all its vital signs.
- *Patients assigning to nurses:* A grouping method that aims to distribute a group of patients into several clusters is proposed within this framework. Each cluster is assigned to one nurse. We introduce a strategy that provides a fair distribution of patients on active nurses, in a way to normalize the exerted nurse efforts. This fair distribution takes into consideration the number and the criticality level of patients.
- *Patients scheduling:* A routing protocol is build within this framework in order to organize dynamically the nurse-patient schedule. This protocol allows to specify the best traffic that a nurse should follow in order to serve all her assigned patients. This routing algorithm depends on several parameters such as the patient's severity level and their age. Moreover, this protocol controls which patient should be examined and estimates the examination duration.

The rest of this chapter is organized as follows: Sections 4.2 and 4.3 details the two phases proposed in our framework respectively. The simulation results with the corresponding discussion are shown in section 4.4. Finally, we deduce a conclusion in section 4.5.

4.2 Patient Classification Phase

In this section, we present the patient classification method that classifies the patients according to the severity of their vital signs. Subsequently, an intelligent classification model is built passing by the training and testing phases (Figure 4.1). This model allows the nurses to identify periodically the level of criticality of each patient either low, medium, or high. These classes let the nurse to follow up their situation and to prevent any health deterioration through appropriate medical intervention.

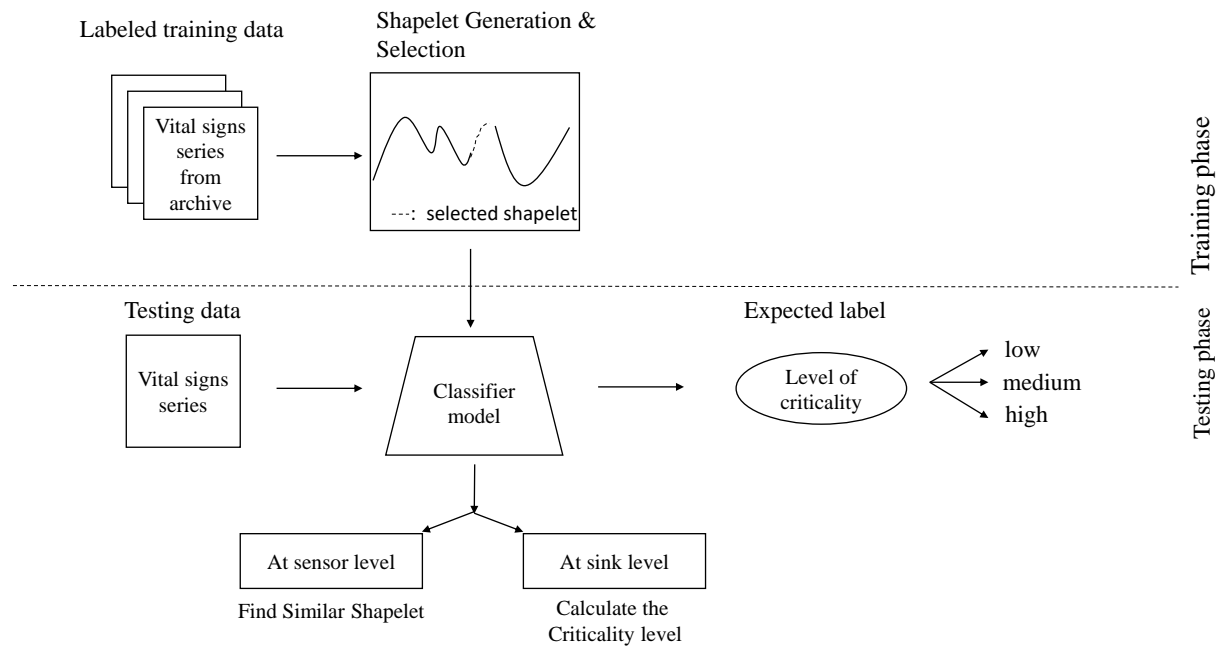


Figure 4.1 Patient classification system.

4.2.1 Training Phase

This phase consists of two steps: shapelet generation and shapelet selection. A shapelet SH with length l is a subsection election from one of the data series from archive D with length L . A SH represents the D and replaces it, where $L \gg l$.

Shapelet Generation

We generate several shapelets from patients history where each shapelet $SH_i(D, vs, level, l)$ is specific for one vital sign vs . In addition a shapelet inherits its $level$ from the level of data series D from which it is selected. The SH_i represents the i th shapelet. In order to select the optimal shapelet that represents the data series D , the Dynamic Time Warping (DTW) algorithm is applied [93]. DTW allows to calculate the distance between two sequential data of different length. In Figure 4.2, we give an example of heart rate series D with high level and $L=8$. Where the goal is to select a shapelet SH of size 4 from D using a shift window of size 1 to obtain all shapelets. Then, we calculate the distance between each shapelet SH_i and the data series D based on equations 4.1 and 4.2. The relationship between D and SH is represented by a time warping path W , where w_k is the element indicates the alignment and matching relationship between each two readings in the two series, and $\max(L, l) \leq k \leq L + l + 1$, where $d_{ij} = d(r_i, r_j)$ is the distance between each two readings r in each two series D and SH . Since we can use any distance measurements, we apply in

our framework the euclidean distance. Finally, the optimum shapelet having the minimum distance with the original D is elected. Therefore, for each vital sign, we obtain many shapelets that satisfy all levels space (low, medium and high).

$$DTW(D, SH_i) = \min \sum d_{ij}, W = (w_1, w_2, \dots, w_k) \quad (4.1)$$

$$d(r_i, r_j) = |r_i - r_j| + \min\{d(r_{i-1}, r_{j-1}), d(r_i, r_{j-1}), d(r_{i-1}, r_j)\} \quad (4.2)$$

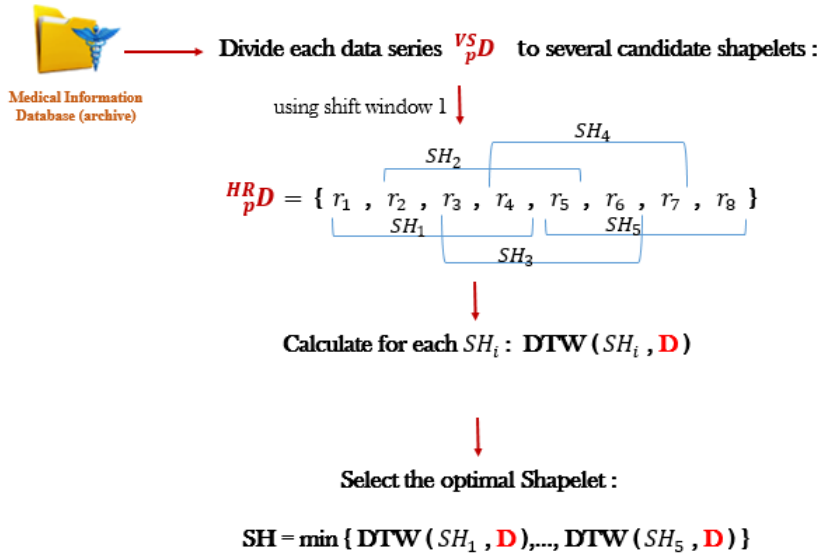


Figure 4.2 Illustrative example of shapelet generation.

Shapelet Selection

In order to eliminate the redundant shapelets, therefore to reduce the time consumption of the test process, a shapelet selection mechanism is applied, after the step of generation. This mechanism remove some of shapelets from similar shapelets. After the generation step, for each vital sign three groups of shapelets obtained: group for low shapelets; group for medium shapelets; and group for high shapelets. The sink works in each group following this steps:

- Calculate the percentage of similarity between all the shapelets using DTW (see above).
- Aggregate each two shapelets having maximum correlation (minimum DTW).
- Remove randomly one of them.
- Repeat until having sufficient number of shapelets.

4.2.2 Testing Phase

After shapelet generation, each set of shapelets for specific life's parameter is saved in the corresponding sensor that collects this type of data. The testing phase aims to determine the level of criticality of a patient at each period according to the levels of all vital signs. It operates on two levels: sensors and sink.

At Sensor Level

At the end of each period, the sensor starts to find the similar shapelet, among all saved shapelets, with the data collected during the period of time, according to the following steps:

- Divide the set of data collected to l equal division (where l indicates the shapelet length).
- Find the mean of each division.
- Compare the similarity between the serie of means with each shapelet by calculating the DTW.
- Select the shapelet having minimum DTW with the serie of means.
- Send the level of the similar shapelet selected to the sink.

At Sink Level

The sink received from one patient the levels of all vital signs, heart rate level; saturation of oxygen level; etc. Each vital sign vs_i is assigned to a score s_i , according to its level, as shows in Table 4.1. If the level of vs_i is low its score is 0; if it is medium, the score is 1; otherwise, the score is 2. Then, the sink assigns the criticality level of this patient, based on these scores of all vital signs. Equation 4.3 calculates the rate of criticality \mathcal{C} where Table 4.2 describes the relation between the criticality rate and the final level of severity of a patient.

$$\mathcal{C} = \frac{\sum_{i=0}^{NS} s_i}{2 \times NS} / \text{where } NS \text{ is the number of sensors.} \quad (4.3)$$

Shapelet level of vs_i	low	medium	high
score of vs_i, s_i	0	1	2

Table 4.1 Level and corresponding score of vs .

Criticality rate (\mathcal{C})	Criticality level
$0 \leq \mathcal{C} \leq 0.3$	low
$0.3 < \mathcal{C} \leq 0.6$	medium
$0.6 < \mathcal{C} \leq 1$	high

Table 4.2 Description of patient severity level.

4.3 Nurse Scheduling Phase

For accurate regulation in the hospital, patient scheduling is one of the most important challenges discussed by many researchers. The patients scheduling is the research problem in processes to find the optimal way to assign nurses to shifts, usually with a set of difficult constraints. This part of our framework is dedicated to the sink, that connected with all patient biosensors. Our technique is complemented by these two steps:

4.3.1 Patients Assigning to Nurses Algorithm

In order to better organize the work of nurses, this section starts by the distribution of patients, after the classification patient phase, at each period taking into consideration two parameters: the level and the number of patients. This step aims to divide the patients on the nurses existing in the same floor and to help the medical staff to acquire all patients in an organized manner. Two clustering-based mechanisms, called patients distribution based criticality balanced (PDCB) and patients sorting-based grouping (PSG), are proposed and in other hand we proposed a hybrid optimization algorithm, called Genetic Algorithm combined with Particle Swarm Optimization (GA-PSO), that combines genetic algorithm (GA) and particle swarm optimization (PSO). This proposed techniques aimed to assure the similarity between clusters in the number of patient and in the levels of criticality, and ensuring dissimilarity in patient levels in the same cluster. In the following sections, we describe the three techniques:

Patients Distribution-based Criticality Balanced (PDCB) Mechanism

Algorithm 8 explains the process of PDCB mechanism. It aims to allocate, for each nurse, the group of patients who should follow up during a period p . At the end of p , the sink owns the list of criticality rates Q of all the patients, where each \mathcal{C}_i in Q indicates the criticality rate of the patient i . The algorithm takes the following parameters: the number of patients (NP), the number of nurses (NN), and Q . First, the mechanism forms the clusters based on the number of nurses, where G_i is the i^{th} cluster for i^{th} nurse (lines 1-5). Then, we select the first NN patients with higher criticality from Q and put each of them in a cluster. After that, the criticality of each cluster is calculated using the equation in line 9. Furthermore, we balance the clusters that are not equal according to the criticality of the 1st cluster (line 14) by electing a patient from Q defined in the equation in line 15. Finally, we repeat until the Q becomes empty.

Algorithm 8 PCDB Algorithm.

Require: Period size: \mathcal{T} ; Patients number: NP ; Nurses number: NN ; A balancing threshold: ε ; A list of patient criticality during one period: $Q = [\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{NP}]$.

Ensure: Set of patients clusters to each nurse.

```

1: for  $i = 1$  to  $NN$  do
2:   Define cluster  $G_i$ 
3:    $G_i \leftarrow \emptyset$ 
4:    $\mathcal{C}_{G_i} = 0$ 
5: end for
6: repeat
7:   for  $j = 1$  to  $NN$  do
8:     select the Patient  $j$  with the highest critical score  $\mathcal{C}$ 
9:      $G_j \leftarrow G_j \cup \{j\}$   $\mathcal{C}_{G_j} += \mathcal{C}_j$ 
10:    remove  $\mathcal{C}_j$  from  $Q$ 
12:   end for
13:   for  $j = 2$  to  $NN$  do
14:     if  $\mathcal{C}_{G_1} \geq \mathcal{C}_{G_j}$  then
15:       select  $\mathcal{C}_k$  from  $Q / \mathcal{C}_{G_j} + \mathcal{C}_k \geq \mathcal{C}_1 + \varepsilon$ 
16:        $G_j \leftarrow G_j \cup \{Patientk\}$ 
17:        $\mathcal{C}_{G_j} += \mathcal{C}_j$ 
18:       remove  $\mathcal{C}_k$  from  $Q$ 
19:     end if
20:   end for
21: until no patient in  $Q$ 

```

Patients Sorting-based Grouping (PSG) Mechanism

Algorithm 10 shows the process of PSG mechanism that takes the same parameters of Algorithm 8. It starts by sorting the criticality list of all patients (Q) by decreasing order (line 1) and by forming NN clusters (lines 2-6). Then, it iteratively selects the first and the last patients in the list Q , and put each couple in a cluster G_i . Then, it calculates the criticality of the cluster by adding the criticality rates of patients in the cluster (lines 7-14). This step is repeated until Q becomes empty, or if there remains a number of patients smaller or equal to the number of clusters. After that, the algorithm distributes the patients in a way that balances the criticality of all clusters; this is done by selecting the cluster has the minimum total criticality \mathcal{C}_{G_i} and put in them the patient has the maximum criticality rate (lines 17-22).

Genetic Algorithm combined with Particle Swarm Optimization (GA-PSO)

Meta-heuristics are problem-independent strategies that allow us to explore the solution space more completely and, ideally, arrive at a good solution that is sometimes the same as the global optimum. GA-PSO nodes are encoded as genes during the GA phase. A chromosome encodes the path from the source to destination. The initial population is the set of all potential pathways from source to destination. The fitness function must be stated in terms of the scheduling parameters that are necessary. The fitness function will be used to evaluate fitness values for each chromosome. Begin the PSO phase, which consists of selecting parents from the population. During the GA phase, the population is initialized, and the fitness function is evaluated, later on in PSO phase start generating particle positions and velocities, velocity update, and finally position update are performed. This leads to the discovery of the best solutions from the solution space, which is used as parents in GA's crossover and mutation operations. When the termination requirement is reached, the new population is presented as a solution set with optimal paths that satisfy the relevant schedule metrics.

- **GA-PSO Algorithm:** Algorithm 1 shows the process of GA-PSO that allows to select the best feature combination in this mechanism. The algorithm takes, as input, the criticality rate of each patient along with the configuration parameters required for GA and returns, as output, the best solution B for nurse scheduling for the purpose of serving the patient in timely manner after collecting of the vital signal. Firstly, the algorithm randomly generates an initial list of ω combined nurses-patients re-partition (line 1). Then, each combined solution set is evaluated according to the Fitness function (line 3) where the set having the max fitness value is selected to be the best current solution (lines 10-13) in order to produce the new chromosomes (set of children) by the crossover with the each other chromosome in ω (line 5). Then each chromosome (child) is performed using the mutation operations (line 6).

After that, a loop is started in which the GA coefficient is enhanced (line 9) and the combined solutions are updated (line 15). Then, a new list of combined schedule sets is randomly recalculated and the current best solution (local best position $pbest$, global best position $gbest$) is updated in case a new better is detected (line 10-14). The loop is continued until reaching the maximum number of iterations or the convergence criteria is met.

- **Initial Population:** First, many potential solutions are randomly generated to form an initial population ω . The population consists of S individuals that means S chromosomes or Solutions. The initial population represents set of all potential paths.
- **Fitness Function:** The fitness function is always problem dependent and measures the quality of the depicted solution. Individual performance should be reflected in the fitness function. The fitness value is used to evaluate each solution of the population and to indicate how close it came to the optimum. We define the fitness function as f .

The settings and notations of the proposed model are described as follows:

Algorithm 9 GA-PSO Algorithm.

Require: Period p , Period size: τ , Patient number NP , Nurse number NN , List of patients criticality during p : $Q = [\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_P]$.

Ensure: Best solution: BS

- 1: Generate the initial random population ω
 - 2: **repeat**
 - 3: Calculate the fitness value f for each individual based on equation (4.4)
 - 4: Select the parent individuals for reproduction
 - 5: Produce set of children by crossover operation
 - 6: Perform mutation operations
 - 7: Evaluate the fitness value for each child
 - 8: Select the best individual for the current generation:
 - 9: **for** each individual i **do**
 - 10: **if** $f(i) > f(pbest)$ **then**
 - 11: $pbest \leftarrow i$
 - 12: **end if**
 - 13: **if** $f(i) > f(gbest)$ **then**
 - 14: $gbest \leftarrow i$
 - 15: **end if**
 - 16: Update positions and velocity for each individual according to equations (4.8) and (4.9)
 - 17: **end for**
 - 18: **until** convergence criteria is satisfied
 - 19: **return** BS
-

$$f = \min C \sum_{k \in SN} \sum_{i \in SP} \sum_{j \in SP, i \neq j} x_{ijk} d_{ij} + X_i \quad (4.4)$$

where:

- x_{ijk} : binary variable representing whether Nurse visits Patient node j after visiting Patient node i
- d_{ij} : Euclidean distance between patient node i and patient node j
- SP : set of patients' nodes
- SN : set of nurses
- C : nurse shift
- Q : load capacity of nurse
- tf : service time at node i , $i \in SP$
- X : position of the solution

Based on the above notations, we define the following constraints:

$$\sum_{i \in SP, i \neq j} x_{ijn} = 1, \forall j \in SP, \forall n \in SN \quad (4.5)$$

$$\sum_{i \in SP} \mathcal{C}_i \leq Q, \forall i \in SP \quad (4.6)$$

$$\sum_{i \in SP} \sum_{j \in SP, j \neq i} x_{ijn} \leq SP, \forall n \in SN \quad (4.7)$$

- **Selection of Parents:** The best individuals are chosen from among those solutions based on the PSO model. In PSO, each solution is referred to as a "particle" and is represented by a "bird" in the search space. All particles have their fitness values evaluated by a fitness function, as well as velocities that direct the particles' flight. These values are optimized to be as accurate as possible. The particles can easily navigate through the space by following the path of the ideal particles and performing the search for optima using the generations. Figure 4.3 illustrates that for each particle i , at each time t , consists of a position x_i^t a candidate solution a velocity v_i^t and the best location it has ever visited $pbest_i^t$ with respect to a fitness function. The particles of the swarm share information about the search space via the global best position $gbest_i^t$, which is the best position among all particles. Equations (4.8) and (4.9) control the movement of a particle:

$$V_i = V_i + c_1 \times r_1 \times (pbest_i - X_i) + c_2 \times r_2 (gbest - X_i) \quad (4.8)$$

$$X_i = X_i + V_i \quad (4.9)$$

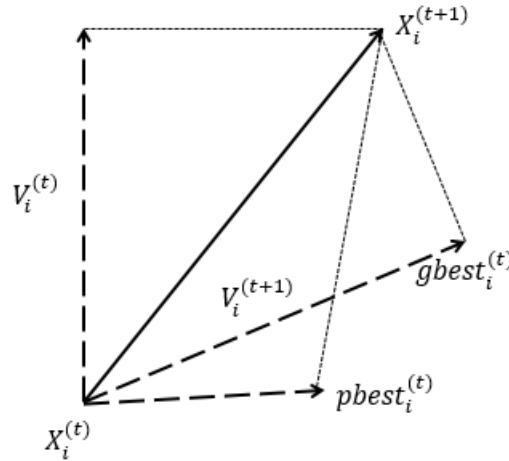


Figure 4.3 Illustrative example of PSO exploration space search.

Where i is the number of the particle, t is the iteration index. c_1 and c_2 are constants called learning factors that control the effect of the personal memory of a particle and the shared information of the swarm, respectively. $r_1^l(t)$ and $r_2^l(t)$ are generate randomly in $[0,1]$. The process is repeated in discrete time steps until a stopping criterion is met. By modifying its parameters, it is able to control the swarm's trade-off between exploration, having to visit to new locations and learn more about the search space, and exploitation, deciding to stay closes the best position found so far.

- **Crossover Operation:** A child created after a crossover operation is performed between the two paths chosen. Cross over is the process of combining multiple parent solutions to create a child solution. A pair of chromosomes is chosen as the parents in the PSO selection model to produce a single offspring. The crossover procedure consists of four steps:

- Step 1: select a segment group randomly from the chromosomes (Figure 4.4).

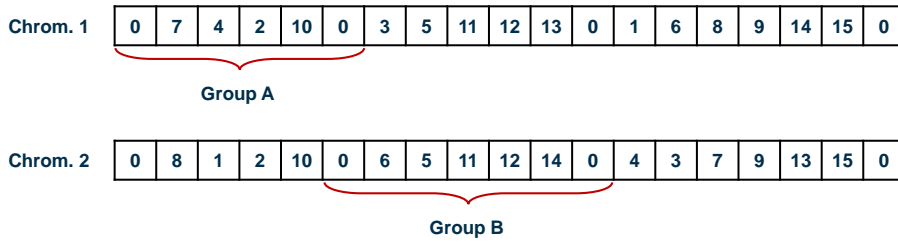


Figure 4.4 Segment selection from the chromosomes.

- Step 2: move the chosen groups in each parent to the chromosome's front genes (Figure 4.5).

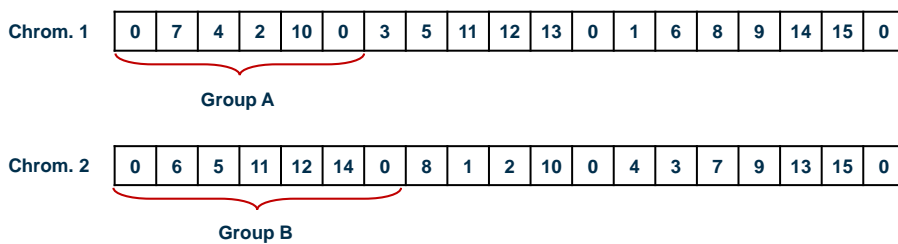


Figure 4.5 Group shifting.

- Step 3: in offspring 1, keep Group A of chromosome 1, but add the genes of chromosome 2 that chromosome 1 does not include after Group A as their sequence (Figure 4.6).

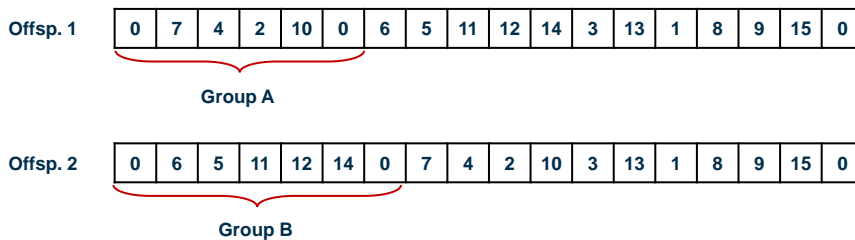


Figure 4.6 Combining genes.

- Step4: add number zero in any gene after Group A. Offspring 2 is obtained in the same manner (Figure 4.7).

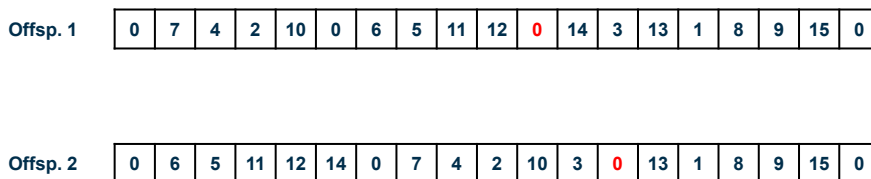


Figure 4.7 Adding zeros.

- **Mutation:** Following the crossover operation, the new offspring is subjected to a mutation operation. The operator chooses two different locations to relocate to, and each operation is capable of achieving five new solutions. The mutation is used to broaden the scale of searching, and the set of mutation probability can control the search trend (Figure 4.8). As a result, a new population is formed. The fitness value is applied to the newly generated offspring once more. The newly generated efficient unicast multiple paths can then be accepted if they provide the best solution. Otherwise, it is chosen for reproduction once more. As a result, the resulting paths will provide optimized multiple unicast paths from the source to the destination.

Offsp. 1	0	7	4	2	10	0	6	5	11	9	0	14	3	13	1	8	12	15	0
Offsp. 2	0	6	5	11	12	1	0	7	4	2	10	3	0	13	14	8	9	15	0

Figure 4.8 Offspring after mutation operation.

4.3.2 Patient-Nurse Scheduling (PNS) Algorithm

Appointment scheduling is especially important in medical centers, where a long list of patients with limited equipment, rooms and human resources should be examined within a certain period of time. After the distribution of patients, each nurse became responsible for a specific group of patients. Here it begins the stage of finding the best way in which the pathogen has to follow up in order to move from patient to a another for examination, and to give him medicine if there is a need. During each period, nurse should control the patients of high and medium level while the low patient is not necessary to follow it at this period. The principal steps of PNS algorithm 11 are summarized as follows:

- We first calculate the duration of every single appointment based on patients' particular therapy requirements. The calculation consists on dividing the period time on the patients according to equations (4.10a) and (4.10b) where: ht is the time required for controlling high patient and mt for the medium patient; nhp is the number of high patients and nmp is the number of medium patients (line 2). Subsequently, equation (4.10a) represents the duration of the period with a decrease δ of time. This threshold is used for the subjectivity of the nurse that can increase/decrease it according to the patient need and necessity. In addition, this threshold allows the nurse to be able to take into account any emergency and sudden event for low patients, or for the sake of convenience. Whilst equation (4.10b) means that the time for high patients should be the double of the time of medium patient.

$$nht \times ht + nmp \times mt = \tau - \delta \quad (4.10a)$$

$$ht = 2 \times mt \quad (4.10b)$$

- In order to determine the order of patients for regular follow-up, a priority for each patient is calculated based on two constraints: the criticality rate (\mathcal{C}) and the age (A) of the patient. On one hand, the higher the patient's level of risk, the higher the priority for his examination. On the other hand, if a patient is in the youth stage he can bear more than others. Therefore, we propose two equations (4.11a) and (4.11b) that allows to calculate the priority of a patient. In

Algorithm 10 PSG Algorithm.

Require: Patients number: NP ; Nurses number: NN ; A list of patient criticality during period p :

$$Q_p = [\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{NP}].$$

Ensure: Set of clusters of patients of each nurse.

```

1: Sort  $Q$  in descending order
2: for  $i = 1$  to  $NN$  do
3:   Define cluster  $G_i$ 
4:    $G_i \leftarrow \emptyset$ 
5:    $\mathcal{C}_{G_i} = 0$ 
6: end for
7: while  $NP > 2NN$  do
8:   for  $j = 1$  to  $NN$  do
9:      $G_j \leftarrow G_j \cup \{Patient\ j, Patient\ NP\}$ 
10:     $\mathcal{C}_{G_j} += \mathcal{C}_j + \mathcal{C}_{NP}$ 
11:    remove  $\mathcal{C}_j$  and  $\mathcal{C}_{NP}$  from  $Q$ 
12:     $NP - = 2$ 
13:   end for
14: end while
15: for  $i = 1$  to  $NP$  do
16:   select the cluster having minimum criticality  $G_k$ 
17:    $G_k \leftarrow G_k \cup \{Patient\ i\}$ 
18:    $\mathcal{C}_{G_k} += \mathcal{C}_i$ 
19:   remove  $\mathcal{C}_i$  from  $Q$ 
20:    $NP - = 1$ 
21: end for

```

this equations, λ represents the weight of the patient criticality and ψ represents the weight of the patient age where the sum of the two weights must be 1; we consider that λ should be greater than ψ because criticality is more important than age in the patient selection to follow it, and their summation is equal to 1. Equation (4.11b) is dedicated to the patient children that are considered critical as the elderly patient; ϕ is a parameter that allows to increase the priority of patient children, which ϕ is much greater than the patient age (lines 3-11).

$$PRI = \lambda \times \mathcal{C} + \psi \times A, \text{ if } A > 18 \text{ years} \quad (4.11a)$$

$$PRI = \lambda \times \mathcal{C} + \psi \times (\phi - A), \text{ if } A \leq 18 \text{ years / where } \phi \gg A \quad (4.11b)$$

- We finally sort the patients by decreasing order of their priorities (line 12). Thus we obtain the queue that contains the patients arranged with their assigned time calculated on the first step. Consequently, the nurse will assign the patient queue for better and fast service, and for good reputation and humanity in the hospital.

Illustrative example of PNS: Given a set of six patients (Patient 1 to Patient 6) assigned to a nurse, with their criticality rates (\mathcal{C}_1 to \mathcal{C}_6 respectively), their levels and their ages during one period with $\mathcal{T} = 60$ minutes. we assume $\lambda = 0.8$, $\psi = 0.2$, $\phi = 90$, and $\delta = 10$ min (Figure 4.9). Consequently, the sink calculates the time necessary to follow up high patients $ht = 6.25$ min and medium patients $mt = 12.5$ min, where there is three high patients and two medium patients, with the possibility to extend or reduce the duration of the examination, that by keeping $\delta = 10$ min. Then, the sink computes the priority for each patient; for instance, the priority of Patient 1 is calculated as $PRI_1 = 0.8 \times \mathcal{C}_1 + 0.2 \times A_1 = 0.8 \times 90 + 0.2 \times 36 = 79.2$, because $A_1 = 36 > 18$. For Patient 4, its $A_4 = 13$

Algorithm 11 PNS Algorithm.

Require: Period size: \mathcal{T} ; A Nurse i ; A list of patient criticality assigned for Nurse i during p : $L_p = [\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_{NS}]$; A list of patient age: $A_p = [A_1, A_2, \dots, A_{NS}]$; Number of high patients: nhp ; Number of medium patients nmp ; Subjectivity threshold δ ; Coefficients: λ and ψ .

Ensure: Patients queue assigned to Nurse i : Q .

- 1: $Q \leftarrow \emptyset$
- 2: Calculate ht & mt based on equations (4.10a) and (4.10b).
- 3: **for** $j = 1$ to NS **do**
- 4: **if** $A_j < 18$ **then**
- 5: Calculate PRI_j based on equation (4.11a).
- 6: $Q \leftarrow Q \cup \{Patient\ j\}$
- 7: **else**
- 8: Calculate PRI_j based on equation (4.11b).
- 9: $Q \leftarrow Q \cup \{Patient\ j\}$
- 10: **end if**
- 11: **end for**
- 12: Sort Q in descending order

< 18 , so its priority is calculated as $PRI_4 = 0.8 \times 70 + 0.2 \times (90 - 13) = 71.4$, and so one. Finally, we order the patients according to their priorities while setting the time for each one.

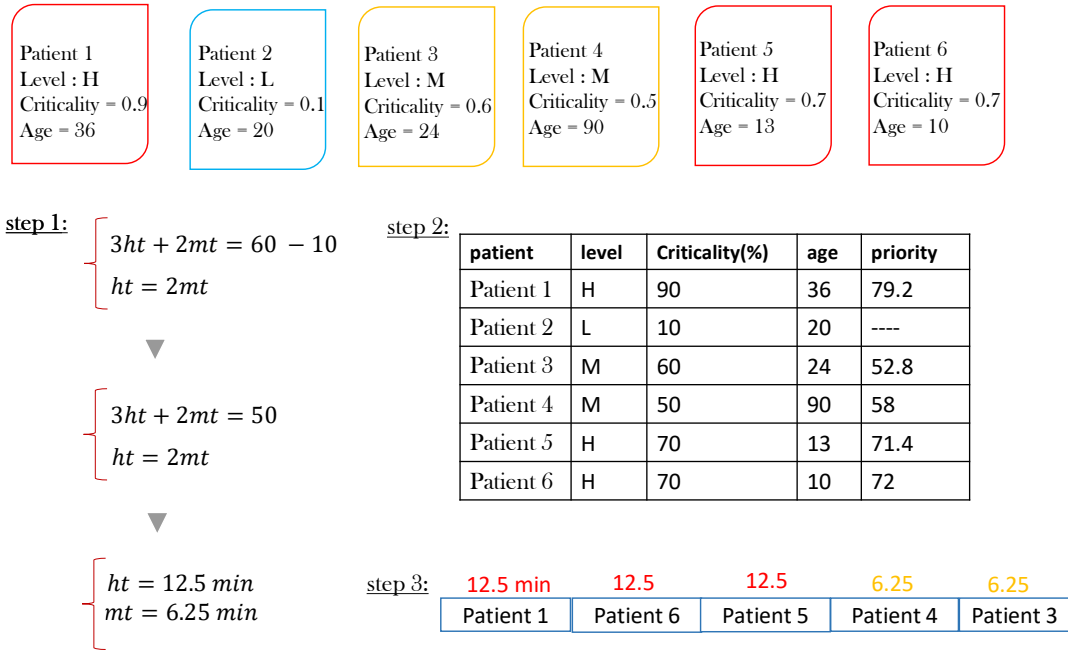


Figure 4.9 Illustrative example of PNS algorithm.

4.4 Research Results

In this section, we show the relevance of our framework that is implemented based in Java. We used real data health published in Multiple Intelligent Monitoring in Intensive Care (MIMIC)

database of PhysioNet [22]. In our simulation, we are interested on three vital signs: heart rate (HR), respiration rate (RESP), and oxygen saturation (SPO2). We run our framework on 10 periods where the simulation environment is set according to the following table :

Parameter	Description	Value
\mathcal{T}	period size	1000
NS	number of sensors	3
δ	subjectivity threshold	10
NN	number of nurses	3,8,15
NP	number of patients	15,18,30,50
vs	vital sign	HR, RESP, SPO2
s	score of vital sign	0, 1 or 2
λ	weight of the patient criticality	0.8
ψ	weight of the patient age	0.2
ST_1 and ST_2	Serving time	0,8.15,17.5 and 1,6,10
ω	Population size	50
CV	Cross validation constant	0.9
MC	Mutation constant	1/20
r_1, r_2	Random number	0,1
c_1, c_2	Learning factors	2,2

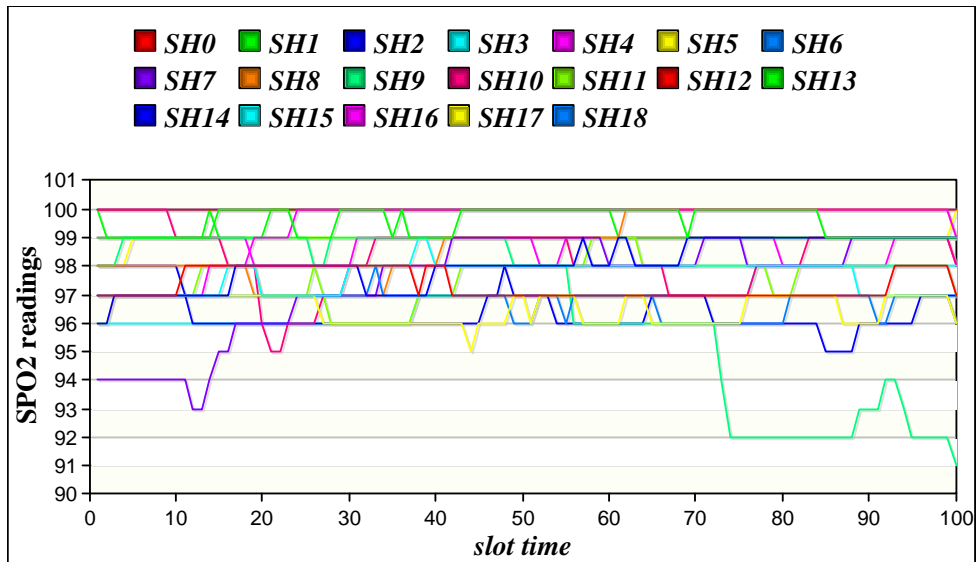
Table 4.3 Simulation environment.

4.4.1 Performance Study of Shapelet Generation Method

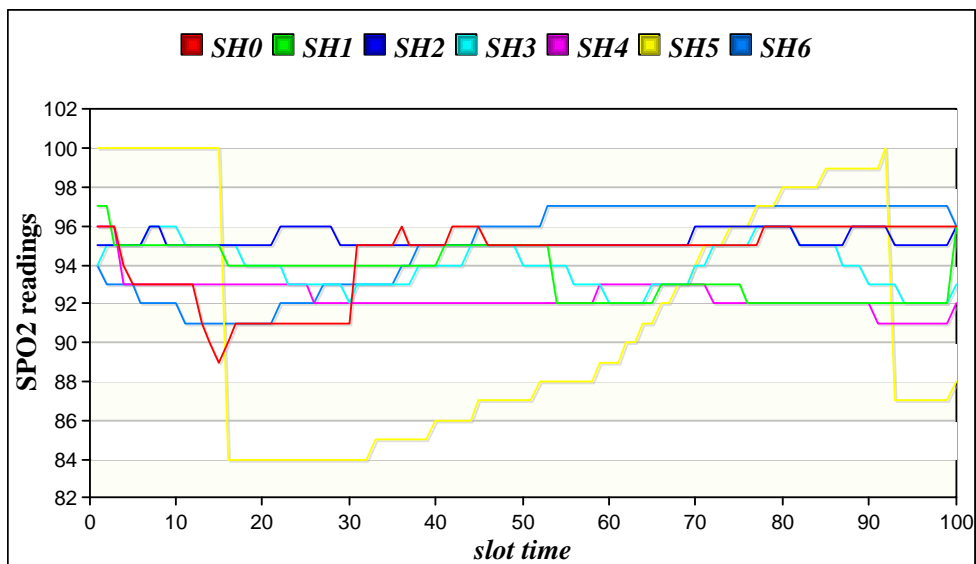
Figure 4.10 shows the results of shapelet generation step for the oxygen saturation (*SPO2*) vital sign according to the first step of the training phase in the classification model system. These shapelets are selected from archive of some patients with diversity of their levels of criticality in order to cover all Situations kinds of new patients. Our purpose is to increase the effectiveness of the model based on machine learning approach. The obtained results divided of 19 shapelets with low-level patients (Figure 4.10(a)), 7 shapelets with medium- level patients (Figure 4.10(b)), and 15 shapelets with high-level patients (Figure 4.10(c)).

4.4.2 Performance Study of Finding Similar Shapelet

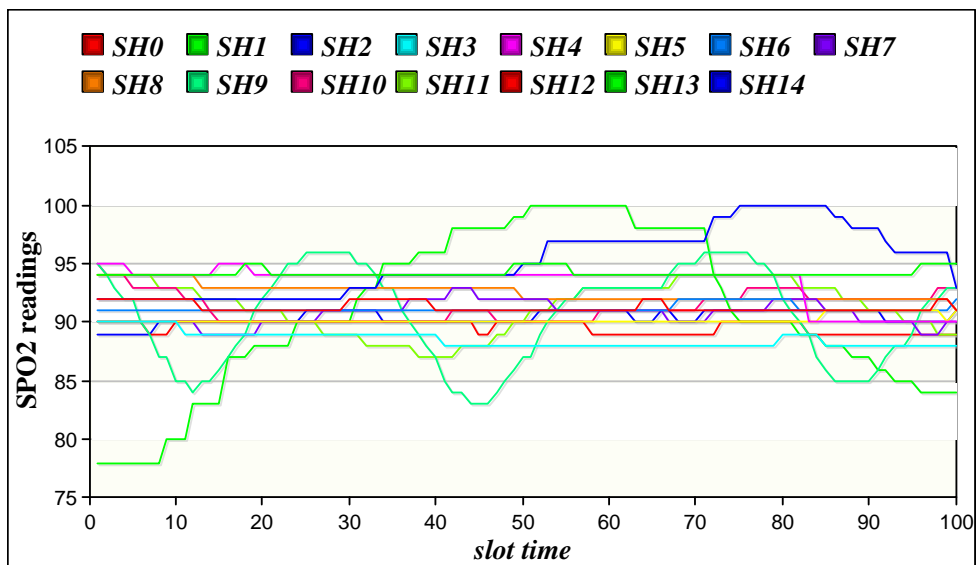
In order to study the accuracy of the patient classification system, the shapelets elected is shown to determine the level of each vital sign. The green curve in each figure shows the readings detected by the sensor during a period. And the other curve shows the similar shapelet obtained by this model, if their color is red that means the level of shapelet is high, if it is orange the level is medium, and if it blue so the level is low. Figure 4.11 indicate the results for Patient 1, their heart rate level is high because the similar shapelet obtained has a high level shown in Figure 4.11(a) , and the Figure 4.11(d) shows the score frequencies for each vital sign, and it verifies the high level criticality of heart rate, because the heart rate readings collected of this patient (Figure 7(a)) are totally in the high range (score = 2 and 3). The selected shapelet for respiration rate *RESP* has high level (Figure 4.11(b)), in other hand the result confirmed in Figure 4.11(d), where the most scores of respiration data are of a high level. And also all the oxygen saturation *SPO2* readings in the normal readings (Figure 4.11(d), corresponding to their obtaining shapelet similar is for low level (Figure 4.11(c)). The same for the Patient 9 their three selected shapelets illustrated in the Figures 4.12, where their levels are low, medium, and high for vital signs respectively *HR*, *RESP*, and *SPO2*. And also the results of Patient 3 are shown in Figure 4.13, where their criticality level moving towards the low.



(a) low shapelets



(b) medium shapelets



(c) high shapelets

Figure 4.10 The shapelets generated for SPO2.

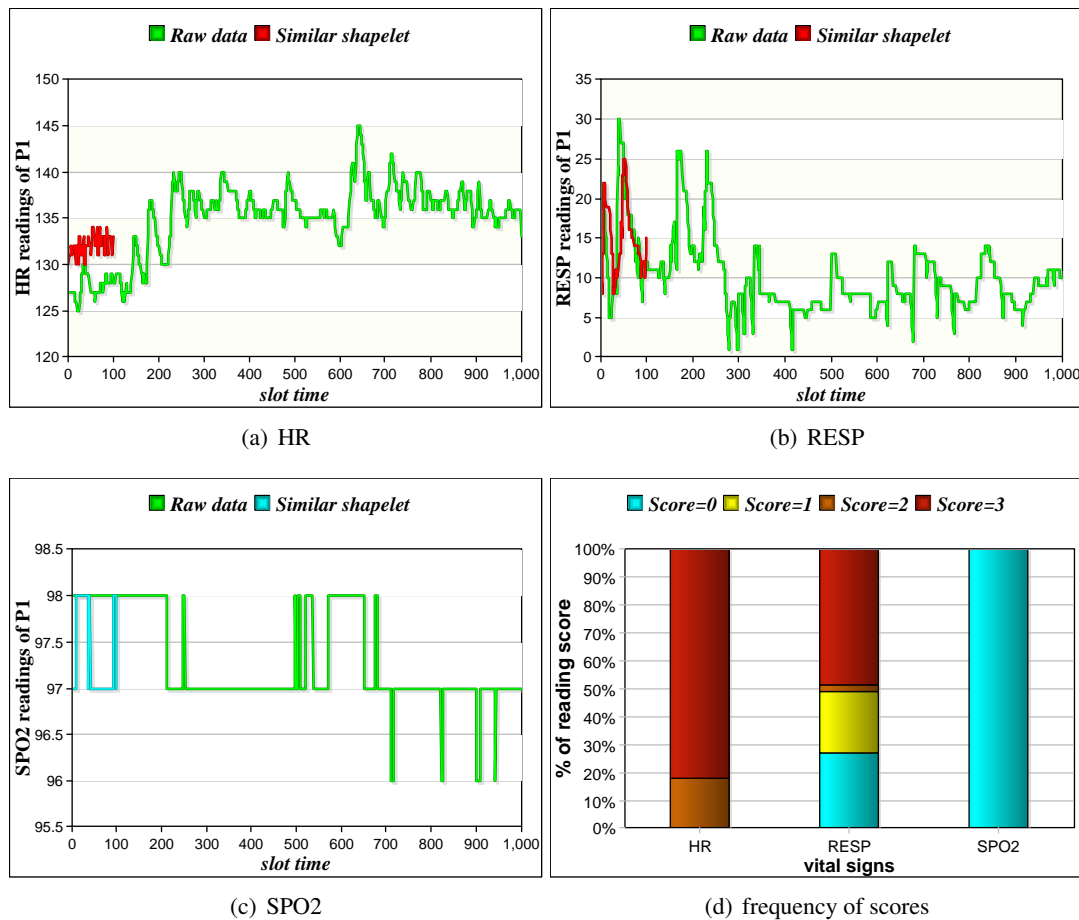


Figure 4.11 Similar shapelet elected for every vital sign of the high Patient 1.

4.4.3 Performance of Patient Classification Model

This section presents the results of the classification for two patients Patient 1 and Patient 9 (Figure 4.14 and Figure 4.15 respectively), obtained during the first round (10 periods). This technique depends on the level of the three vital signs (*HR*, *RESP*, *SPO2*). The results are represented in figures 4.14 with this rule: that if the level = 1 that means it is low, if equal to 2 it is medium , and if 3 their level is high. For the first period, Figure 4.14(a) shows the levels of the vital signs, where *HR* is high, *RESP* is high, and *SPO2* is low, gives a high level of total criticality in Figure 4.14(b). But in the second period *HR* remains high and likewise *SPO2* remains low but *RESP* was high and becomes medium, consequently the criticality decreases from high level to medium level, and so on.

4.4.4 Patients Assigning to Nurses Study

This framework allows each nurse to control a specific group of patients, during a period, by the patients distribution method following their criticality levels. Therefore, this framework prevents a nurse to getting a group of patients where all of them are in critical condition, and their health requires time to serve them.

Performance of PDCB and PSG

Figure 4.16 shows the slight difference between the two clustering-based mechanisms proposed in this paper. The nurses number is fixed to three, and the patients number is 18. So the goal is to distribute the 18 patients on the three nurses. The shown results are for after the first period. Figure 4.16(a) for mechanism 1, there are seven high level patients (represented in red color). The high

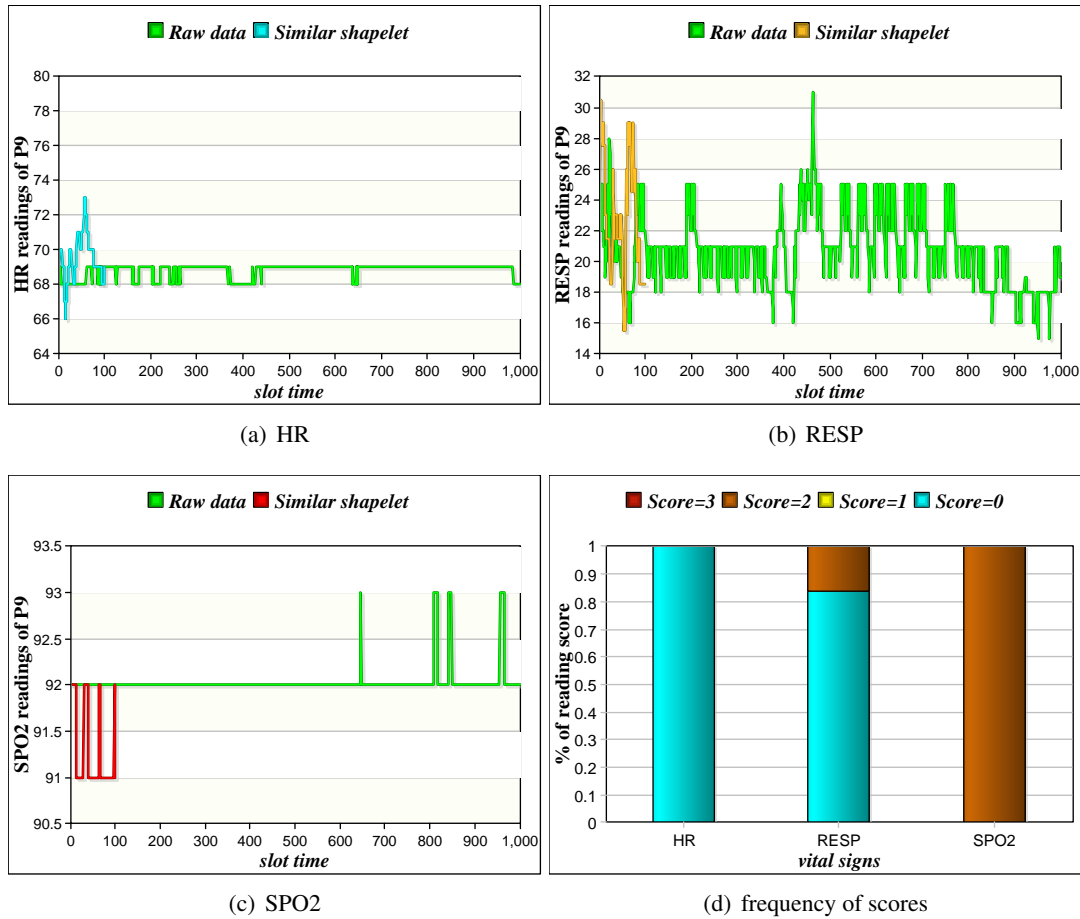


Figure 4.12 Similar shapelet elected for every vital sign of the medium Patient 9.

patients are distributed three for Nurse 1 , two for Nurse 2, and two for Nurse 3. And there are nine medium level patients (in orange color), they are distributed two for Nurse 1, three for Nurse 2, and four for Nurse 3. Also, there are two low level patients (in blue color), they are joined with the group of the nurse Nurse 2. Note that nurse Nurse 1 that has more high level patients(three) has two medium patient, but the other has three and four medium patients. Here Figure 4.16(b) shows the results of the second mechanism (2), also the seven high patients and the nine medium patients are distributed in the same way as the first mechanism. But the two low patients are distributed between the 1st nurse (Nurse 1) and the 2nd (Nurse 2). These figures verify the performance of these two methods in equal distribution and diversity of criticality levels for patients in each cluster.

In order to evaluate the accuracy of this two grouping mechanisms, the number of patient obtained for each nurse and the distribution of levels (Figure 4.17) are studied:

- Number of patients in each cluster: The equality in the number of patient and their levels for each nurse, is one of the principal goal that determine the efficiency of the distribution. This is confirmed in the pictures Figure 4.17(a) and Figure 4.17(b) (mechanism 1 and mechanism 2 respectively), where in the 1st mechanism the distribution of patients is in the following form: five for the 1st nurse (Nurse 1), seven for the 2nd (Nurse 2), and six for the third nurse (Nurse 3). In the second mechanism the distribution is equal to nurses, e.g six in each cluster.
- Accumulation criticality: For a fair distribution between the nurses, the volume of criticality in each cluster should be almost equal. These pictures Figure 4.18 show that the two mechanisms gave excellent results, by distributing the high critical patients to the nurses, not by grouping them into one group, but by dispersing them over the groups. As well as for the medium

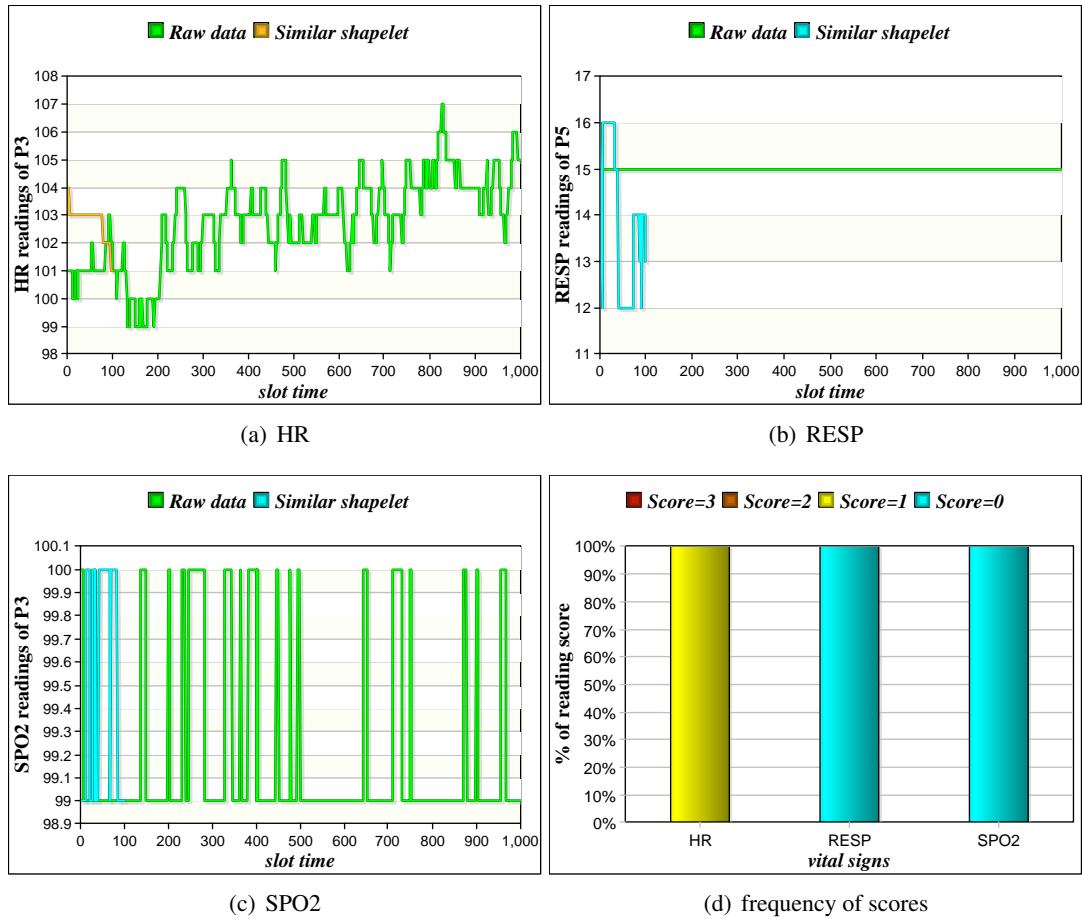


Figure 4.13 Similar shapelet elected for every vital sign of the low Patient 3.

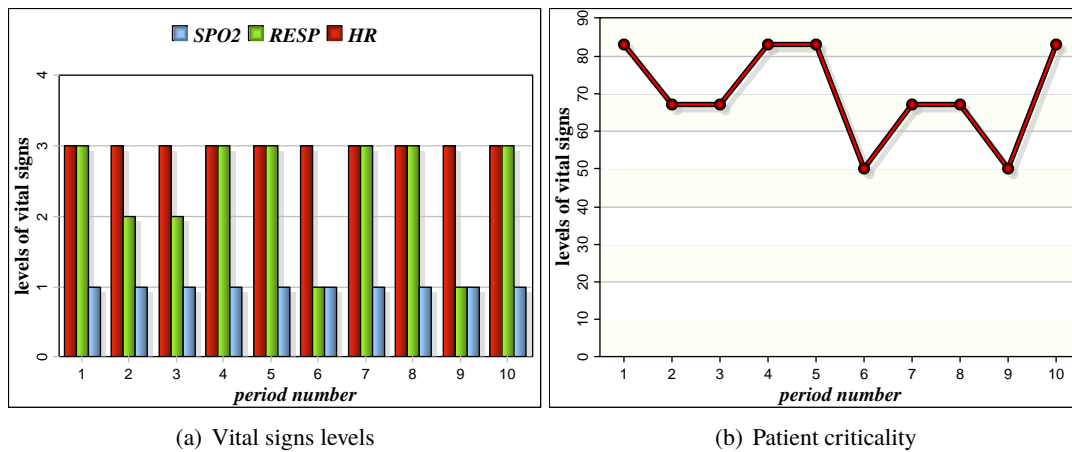


Figure 4.14 The criticalities levels of Patient 1 during 10 periods.

patients, and joining the low critical patient in the appropriate cluster in terms of number and total of the criticality rate.

Performance of GA-PSO

We tested GA-PSO with two groups of criticality: ST_1 and ST_2 . In ST_1 , the nurse assigns 17.5 and 8.5 minutes to serve patients with high and medium criticality respectively, while patient with low

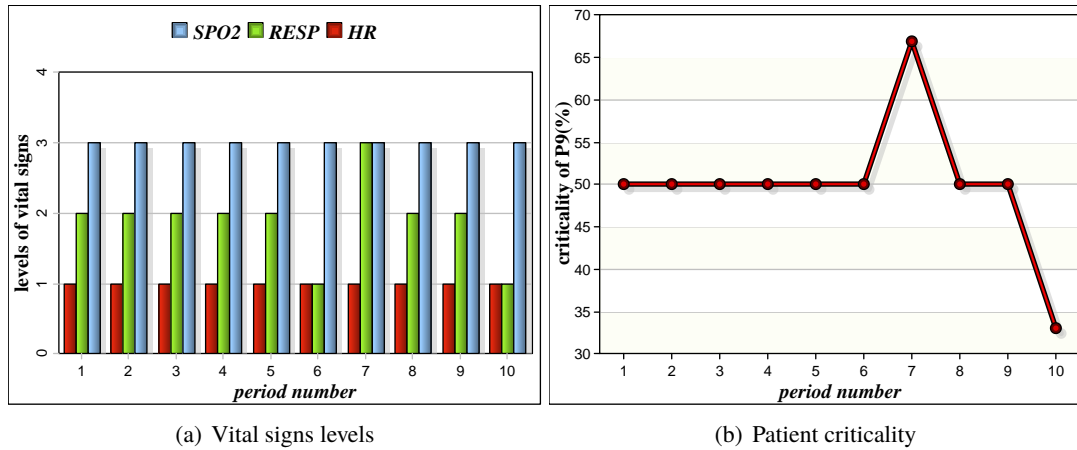


Figure 4.15 The criticalities levels of Patient 9 during 10 periods.

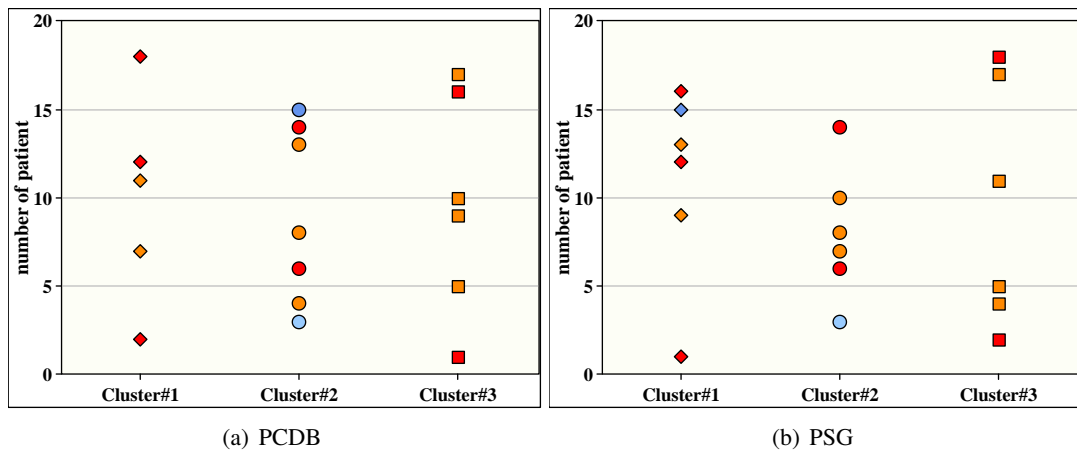


Figure 4.16 Distribution of patients after the first period.

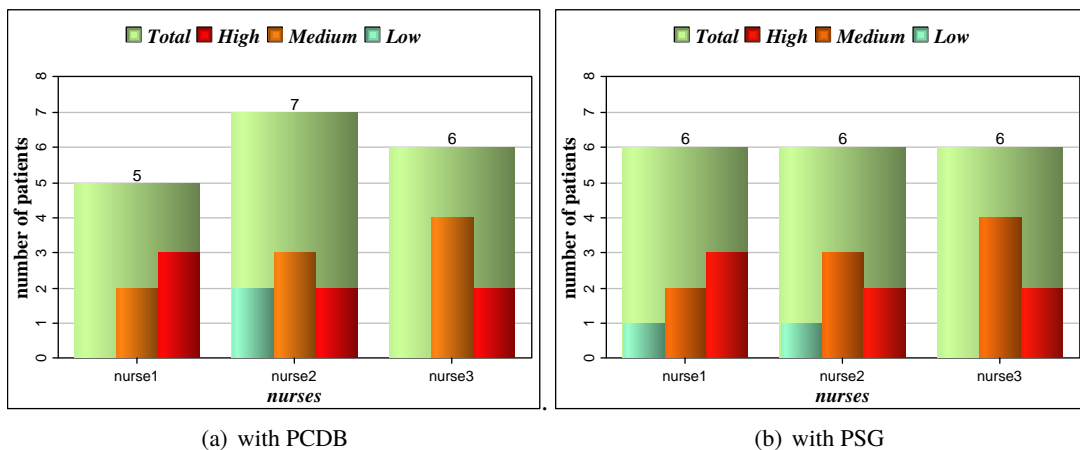


Figure 4.17 The variation of the patient number in each cluster.

criticality level does not assigned any serving time. In ST_2 , the nurse assigns 10, 6 and 1 minutes to serve patients with high, medium and low levels respectively. Furthermore, our mechanism GA-PSO presents a schedule for each nurse to follow-up the patients and the necessary time to be served that

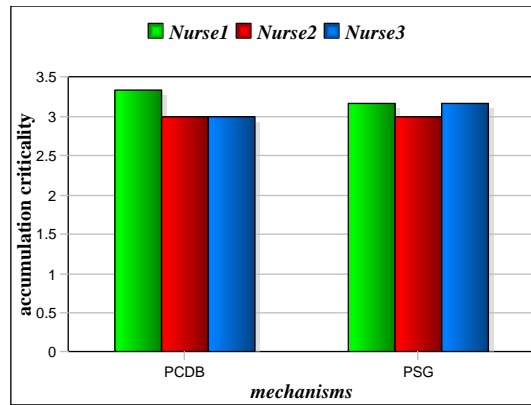


Figure 4.18 Accumulation criticality in each cluster.

the nurse should follow it to treat. Both tables 4.4 and 4.5 verify the performance of our mechanism GA-PSO in diversity of criticality levels for patients for each nurse. In order to evaluate the fairness of our mechanism, the number of patient obtained for each nurse and the distribution of levels are studied:

- **Number of patients for each Nurse:** A fair equality in the number of patients and their criticality levels for each nurse, is one of the principal goal that determine the efficiency of the distribution. It is confirmed in the tables 4 and 5, where by comparing the served patients with the number of nurses available there is a fair distribution. For instance, in the first approach ST_1 , the distribution of Fifteen patients between three nurses is in the following form: five for the Nurse1, four for the Nurse2, and six for the Nurse3. As for the second approach ST_2 , the distribution is four for the Nurse1, five for the Nurse2, and six for the Nurse3.
- **Accumulation criticality:** For a fair distribution between the nurses, the volume of criticality for each nurse should be almost equal. The tables 4.4 and 4.5 show that our mechanism with different parameters give excellent results, by distributing the high critical patients to the nurses, not by grouping them into one group, but by dispersing them over the groups. As well as for the moderate patients and distributing the low critical patient for the appropriate nurse in terms of number and total of the criticality.
- **Performance Evaluation of GA-PSO in function of Criticality**

The volume of criticality should be nearly equal for a fair distribution among the nurses by distributing the high-risk patients to the nurses in groups rather than in a single group, as well as to distribute the medium and low patients in the proper group.

Our GA-PSO mechanism with two simulations produced reasonable results by ensuring fairness in distribution among the nurses where taking into consideration the criticality of each patient. For instance, where $NN = 3$ and $NP = 15$ we observe that in the ST_1 the distribution of patients is in the following form: five for the 1st nurse, four for the 2nd nurse, and six for the third nurse. As for ST_2 the distribution is better than ST_1 , four are allocated for the 1st nurse, five for the 2nd nurse, and six for the third nurse. Consequently, we observe that the criticality values affect the optimization, as we balance the 3 factors with appropriate numbers with a lower gap such as 1, 6, 10 leads to a better performance than 0, 8.5, 17.5.

- **Performance Evaluation of GA-PSO in function of Dataset:** This section explored the effect of dataset size on the performance of our mechanism. GA-PSO has shown to perform better on dataset of larger size with respect to the length of the number of nurses and the

NN=3, NP=15	Served patients	Service time (minutes)	Patient Distribution (%)	NN=8, NP=30	Served patients	Service time (minutes)	Patient Distribution (%)	NN=15, NP=50	Served patients	Service time (minutes)	Patient Distribution (%)
Nurse 1	5 : {2, 1, 2}	43.5	33.46	Nurse 1	4 : {1, 1, 2}	26	12.59	Nurse 1	4 : {1, 1, 2}	26	6.83
Nurse 2	4 : {2, 2, 0}	52	40.00	Nurse 2	3 : {1, 2, 0}	34.5	16.71	Nurse 2	2 : {1, 1, 0}	26	6.83
Nurse 3	6 : {1, 2, 3}	34.5	26.54	Nurse 3	5 : {1, 2, 2}	34.5	16.71	Nurse 3	3 : {1, 0, 2}	17.5	4.60
				Nurse 4	5 : {1, 2, 2}	34.5	16.71	Nurse 4	3 : {1, 0, 2}	17.5	4.60
				Nurse 5	4 : {0, 2, 2}	17	8.23	Nurse 5	4 : {0, 2, 2}	17	4.47
				Nurse 6	4 : {0, 2, 2}	17	8.23	Nurse 6	4 : {0, 2, 2}	17	4.47
				Nurse 7	3 : {0, 2, 1}	17	8.23	Nurse 7	4 : {1, 2, 1}	34.5	9.07
				Nurse 8	2 : {1, 1, 0}	26	12.59	Nurse 8	3 : {1, 1, 1}	26	6.83
								Nurse 9	4 : {1, 2, 1}	34.5	9.07
								Nurse 10	3 : {1, 1, 1}	26	6.83
								Nurse 11	4 : {1, 2, 1}	34.5	9.07
								Nurse 12	3 : {1, 1, 1}	26	6.83
								Nurse 13	3 : {1, 1, 1}	26	6.83
								Nurse 14	3 : {1, 1, 1}	26	6.83
								Nurse 15	3 : {1, 1, 1}	26	6.83
Total	15.00	260.00	100.00	30.00	413.00	100.00	50.00	380.50	100.00	380.5	100

Table 4.4 GA-PSO results with ST_1 .

NN=3, NP=15	Served patients	Service time (minutes)	Patient Distribution (%)	NN=8, NP=30	Served patients	Service time (minutes)	Patient Distribution (%)	NN=15, NP=50	Served patients	Service time (minutes)	Patient Distribution (%)
Nurse 1	4 : {2, 1, 1}	27	31.76	Nurse 1	3 : {1, 1, 1}	17	9.88	Nurse 1	3 : {1, 1, 1}	17	5.84
Nurse 2	5 : {2, 2, 1}	33	38.82	Nurse 2	3 : {1, 2, 0}	22	12.79	Nurse 2	4 : {2, 0, 2}	22	7.56
Nurse 3	6 : {1, 2, 3}	25	29.41	Nurse 3	3 : {1, 2, 0}	22	12.79	Nurse 3	2 : {2, 0, 0}	20	6.87
				Nurse 4	4 : {1, 2, 1}	23	13.37	Nurse 4	4 : {1, 1, 2}	18	6.19
				Nurse 5	5 : {1, 2, 2}	24	13.95	Nurse 5	3 : {1, 0, 2}	12	4.12
				Nurse 6	5 : {1, 2, 2}	24	13.95	Nurse 6	4 : {2, 0, 2}	22	7.56
				Nurse 7	4 : {1, 2, 1}	23	13.37	Nurse 7	3 : {1, 0, 2}	12	4.12
				Nurse 8	3 : {1, 1, 1}	17	9.88	Nurse 8	3 : {2, 0, 1}	21	7.22
								Nurse 9	3 : {2, 0, 1}	21	7.22
								Nurse 10	3 : {1, 0, 2}	12	4.12
								Nurse 11	3 : {2, 0, 1}	21	7.22
								Nurse 12	4 : {2, 1, 1}	27	9.28
								Nurse 13	5 : {1, 2, 2}	24	8.25
								Nurse 14	3 : {2, 0, 1}	21	7.22
								Nurse 15	3 : {2, 0, 1}	21	7.22
Total	15	85	100		30	172	100		50	291	100

Table 4.5 GA-PSO results with ST_2 .

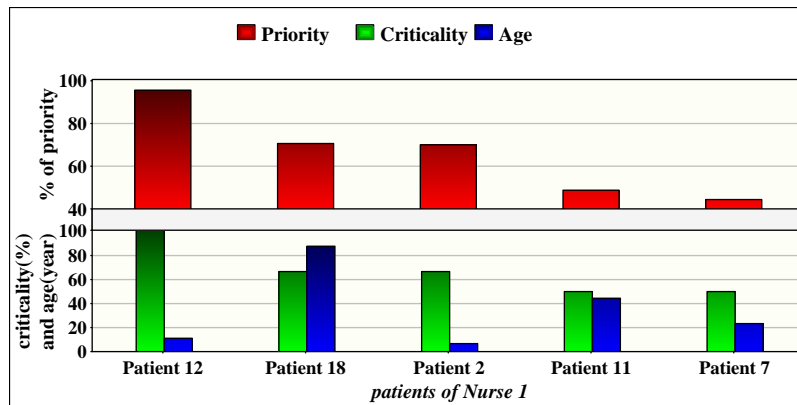
number of patients. Reasonable accuracy is also observed in GA-PSO mechanism, particularly for larger dataset, as it was shown in the experimental results of tables 4.4 and 4.5.

We observe that on a low number of nurses and patients, the optimization process is less accurate, which means that distribution is not as fair as possible. However, when increasing these parameters the optimization goes up as it increases due to a better combination of patients' chromosomes. Performance wise, increasing the parameters will lead to a slight delay addition but this could be tweaked by decreasing the number of iterations. On the other hand, any random combination of patients selected as gbest shows the equal distribution of these patients over any specified number of nurses considering the fact that every patient has a certain priority, these nurses must serve patients in order, and as distributed inside the chromosome per nurse. If we try to confirm this condition, we can sum up all the number of patients per nurse and verify with the criticality values by adding them once these patients are categorized as high, medium or low. For instance, in the tables shown above, we can determine the percentage of service for each nurse with the number of patients distributed equally over the dataset and for each scenario the designated set of criticality measures. As we also note, the number of patients is ranged between two certain numbers close to each other taking into consideration the priority and number of patients per nurse. For example, in both tables 4.4 and 4.5 where $NN = 15$ and $NP = 50$, it is ranged from 2 to 5 depending on how critical the patients are and the schedule of each nurse. As we see here, the gap between nurses is around 10% in distribution terms for 3 nurses, which is even lower for 8 and also lower for 15 nurses. We observe that this gap decreases when the number of nurses and patients increase, and thus the performance of the algorithm. If we consider the other set of criticalities below, it is observed that this distribution is better which leads to a low percentage of difference, so depending on the gap of the criticality measures, the performance increases when this gap decreases, and the 0 for "Low" is not the best practice as it adds nothing to the nurse's schedule. Not only the distribution is fair over number of patients but also for the category of criticality, nurses go on different missions with low, medium and high categories which also depend on the number of patients per nurse. For example in the set of criticalities below, 6 low patients are considered equal to one medium patients and a high patient is a combination of both. Finally, the service time is properly managed in a low range of gaps between each nurse, so a nurse might have a total time of 26, the other at 34 and so on, and this gap is almost equal when changing parameters such as number of patients and nurses.

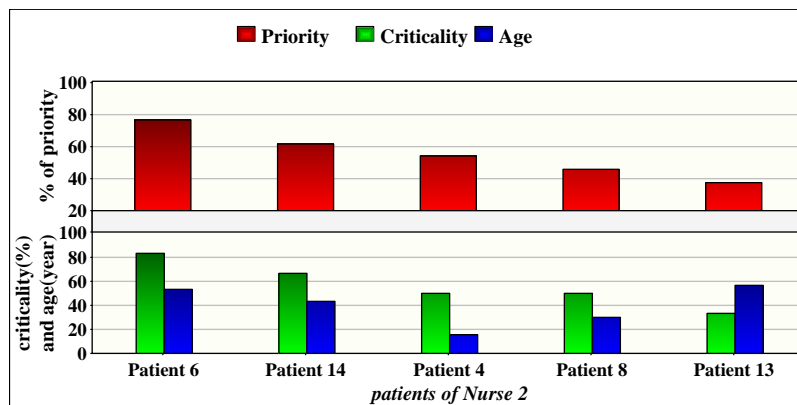
4.4.5 Performance of Patients Scheduling

This section verify the efficiency of the final step of this framework, that called patients scheduling. This step aims to organise and range the patients for each nurse, in order to follow and serve them during each period. These figures 4.19 shows the priorities of the patients of the three groups of nursing (Nurse 1, Nurse 2, Nurse 3), according to the criticality and the age of each patient. By making the weight of criticality $\lambda = 0.8$ greater than the weight of age $\psi = 0.2$. The pink rectangular bar in each figure presents the patient criticality detected, the blue rectangular bar presents the age of the patient, and the light yellow rectangular bar for their priority calculated. It is noted that if the criticality of the vital signs of two patients are equal, the priority becomes for the patient who has one critical age. Figure 4.19(a) shows the different priorities for patients assigned to the Nurse 1. The Patient 12 is ranked first in terms of priority in the examination, where it has the max criticality rate and their age is 11 years. The two patients Patient 18 and Patient 2 have the same criticality, but the priority is for the Patient 18 where their age equal to 87 years grater critical than the age of P_2 equal to seven years. In Figure 4.19(b), the criticalities rate of patients Patient 4 and Patient 8 are equal, priority was given to the patient Patient 4, as his age is 16 years more critical to the age of Patient 8 ($= 30$ years). And in Figure 4.19(c) shows the priorities assigned to the third Nurse (3),

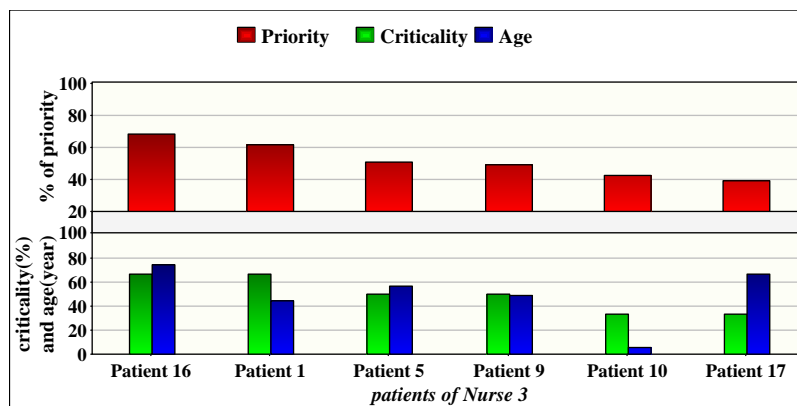
where the patients were arranged as follows: Patient 16, Patient 1, Patient 5, Patient 9, Patient 10, Patient P17.



(a) Nurse 1



(b) Nurse 2

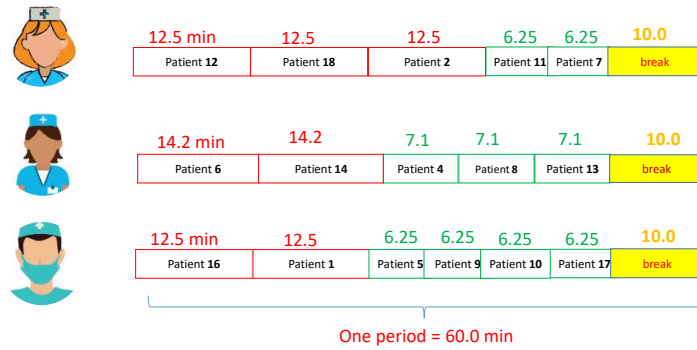


(c) Nurse 3

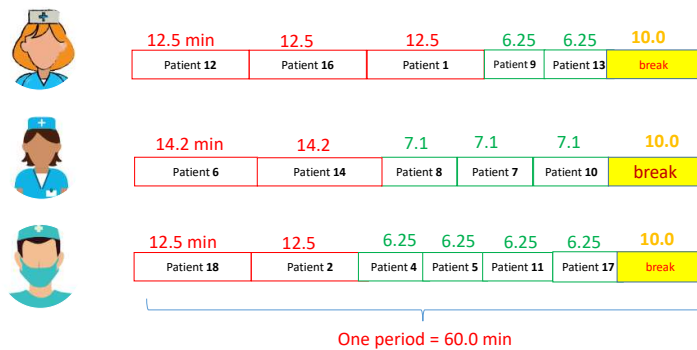
Figure 4.19 The priority level of patients obtained in function of age and criticality rate.

Figure 4.20 presents for each nurse the schedule after the first period, to follow-up the patients during the second period. And this figure shows for each patient the appropriate time, that the nurse should follow it to treat and service them during one period(= 60 min). While respecting the nurse’s responsibility, where the nurse can increase the permissible time in the event it is necessary, by giving him a period of time as an addition ($\delta = 10 \text{ min}$), as shown in figures 4.20 as a break. With the two mechanisms (PCDB and PSG), the first Nurse 1 has the

time to follow the high patient equal to 12.5 minutes, and for medium patient the time equal to 6.25 min. The Nurse 2 has more time to treat the high patient (14.2 min), and that because there is two high critical patients less than for Nurse 1 where has three high critical patients.



(a) with PCDB mechanism



(b) with PSG mechanism

Figure 4.20 The scheduling results obtained.

4.5 Conclusion

In public health, the patient classification based on wireless body sensor network and the appointment scheduling techniques can significantly reduce the workload of the medical staff. In this chapter, we have proposed an efficient framework for intelligent nurse-patient scheduling that consists of two phases: patient classification, and nurse scheduling. Our framework aims to classify patients according to their criticalities, and introduce an efficient nurse scheduling. Through simulations on real health data, we demonstrated the effectiveness of our framework in providing a balanced nurse

workloads where each nurse serves, during each period of one hour, an average of 5 patients with corresponding cumulative criticality of 62 %.

5 Conclusion and Perspectives

5.1 Conclusion

The healthcare applications relies on the WBSN which is the main technology that plays an important role in monitoring healthcare status. In addition, the researchers have gained insights about the important role of nurses in hospitals. Indeed, WBSN is becoming a staple for helping nurses in several tasks such as: emergency detection; controlling vital signs; data analysis and making decision. Thus, in this work we focused on proposing several techniques that improves this tasks and takes into account the conserving the limited sensor energy.

At sensor level, we proposed several techniques that aims to reduce the energy consumption, in order to extend the network lifetime. These techniques are based on reducing the amount of collected and transmitted data from the sensors. First, an emergency detection algorithm is proposed, that aims to send data only if the patient's condition has varied in order to reduce the redundant transmitted data. Second, we presented an adaptive sensing frequency that aims to eliminate the redundant collected data, based on the patient's situation. Third, we built the first phase of the patient classification model that compares the vital sign series of the patient with the vital signs series provided from the archive using the DTW method, in order to select and send the criticality level of the vital sign to the sink. Finally, we proposed a multi-hop protocol that minimize the required signal strength of sensor in transmission process and therefore reducing the energy consumption. Third,

At sink level, we proposed several technique that allows the medical staff to make the appropriate decision. First, we built an efficient framework based on hadoop tools for big data collection, processing and storage in wireless body sensor network. The proposed framework aims to ingest data in real time (using Flume and Spark), then preprocess them and store the big data for analyzing process (using HDFS), and finally allows the end user to analyse the data (using Grafana and Matplotlib). Second, we proposed a patient records archive technique in order to regenerate the health data of each patient for analysis purposes. Third, we proposed a prediction technique that allows to predict any abnormal situation, using the prophet method, in order to obtain the right medical intervention on time. Fourth, we built a patient classification model that classify the patient according to its criticality which is calculated according to the received criticality levels from the all sensors. Finally, We proposed a patient-nurse scheduling which started by grouping mechanisms that aim to assign patients to nurses in a fair way, we proposed two mechanisms that assigns the patients according to the two parameters: patient's criticality level and the number of patients, and we proposed third mechanism which is an hybrid of the genetic algorithm and the particle swarm optimization. Then, proposing a scheduling algorithm that aim to find the path that the nurse should

follow in order to examine her group of patients. Which this algorithm is based on the patient's priority that calculated according to its criticality level and age.

5.2 Perspectives

In this section, we present two directions of perspectives in order to ameliorate the frameworks of patients controlling and assessment presented in this thesis: Short to Mid Term Perspectives that related to our proposed mechanisms and Log Term Perspectives.

5.2.1 Short to Mid Term Perspectives

As future work, we propose some perspectives in order to enhance the mechanisms presented in this thesis, which can be summarized as follows:

- Our approach in chapter 2 can be extended by adapting the sensor sensing frequency according to the patient's criticality. Indeed, the sampling rate of the sensors could be adapted and therefore reducing the energy consumption.
- We have two main directions to extend our framework in chapter 3. First, we plan to use aggregation techniques at the processing stage in order to eliminate the redundancy among the collected data thus reduce the amount of data storage. We want to search the similarity between the collected vital signs using several aggregation functions such as jaccard similarity; cosin similarity or several distance functions like euclidean; cosine. Second, we seek to enhance the performance of our framework by testing another Hadoop tools such as Hive; kafka.
- We have three main directions to enhance our framework in chapter 3. First, we plan to test our platform in real-case scenarios in order to validate its performance. Second, we seek to adapt our platform to take into account various types of patient data like images for organs, video for operations, etc. The purpose of adding more information (images and videos) to the digital data is to support the decision making. Finally, we plan to develop a mobile application in order to help clinicians closely and remotely monitor critical patients.
- Our platform in chapter 4 can be enhanced in several ways. First, we seek to integrate a mobile application for the medical staff in order to determine for each nurse their scheduling appointments. Second, we plan to improve the performance of our framework by testing another parameters in the patients scheduling algorithm in addition to the criticality rate and the age of the patients, such as the capability of each nurse.
- We have three future directions to enhance NPS mechanism presented in chapter 4. First, we seek to take into account more criteria when assigning nurses to patients, such as nurse skills and decision subjectivity. Second, we plan to enhance the service time of each patient in a dynamic way rather than static one. Third, we seek to increase the performance of NPS either by enhancing the process of GA-PSO or by testing another heuristic algorithms.

5.2.2 Long Term Perspectives

Despite the big attention payed by the researchers in order to improve the healthcare applications, the field is still open to research. Subsequently, several issues need to be more explored, that are related on the data management; energy conservation, decision making; and patients-nurse scheduling. In this section, we give some issues in order to attract the attention of researchers.

The first issue is based on the data collection techniques in WBSN. The integration of the Federated Learning (FL) technique in the process of the collect and transmit data prolong the network lifetime and reduce the complexity of decision making.

The second issue is related to the patients-nurse scheduling technique. Indeed, in the hospital a nurse is distinguished from a nurse in activity, intelligence and experience. Therefore, in addition to identifying the group of patients, assigning the right nurse to this group improve the provided health services.

Finally, the time synchronization is one of the issue that is not yet largely explored. Any loss or delayed cause a change in the time synchronization at the sink and therefore lead a maltreatment. Subsequently, more effort should be made for enhancing the time synchronization in WBSN in order to prevent any undesired patient's situation.

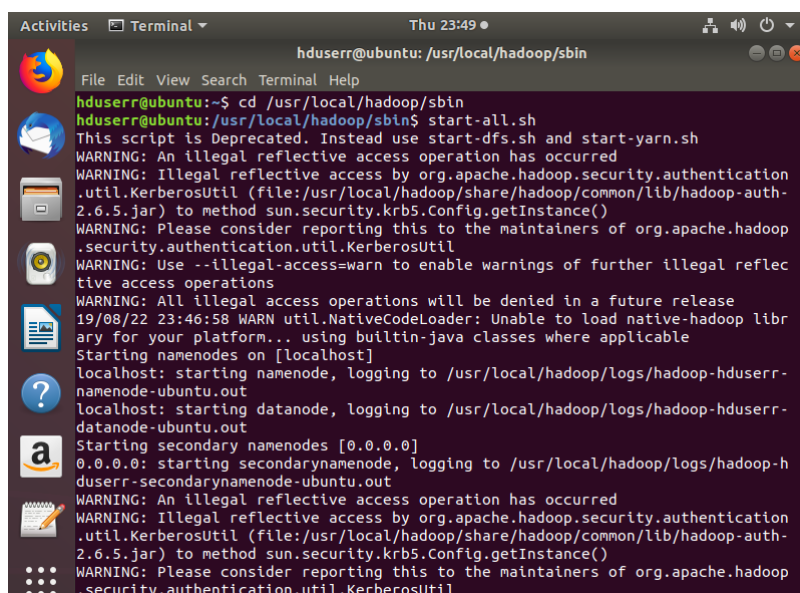
Appendix A

Hadoop Implementation

The main objective of using Hadoop is to perform parallel data storage and processing thus it reduces the effect of data loss during node failure and decreases the processing latency. Therefore, we used hadoop tools in Chapter 3 in order to handle the big data generated from the patients.

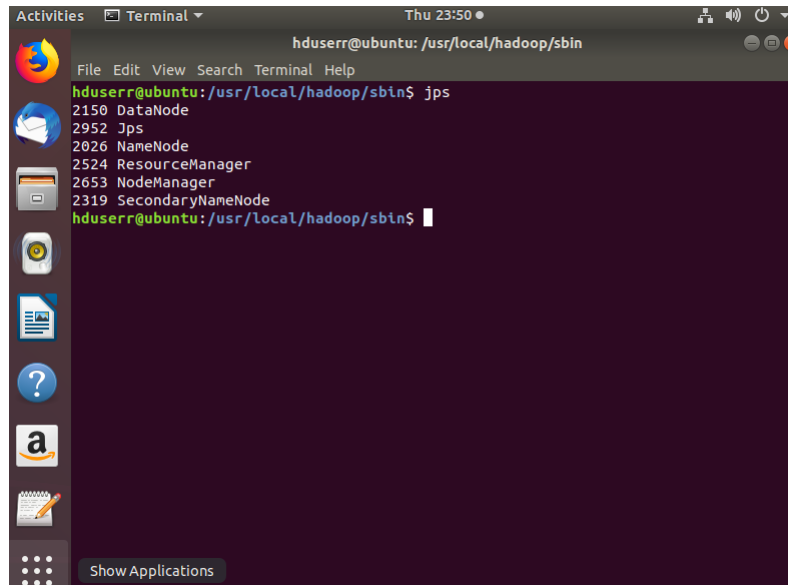
A.1 Hadoop Starting

To show the effectiveness of our framework in store and process the big data with the diversity of information in real time and the speed of generation, we implemented the system on hadoop using ubuntu, Figure A.1 presents the starting of a single hadoop node and Figure A.2 confirms that hadoop is well installed.



```
Activities Terminal Thu 23:49
hduser@ubuntu: /usr/local/hadoop/sbin
File Edit View Search Terminal Help
hduser@ubuntu:~$ cd /usr/local/hadoop/sbin
hduser@ubuntu: /usr/local/hadoop/sbin$ start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication
.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-
2.6.5.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop
.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflec
tive access operations
WARNING: All illegal access operations will be denied in a future release
19/08/22 23:46:58 WARN util.NativeCodeLoader: Unable to load native-hadoop libr
ary for your platform... using builtin-java classes where applicable
Starting namenodes on [localhost]
localhost: starting namenode, logging to /usr/local/hadoop/logs/hadoop-hduser-
namenode-ubuntu.out
localhost: starting datanode, logging to /usr/local/hadoop/logs/hadoop-hduser-
datanode-ubuntu.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to /usr/local/hadoop/logs/hadoop-h
duser-secondarynamenode-ubuntu.out
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication
.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-
2.6.5.jar) to method sun.security.krb5.Config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop
.security.authentication.util.KerberosUtil
```

Figure A.1 Hadoop starting.



```

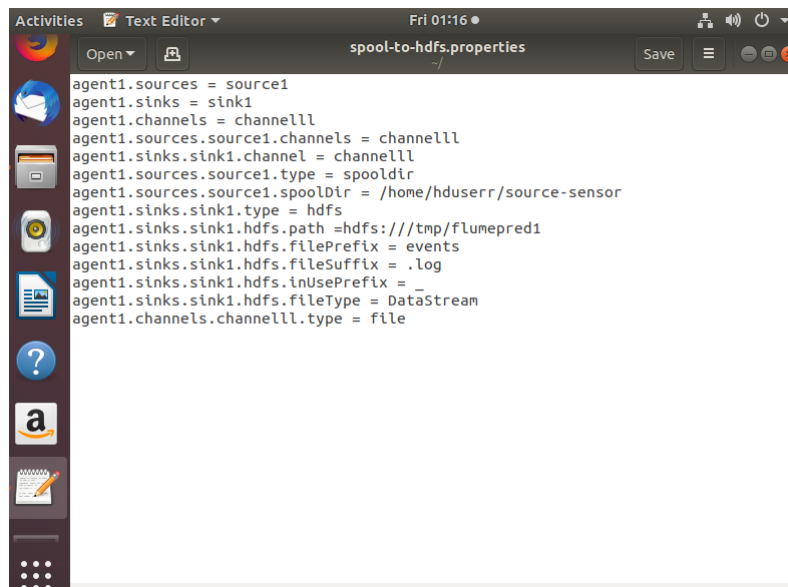
hduser@ubuntu: /usr/local/hadoop/sbin$ jps
2150 DataNode
2952 Jps
2026 NameNode
2524 ResourceManager
2653 NodeManager
2319 SecondaryNameNode
hduser@ubuntu: /usr/local/hadoop/sbin$

```

Figure A.2 jps.

A.2 Results based on Data Collected and Storage

The tool responsible for collecting data in our system is flume, Figure A.3 illustrated the properties of flume. Flume stay active to be ready to receive data in real time shown in Figure A.4(a) and deliver them for storage or processing purpose. In Figures A.4(b) and A.4(c) show that the data is well delivered, and Figure A.4(d) confirms for users that the data is stored in HDFS.



```

agent1.sources = source1
agent1.sinks = sink1
agent1.channels = channelll
agent1.sources.source1.channels = channelll
agent1.sinks.sink1.channel = channelll
agent1.sources.source1.type = spooldir
agent1.sources.source1.spoolDir = /home/hduser/source-sensor
agent1.sinks.sink1.type = hdfs
agent1.sinks.sink1.hdfs.path =hdfs:///tmp/flumepred1
agent1.sinks.sink1.hdfs.filePrefix = events
agent1.sinks.sink1.hdfs.fileSuffix = .log
agent1.sinks.sink1.hdfs.inUsePrefix = _
agent1.sinks.sink1.hdfs.fileType = DataStream
agent1.channels.channelll.type = file

```

Figure A.3 Properties of flume.

```

Fri 00:00
hduser@ubuntu:~$
File Edit View Search Terminal Help
hduser@ubuntu:~$
2019-08-23 00:00:07.517 [lifecycleSupervisor-1-3] [INFO - org.apache.flume.source.SpooledDirectorySource.start(SpooledDirectorySource.java:85)] SpooledDirectorySource source starting with directory /home/hduser/source-sensor
2019-08-23 00:00:07.540 [lifecycleSupervisor-1-1] [INFO - org.apache.flume.instrumentation.MonitoredCounterGroup.register(MonitoredCounterGroup.java:119)] Monitored counter group for type: SINK, name: sink1: Successfully registered new HBean.
2019-08-23 00:00:07.543 [lifecycleSupervisor-1-1] [INFO - org.apache.flume.instrumentation.MonitoredCounterGroup.start(MonitoredCounterGroup.java:95)] Component type: SINK, name: sink1 started
2019-08-23 00:00:07.807 [SinkRunner-pollingRunner-DefaultSinkProcessor] [INFO - org.apache.flume.sink.hdfs.HDFSDatastream.configure(HDFSDatastream.java:57)] Serializer: TEXT, UseRawLocalFileSystem = false
2019-08-23 00:00:07.891 [lifecycleSupervisor-1-3] [INFO - org.apache.flume.instrumentation.MonitoredCounterGroup.register(MonitoredCounterGroup.java:119)] Monitored counter group for type: SOURCE, name: source1: Successfully registered new HBean.
2019-08-23 00:00:07.876 [lifecycleSupervisor-1-3] [INFO - org.apache.flume.instrumentation.MonitoredCounterGroup.start(MonitoredCounterGroup.java:95)] Component type: SOURCE, name: source1 started
2019-08-23 00:00:09.073 [SinkRunner-pollingRunner-DefaultSinkProcessor] [INFO - org.apache.flume.sink.hdfs.BucketWriter.open(BucketWriter.java:246)] Creating hdfs://tmp/flume-sensor_events:1565705509848.log.tmp
2019-08-23 00:00:11.070 [hdfs-sink1-call-runner-0] [WARN - org.apache.hadoop.util.NativeCodeLoader.<clinit>(NativeCodeLoader.java:62)] Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
    
```

(a) Flume starting

```

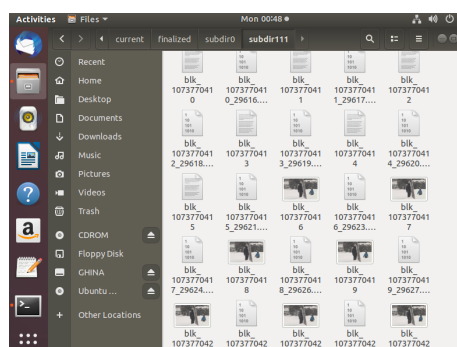
Fri 23:46
hduser@ubuntu:~/jusr/focal/hadoop
File Edit View Search Terminal Help
hduser@ubuntu:~/jusr/focal/hadoop$ bin/hadoop dfs -cat /tmp/flume-sensor/*
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.6.5.jar) to method sun.security.krb5.config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
19/08/23 23:44:41 warn util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
2004-02-28 00:59:16.02785 1 1 19.9884 37.0933 45.08 2.69964
2004-02-28 00:59:16.02785 2 1 19.9884 37.0933 45.08 2.69964
2004-02-28 00:59:16.02785 3 1 19.9884 37.0933 45.08 2.69964
2004-02-28 00:59:16.02785 4 1 19.9884 37.0933 45.08 2.6996
2004-02-28 00:59:16.02785 5 1 19.9884 37.0933 45.08 2.6996
2004-02-28 00:59:16.02785 6 1 19.9884 37.0933 45.08 2.6996
2004-02-28 00:59:16.02785 7 1 19.9884 37.0933 45.08 2.6996
2004-02-28 00:59:16.02785 8 1 19.9884 37.0933 45.08 2.6996
2004-02-28 00:59:16.02785 9 1 19.9884 37.0933 45.08 2.6996
2004-02-28 00:59:16.02785 10 1 19.9884 37.0933 45.08 2.6996
2004-02-28 01:03:16.33393 11 1 19.3024 38.4629 45.08 2.68742
2004-02-28 01:03:16.33393 12 1 19.8512 37.3672 45.08 2.6972
    
```

(b) Data divided to events

```

Fri 23:43
hduser@ubuntu:~/jusr/focal/hadoop
File Edit View Search Terminal Help
hduser@ubuntu:~/jusr/focal/hadoop$ bin/hadoop dfs -ls /tmp/flume-sensor
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.
WARNING: An illegal reflective access operation has occurred
WARNING: Illegal reflective access by org.apache.hadoop.security.authentication.util.KerberosUtil (file:/usr/local/hadoop/share/hadoop/common/lib/hadoop-auth-2.6.5.jar) to method sun.security.krb5.config.getInstance()
WARNING: Please consider reporting this to the maintainers of org.apache.hadoop.security.authentication.util.KerberosUtil
WARNING: Use --illegal-access=warn to enable warnings of further illegal reflective access operations
WARNING: All illegal access operations will be denied in a future release
19/08/23 23:43:26 warn util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
Found 3079 items
-rw-r--r-- 1 hduser supergroup 604 2019-08-13 07:12 /tmp/flume-sensor/events.1565705509843.log
-rw-r--r-- 1 hduser supergroup 616 2019-08-13 07:12 /tmp/flume-sensor/events.1565705509845.log
-rw-r--r-- 1 hduser supergroup 621 2019-08-13 07:12 /tmp/flume-sensor/events.1565705509845.log
-rw-r--r-- 1 hduser supergroup 624 2019-08-13 07:12 /tmp/flume-sensor/events.1565705509846.log
-rw-r--r-- 1 hduser supergroup 631 2019-08-13 07:12 /tmp/flume-sensor/events.1565705509847.log
-rw-r--r-- 1 hduser supergroup 635 2019-08-13 07:12 /tmp/flume-sensor/events.1565705509848.log
    
```

(c) Data with prediction data stored



(d) Data with prediction data stored

Figure A.4 Results based on Data Collected and Storage.

Publications

1 Accepted and Submitted Journals

- 1 Ghina Saad, Abdelhafid Abouaissa, Lhassane Idoumghar, Hassan Harb and Nour Charara. P2D: An Efficient Patient-to-Doctor Framework for Real-Time Health Monitoring and Decision Making. In IEEE Sensors Journal, pp. 14240-14252, 2020.
- 2 Ghina Saad, Hassan Harb, Abdelhafid Abouaissa, Lhassane Idoumghar and Nour Charara. A Sensing-Based Patient Classification Framework for Efficient Patient Nurse Scheduling. Submitted to the Sustainable Computing: Informatics and Systems Journal.
- 3 Aline Abboud, Ghina SAAD, Abdelhafid Abouaissa, Lhassane Idoumghar, Hassan Harb and Nour Charara. An Efficient Mechanism for Nurse-Patient Management Through IoT and Optimisation Technologies. Submitted to Applied Soft Computing Journal.

2 Accepted and Published Conferences

- 1 Ghina Saad, Abdelhafid Abouaissa, Lhassane Idoumghar, Hassan Harb and Nour Charara. An Efficient Hadoop-Based Framework for Data Storage and Fault Recovering in Large Scale Multimedia Sensor Networks. In IWCMC 2020, 16th International Wireless Communications and Mobile Computing Conference, June 15-19, 2020 - Limassol, Cyprus, pages 316-321.
- 2 Ghina Saad , Hassan Harb, Abdelhafid Abouaissa, Lhassane Idoumghar and Nour Charara. Recent Advances of Patient Monitoring in Internet of Healthcare Things: A Comparative Study. In IWCMC 2022, 18th International Wireless Communications Mobile Computing Conference, 30 May - 3 Jun, 2022 - Valamar Dubrovnik President Hotel, Croatia.

Bibliography

- [1] Y. Qu, G. Zheng, H. Wu, B. Ji, and H. Ma, "An energy-efficient routing protocol for reliable data transmission in wireless body area networks," *Sensors*, vol. 19, no. 19, p. 4238, 2019.
- [2] R. A. Khan, Q. Xin, and N. Roshan, "Rk-energy efficient routing protocol for wireless body area sensor networks," *Wireless Personal Communications*, vol. 116, no. 1, pp. 709–721, 2021.
- [3] C. Habib, A. Makhoul, R. Darazi, and C. Salim, "Self-adaptive data collection and fusion for health monitoring based on body sensor networks," *IEEE transactions on Industrial Informatics*, vol. 12, no. 6, pp. 2342–2352, 2016.
- [4] R. Zhang, A. Nayak, and J. Yu, "Sleep scheduling in energy harvesting wireless body area networks," *IEEE Communications Magazine*, vol. 57, no. 2, pp. 95–101, 2019.
- [5] S. Ghatole and P. Dahikar, "Survey on wireless body area network for healthcare applications," *International Journal of Researches in Biosciences, Agriculture and Technology (IJRBAT)*, vol. 4, no. 3, pp. 14–17, 2016.
- [6] K. Yoshioka-Maeda, C. Honda, and R. Iwasaki-Motegi, "Preparing for complex emergencies while combating covid-19: The role of public health nurses in japan," *Asia Pacific Journal of Public Health*, vol. 32, no. 8, pp. 489–490, 2020.
- [7] J. Kim, A. S. Campbell, B. E.-F. de Ávila, and J. Wang, "Wearable biosensors for healthcare monitoring," *Nature biotechnology*, vol. 37, no. 4, pp. 389–406, 2019.
- [8] R. K. Kodali, G. Swamy, and B. Lakshmi, "An implementation of iot for healthcare," in *2015 IEEE Recent Advances in Intelligent Computational Systems (RAICS)*. IEEE, 2015, pp. 411–416.
- [9] A. Alkhayyat, A. A. Thabit, F. A. Al-Mayali, and Q. H. Abbasi, "Wbsn in iot health-based application: toward delay and energy consumption minimization," *Journal of Sensors*, vol. 2019, 2019.
- [10] M. Asif-Ur-Rahman, F. Afsana, M. Mahmud, M. S. Kaiser, M. R. Ahmed, O. Kaiwartya, and A. James-Taylor, "Toward a heterogeneous mist, fog, and cloud-based framework for the internet of healthcare things," *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 4049–4062, 2018.
- [11] F. Ullah, I. U. Islam, A. H. Abdullah, and A. Khan, "Future of big data and deep learning for wireless body area networks," pp. 53–77, 2019. [Online]. Available: https://doi.org/10.1007/978-981-13-3459-7_5

- [12] S. Majumder, T. Mondal, and M. J. Deen, "Wearable sensors for remote health monitoring," *Sensors*, vol. 17, no. 1, p. 130, 2017.
- [13] S. Majumder, E. Aghayi, M. Noferesti, H. Memarzadeh-Tehran, T. Mondal, Z. Pang, and M. J. Deen, "Smart homes for elderly healthcare—recent advances and research challenges," *Sensors*, vol. 17, no. 11, p. 2496, 2017.
- [14] M. J. Deen, "Information and communications technologies for elderly ubiquitous healthcare in a smart home," *Personal and Ubiquitous Computing*, vol. 19, no. 3-4, pp. 573–599, 2015.
- [15] R. A. Khan and A.-S. K. Pathan, "The state-of-the-art wireless body area sensor networks: A survey," *International Journal of Distributed Sensor Networks*, vol. 14, no. 4, p. 1550147718768994, 2018.
- [16] L. Syed, S. Jabeen, S. Manimala, and H. A. Elsayed, "Data science algorithms and techniques for smart healthcare using iot and big data analytics," in *Smart Techniques for a Smarter Planet*. Springer, 2019, pp. 211–241.
- [17] R. Zhang and J. Yu, "Energy-efficient and reliable sleep scheduling algorithms in wbsns," in *Energy-Efficient Algorithms and Protocols for Wireless Body Sensor Networks*. Springer, 2020, pp. 101–121.
- [18] M. K. M. Rabby, M. S. Alam, and M. S. A. Shawkat, "A priority based energy harvesting scheme for charging embedded sensor nodes in wireless body area networks," *PloS one*, vol. 14, no. 4, 2019.
- [19] R. A. Khan, K. H. Mohammadani, A. A. Soomro, J. Hussain, S. Khan, T. H. Arain, and H. Zafar, "An energy efficient routing protocol for wireless body area sensor networks," *Wireless Personal Communications*, vol. 99, no. 4, pp. 1443–1454, 2018.
- [20] S. Y. Park, D. Y. Yun, T. Kim, J.-Y. Lee, D. Lee *et al.*, "An energy efficient enhanced dual-fuzzy logic routing protocol for monitoring activities of the elderly using body sensor networks," *Electronics*, vol. 9, no. 5, p. 723, 2020.
- [21] K. Dhakal, A. Alsadoon, P. Prasad, R. S. Ali, L. Pham, and A. Elchouemi, "A novel solution for a wireless body sensor network: Telehealth elderly people monitoring," *Egyptian Informatics Journal*, vol. 21, no. 2, pp. 91–103, 2020.
- [22] K. M. Awan, N. Ashraf, M. Q. Saleem, O. E. Sheta, K. N. Qureshi, A. Zeb, K. Haseeb, and A. S. Sadiq, "A priority-based congestion-avoidance routing protocol using iot-based heterogeneous medical sensors for energy efficiency in healthcare wireless body area networks," *International Journal of Distributed Sensor Networks*, vol. 15, no. 6, p. 1550147719853980, 2019.
- [23] N. Javaid, A. Ahmad, Q. Nadeem, M. Imran, and N. Haider, "im-simple: improved stable increased-throughput multi-hop link efficient routing protocol for wireless body area networks," *Computers in Human Behavior*, vol. 51, pp. 1003–1011, 2015.
- [24] L. Li and D. Li, "An energy-balanced routing protocol for a wireless sensor network," *Journal of Sensors*, vol. 2018, 2018.
- [25] M. M. Sahndhu, N. Javaid, M. Imran, M. Guizani, Z. A. Khan, and U. Qasim, "Bec: A novel routing protocol for balanced energy consumption in wireless body area networks," in *2015 International Wireless Communications and Mobile Computing Conference (IWCMC)*. IEEE, 2015, pp. 653–658.

- [26] Z. Khan, N. Aslam, S. Sivakumar, and W. Phillips, "Energy-aware peering routing protocol for indoor hospital body area network communication," *Procedia Computer Science*, vol. 10, pp. 188–196, 2012.
- [27] H. Alwan and A. Agarwal, "Mqosr: A multiobjective qos routing protocol for wireless sensor networks," *International Scholarly Research Notices*, vol. 2013, 2013.
- [28] A. Alameen and A. Gupta, "Clustering and classification based real time analysis of health monitoring and risk assessment in wireless body sensor networks," *Bio-Algorithms and Med-Systems*, vol. 15, no. 4, 2019.
- [29] P. V. Raja and E. Sivasankar, "Modern framework for distributed healthcare data analytics based on hadoop," in *Information and Communication Technology-EurAsia Conference*. Springer, 2014, pp. 348–355.
- [30] A. S. Abiodun, M. H. Anisi, I. Ali, A. Akhunzada, and M. K. Khan, "Reducing power consumption in wireless body area networks: a novel data segregation and classification technique," *IEEE Consumer Electronics Magazine*, vol. 6, no. 4, pp. 38–47, 2017.
- [31] S. Qiu, L. Liu, Z. Wang, S. Li, H. Zhao, J. Wang, J. Li, and K. Tang, "Body sensor network-based gait quality assessment for clinical decision-support via multi-sensor fusion," *IEEE Access*, vol. 7, pp. 59 884–59 894, 2019.
- [32] Y. Liu, L. Zhang, Y. Yang, L. Zhou, L. Ren, F. Wang, R. Liu, Z. Pang, and M. J. Deen, "A novel cloud-based framework for the elderly healthcare services using digital twin," *IEEE Access*, vol. 7, pp. 49 088–49 101, 2019.
- [33] J. Xu, B. S. Glicksberg, C. Su, P. Walker, J. Bian, and F. Wang, "Federated learning for healthcare informatics," *Journal of Healthcare Informatics Research*, vol. 5, no. 1, pp. 1–19, 2021.
- [34] M. J. Sousa, A. M. Pesqueira, C. Lemos, M. Sousa, and Á. Rocha, "Decision-making based on big data analytics for people management in healthcare organizations," *Journal of medical systems*, vol. 43, no. 9, pp. 1–10, 2019.
- [35] P. K. Sadineni, "Developing a model to enhance the quality of health informatics using big data," in *2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC)*. IEEE, 2020, pp. 1267–1272.
- [36] A. Rehman, K. Haseeb, T. Saba, J. Lloret, and U. Tariq, "Secured big data analytics for decision-oriented medical system using internet of things," *Electronics*, vol. 10, no. 11, p. 1273, 2021.
- [37] M. Koussaifi, C. Habib, and A. Makhoul, "Real-time stress evaluation using wireless body sensor networks," in *2018 Wireless Days (WD)*. IEEE, 2018, pp. 37–39.
- [38] C. Habibe, A. Makhoul, R. Darazi, and R. Couturier, "Health risk assessment and decision-making for patient monitoring and decision-support using wireless body sensor networks," *Information Fusion*, vol. 47, pp. 10–22, 2019.
- [39] G. Cappon, L. Cossu, F. Boscari, D. Bruttomesso, G. Sparacino, and A. Facchinetti, "An integrated mobile platform for automated data collection and real-time patient monitoring in diabetes clinical trials," *Journal of Diabetes Science and Technology*, p. 19322968211024620, 2021.

- [40] D. Ajerla, S. Mahfuz, and F. Zulkernine, "A real-time patient monitoring framework for fall detection," *Wireless Communications and Mobile Computing*, vol. 2019, 2019.
- [41] H. Harb, A. Mansour, A. Nasser, E. M. Cruz, and I. de la Torre Diez, "A sensor-based data analytics for patient monitoring in connected healthcare applications," *IEEE Sensors Journal*, vol. 21, no. 2, pp. 974–984, 2020.
- [42] L. Schick, W. L. d. Souza, and A. F. d. Prado, "Wireless body sensor network for monitoring and evaluating physical activity," in *Information Technology-New Generations*. Springer, 2018, pp. 81–86.
- [43] A. Harbouche, N. Djedi, M. Erradi, J. Ben-Othman, and A. Kobbane, "Model driven flexible design of a wireless body sensor network for health monitoring," *Computer Networks*, vol. 129, pp. 548–571, 2017.
- [44] F. Aiello, F. L. Bellifemine, G. Fortino, S. Galzarano, and R. Gravina, "An agent-based signal processing in-node environment for real-time human activity monitoring based on wireless body sensor networks," *Engineering Applications of Artificial Intelligence*, vol. 24, no. 7, pp. 1147–1161, 2011.
- [45] S. B. Tambe and S. S. Gajre, "Cluster-based real-time analysis of mobile healthcare application for prediction of physiological data," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 2, pp. 429–445, 2018.
- [46] H. Harb, H. Mroue, A. Mansour, A. Nasser, and E. Motta Cruz, "A hadoop-based platform for patient classification and disease diagnosis in healthcare applications," *Sensors*, vol. 20, no. 7, p. 1931, 2020.
- [47] A. Alameen and A. Gupta, "Optimization driven deep learning approach for health monitoring and risk assessment in wireless body sensor networks," *International Journal of Business Data Communications and Networking (IJBDCN)*, vol. 16, no. 1, pp. 70–93, 2020.
- [48] A. Lanata, A. Greco, S. Di Modica, F. Niccolini, F. Vivaldi, F. Di Francesco, C. Tamantini, F. Cordella, L. Zollo, M. Di Rienzo *et al.*, "A new smart-fabric based body area sensor network for work risk assessment," in *2020 IEEE International Workshop on Metrology for Industry 4.0 & IoT*. IEEE, 2020, pp. 187–190.
- [49] M. Maniruzzaman, M. J. Rahman, B. Ahammed, and M. M. Abedin, "Classification and prediction of diabetes disease using machine learning paradigm," *Health information science and systems*, vol. 8, no. 1, pp. 1–14, 2020.
- [50] E. Deniz, A. Şengür, Z. Kadiroğlu, Y. Guo, V. Bajaj, and Ü. Budak, "Transfer learning based histopathologic image classification for breast cancer detection," *Health information science and systems*, vol. 6, no. 1, pp. 1–7, 2018.
- [51] C. Avram, O. Stan, C. Domuța, V. Mureșan, C. Răcășan, V. Sita, and A. Aștilean, "Scheduling method for non-emergency medical cases," in *2020 IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR)*. IEEE, 2020, pp. 1–6.
- [52] R. Alizadeh, J. Rezaeian, M. Abedi, and R. Chiong, "A modified genetic algorithm for non-emergency outpatient appointment scheduling with highly demanded medical services considering patient priorities," *Computers & Industrial Engineering*, vol. 139, p. 106106, 2020.
- [53] S. Lee and Y. H. Lee, "Improving emergency department efficiency by patient scheduling using deep reinforcement learning," in *Healthcare*, vol. 8, no. 2. MDPI, 2020, p. 77.

- [54] D. Daldoul, I. Nouaouri, H. Bouchriha, and H. Allaoui, "Scheduling patients in emergency department: A case study," in *2017 IEEE International Conference on Industrial Engineering and Engineering Management (IEEM)*. IEEE, 2017, pp. 870–874.
- [55] M. P. Ariyani, C. N. Rosyidi, and A. Aisyati, "An optimization model of nurse scheduling using goal programming method: a case study," vol. 1096, no. 1, p. 012022, 2021.
- [56] S.-J. Kim, Y.-W. Ko, S. Uhm, and J. Kim, "A strategy to improve performance of genetic algorithm for nurse scheduling problem," *International journal of software engineering and its applications*, vol. 8, no. 1, pp. 53–62, 2014.
- [57] H. Jafari and N. Salmasi, "Maximizing the nurses' preferences in nurse scheduling problem: mathematical modeling and a meta-heuristic algorithm," *Journal of industrial engineering international*, vol. 11, no. 3, pp. 439–458, 2015.
- [58] P. Punnaakitikashem, J. M. Rosenberger, D. B. Behan, R. L. Baker, and K. Goss, "An optimization-based prototype for nurse assignment," in *Proceedings of the 7th Asian Pacific industrial engineering and management systems conference*. Citeseer, 2006, pp. 17–20.
- [59] X. Zhong, J. Zhang, and X. Zhang, "A two-stage heuristic algorithm for the nurse scheduling problem with fairness objective on weekend workload under different shift designs," *IIEE transactions on healthcare systems engineering*, vol. 7, no. 4, pp. 224–235, 2017.
- [60] L. M. Steege and J. G. Dykstra, "A macroergonomic perspective on fatigue and coping in the hospital nurse work system," *Applied ergonomics*, vol. 54, pp. 19–26, 2016.
- [61] A. Amindoust, M. Asadpour, and S. Shirmohammadi, "A hybrid genetic algorithm for nurse scheduling problem considering the fatigue factor," *Journal of Healthcare Engineering*, vol. 2021, 2021.
- [62] I.-C. JUNG, Y.-S. KIM, S.-R. IM, and C.-H. IHM, "A development of nurse scheduling model based on q-learning algorithm," *Korea Journal of Artificial Intelligence*, vol. 9, no. 1, pp. 1–7, 2021.
- [63] O. M. Alade and A. O. Amusat, "Solving nurse scheduling problem using constraint programming technique," *arXiv preprint arXiv:1902.01193*, 2019.
- [64] F. Chahyadi, S. Azhari, and H. Kurniawan, "Hospital nurse scheduling optimization using simulated annealing and probabilistic cooling scheme," *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, vol. 12, no. 1, pp. 21–32, 2018.
- [65] S. B. Baker, W. Xiang, and I. Atkinson, "Internet of things for smart healthcare: Technologies, challenges, and opportunities," *IEEE Access*, vol. 5, pp. 26 521–26 544, 2017.
- [66] J. Qi, P. Yang, G. Min, O. Amft, F. Dong, and L. Xu, "Advanced internet of things for personalised healthcare systems: A survey," *Pervasive and Mobile Computing*, vol. 41, pp. 132–149, 2017. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S1574119217303255>
- [67] Ramesh, Chitra and Sekar, Jayalakshmi, "A survey on healthcare systems using internet of things," *ITM Web Conf.*, vol. 37, p. 01015, 2021. [Online]. Available: <https://doi.org/10.1051/itmconf/20213701015>

- [68] W. A. N. W. Abdullah, N. Yaakob, M. E. Elobaid, M. N. M. Warip, and S. A. Yah, "Energy-efficient remote healthcare monitoring using iot: A review of trends and challenges," in *Proceedings of the international conference on internet of things and cloud computing*, 2016, pp. 1–8.
- [69] K. T. Kadhim, A. M. Alsahlany, S. M. Wadi, and H. T. Kadhum, "An overview of patient's health status monitoring system based on internet of things (iot)." *Wireless Personal Communications*, vol. 114, no. 3, 2020.
- [70] H. Kaur, M. Atif, and R. Chauhan, "An internet of healthcare things (ioht)-based healthcare monitoring system," in *Advances in Intelligent Computing and Communication*, M. N. Mohanty and S. Das, Eds. Singapore: Springer Singapore, 2020, pp. 475–482.
- [71] M. Javaid and I. H. Khan, "Internet of things (iot) enabled healthcare helps to take the challenges of covid-19 pandemic," *Journal of Oral Biology and Craniofacial Research*, vol. 11, no. 2, pp. 209–214, 2021.
- [72] U. Royal College of Physicians, London, "National early warning score (news)," <http://www.rcplondon.ac.uk/resources/national-early-warning-score-news>, 2015.
- [73] "Mimic database on physionet,," in <https://www.physionet.org/>, accessed: 2016-10-4, 2000-present.
- [74] O. Bretscher, "Linear algebra with applications.(2005)."
- [75] C. Combes, F. Kadri, and S. Chaabane, "Predicting hospital length of stay using regression models: Application to emergency department," in *10ème Conférence Francophone de Modélisation, Optimisation et Simulation-MOSIM'14*, 2014.
- [76] J. F. Hair Jr, M. Sarstedt, L. Hopkins, and V. G. Kuppelwieser, "Partial least squares structural equation modeling (pls-sem) an emerging tool in business research," *European Business Review*, vol. 26, no. 2, pp. 106–121, 2014.
- [77] M. Gilli, D. Maringer, and E. Schumann, *Numerical methods and optimization in finance*. Academic Press, 2019.
- [78] C. Severance, "The apache software foundation: Brian behlendorf," *Computer*, vol. 45, no. 10, pp. 8–9, 2012.
- [79] A. Spark, "Apache spark," *Retrieved January*, vol. 17, no. 1, p. 2018, 2018.
- [80] D. He, S. Chan, Y. Zhang, and H. Yang, "Lightweight and confidential data discovery and dissemination for wireless body area networks," *IEEE journal of biomedical and health informatics*, vol. 18, no. 2, pp. 440–448, 2013.
- [81] C. Habib, A. Makhoul, R. Darazi, and R. Couturier, "Real-time sampling rate adaptation based on continuous risk level evaluation in wireless body sensor networks," in *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*. IEEE, 2017, pp. 1–8.
- [82] P. L. Anderson, M. M. Meerschaert, and K. Zhang, "Forecasting with prediction intervals for periodic autoregressive moving average models," *Journal of Time Series Analysis*, vol. 34, no. 2, pp. 187–193, 2013.
- [83] S. X. Chen, L. Lei, and Y. Tu, "Functional coefficient moving average model with applications to forecasting chinese cpi," *Statistica Sinica*, pp. 1649–1672, 2016.

- [84] “Grafana labs,” 2014.
- [85] J. Hunter and D. Dale, “The matplotlib user’s guide,” *Matplotlib 0.90.0 user’s guide*, 2007.
- [86] H. Alharthi, “Healthcare predictive analytics: An overview with a focus on saudi arabia,” *Journal of infection and public health*, vol. 11, no. 6, pp. 749–756, 2018.
- [87] I. N. Soyiri and D. D. Reidpath, “An overview of health forecasting,” *Environmental health and preventive medicine*, vol. 18, no. 1, p. 1, 2013.
- [88] I. Medina, D. Montaner, J. Tárraga, and J. Dopazo, “Prophet, a web-based tool for class prediction using microarray data,” *Bioinformatics*, vol. 23, no. 3, pp. 390–391, 2007.
- [89] H. Tyrallis and G. A. Papacharalampous, “Large-scale assessment of prophet for multi-step ahead forecasting of monthly streamflow,” *Advances in Geosciences*, vol. 45, pp. 147–153, 2018.
- [90] Q. Chen, H. Yang, M. Guo, R. S. Kannan, J. Mars, and L. Tang, “Prophet: Precise qos prediction on non-preemptive accelerators to improve utilization in warehouse-scale computers,” in *Proceedings of the Twenty-Second International Conference on Architectural Support for Programming Languages and Operating Systems*, 2017, pp. 17–32.
- [91] I. Yenidoğan, A. Çayır, O. Kozan, T. Dağ, and Ç. Arslan, “Bitcoin forecasting using arima and prophet,” in *2018 3rd International Conference on Computer Science and Engineering (UBMK)*. IEEE, 2018, pp. 621–624.
- [92] M. Navratil and A. Kolkova, “Decomposition and forecasting time series in business economy using prophet forecasting model.” *Central European Business Review*, vol. 8, no. 4, 2019.
- [93] O. Amjad, E. Bedeer, N. Abu Ali, and S. Ikki, “Robust energy efficiency optimization algorithm for health monitoring system with wireless body area networks,” *IEEE Communications Letters*, vol. 24, no. 5, pp. 1142–1145, 2020.